Drawing knot pictures using LaTeXwith XY-pic – An introduction and tutorial

Lars Fischer*

January 2003

Abstract: XY-pic is a TeX-macro package for typesetting graphs and diagrams. It is very modular and there is support for typesetting *mathematical knots and links*. XY-pic's documentation consists of a User's-Guide [Ros99] and a Reference Manual [RM99] available on XY-pic's homepage [Ros]. Unfortunately the knot feature is not covered in detail and (not only) this section is missing some basic examples.

This text is not intended to replace [RM99] or [Ros99]. I want to introduce the knots and link feature under LaTeX and more important I want to give some examples.

Contents

1 Basic training **2**
- 1.1 Loading the package in a LaTeX document . . . 2
- 1.2 Hello unknot . . . 2
- 1.3 Moving around . . . 3
  - 1.3.1 (Hopf-link) . . . 3
  - 1.3.2 Advanced Movements . . . 4
- 1.4 Drawing a straight line . . . 4
- 1.5 Arrowtips . . . 4
- 1.6 Labels . . . 5

2 Real world examples **5**
- 2.1 The Reidemeister moves . . . 5
  - 2.1.1 First Reidemeister Move . . . 6
  - 2.1.2 Second Reidemeister Move . . . 6
  - 2.1.3 Third Reidemeister Move . . . 7
- 2.2 Trefoil is made up with polygons . . . 7
- 2.3 The figure-8 knot . . . 8

*http://www.lars.fischer.de.vu

3 Troubleshooting **9**
3.1 TEX-capacity exceeded 9

A Tables **10**

1 Basic training

While reading the XY-pic documentation, one must keep in mind, that XY-pic is written for TEX, so constructs look different here than in the documentation [RM99], which describes the usage with TEX.

1.1 Loading the package in a LATEX document

At first you must load XY-pic into your document, the next line loads XY-pic and enables the knot feature:

```
\usepackage[all,knot]{xy}
```

In the following I assume, your document contains at least these lines (the `poly` options is needed later):

```
\documentclass{article}
  \usepackage[all,knot,poly]{xy}
\begin{document}
  % -- put the examples here ---
\end{document}
```

The example's code is put somewhere between `\begin{document}` and `\end{document}`.

1.2 Hello unknot

The first example is the simplest knot, you can think of – the unknot:

```
\xygraph{
  !{0;/r2.0pc/:}
  !{\vcap-}
  !{\vcap}
}
```

typesets 

The line `!{0;/r2.0pc/:}` scales the picture, use different values instead of `2.0pc` to get different sizes.

In the Reference Manual [RM99] there are two tables showing every crossing, uncrossing and join. In the joins-table you can see that `\vcap` produces a half circle. (I have included copies of these tables, see figure 2 and 3.)

`\vcap-` is a shorthand for `\vcap[-1]`, which simply rotates the knot piece 180 degrees. Upper and lower half-circle made up the complete circle. If `[n]` is used with a positive number `n` the piece is scaled, `[-n]` scales and rotates.

Notice the use of `!{...}`, every XY-pic drawing command needs to be enclosed like this.

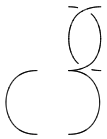
1.3 Moving around

1.3.1 (Hopf-link)

It is important to understand, that XY-pic maintains a `current cursor pos`. Drawing a knot piece changes this position. The next knot piece is drawn starting at the new position. Sometimes you can draw your knot pieces and the position is perfect for the next piece.

But in general one has to change the `current cursor pos` manually. The position is controlled by the `[u],[d],[l],[r]` commands. They change the position up, down, left, and right. Consider the next examples:

```
\xygraph{
  !{0;/r1.0pc/:}
  !{\vunder}
  !{\vunder-}
  !{\hcap[2]}
  [l]!{\hcap[-2]}
}
```



```
\xygraph{
  !{0;/r1.0pc/:}
  !{\vunder}
  !{\vunder-}
  [uur]!{\hcap[2]}
  [l]!{\hcap[-2]}
}
```

(Due to the `[l]` in the first example, it is possible to distinguish the two halfcircles.) The `!{\vunder}` changes the position in the right way, for the next `!{\vunder-}`, but the `!{\hcap[2]}` – the ")" shaped piece – is misplaced. In the second example the position is moved up two times and right once ([uur]). The two up movements are required because of the `!{\vunder} !{\vunder-}` combination. At last, the "(" shaped piece is placed after a jump to the left ([l]).

Notice the fact, that `!{\hcap[2]}`, which is a double sized piece, is exactly as high, as the `!{\vunder} !{\vunder-}` combination.

The tables in the Reference-Manual provide enough information, so that there is no need guessing where the position moves, while drawing a knot piece. But I don't understand enough of that, so I can't explain it here.

When typesetting a new knot diagram I use this approach (trial and error):

1. typeset the first two pieces and control their positions
2. if they are correct, add the others, piece by piece
3. after adding one piece control the position
4. only add a new piece, if the picture is correct

If the pictures get more complex and contain more than one piece of the same type, it is hard to distinguish which piece of this type is misplaced, if you add them simultaneously.

You need a fast computer, since the latex-run is time consuming. For typesetting a new knot diagramm, I use a separate empty document. After everything is perfect, I copy the lines in the real document.

1.3.2 Advanced Movements

If you want to move by a real factor, utilize the `[c(f)]` construct, where `f` is a real number like `0.5` or `2.3` and `c` a character $\in \{$`'u'`, `'d'`, `'l'`, `'r'`$\}$. You can move the whole picture, if your first command is a movement command; consider the next examples, featuring the bracket polynomial:

```
$\bigl<
   \xygraph{
     !{0;/r1.0pc/:}
     !{\xunderv}
   }
 \bigr>$
```

```
$\bigl<
   \xygraph{
     !{0;/r1.0pc/:}
     [u(0.5)]
     !{\xunderv}
   }
 \bigr>$
```

In the second example, the first command `[u(0.5)]` moves the current position up a half step. After that the knot piece is drawn.

1.4 Drawing a straight line

If you look through the tables in chapter 30 of [RM99], there is no obvious way, for producing a straight line. I am using a `\xcaph` which is curved, but it is possible to control the curvature. This is shown in the next example:

```
\xygraph{
  !{0;/r1.0pc/:}
  % straight line:
  !{\xcaph[2]@(0)} [ld]
  % this is curved:
  !{\xcaph[2]}
}
```

1.5 Arrowtips

Sometimes knots and links are oriented. XY-pic supports orientation with the following notations after a knot piece, consider the next examples and figure 1:

String	Arrowtip placement
=<	start
=>	finish
==	start and finish
=!	no arrowtips

String	Label placement
<	start
>	finish
|	middle

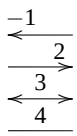
Figure 1: Placement-strings for arrowtips and labels

```
\xygraph{
  !{0;/r1.0pc/:}
  !{\xcaph[2]@(0)=>}
  [dl] !{\xcaph[2]@(0)=<}
  [dl] !{\xcaph[2]@(0)==}
}
```

1.6 Labels

With XY-pic it is possible to label parts of a diagram. Add either >,< or | and the text enclosed with curly braces, after your knot piece. Consider the next example and figure 1:

```
\xygraph{
  !{0;/r1.0pc/:}
  !{\xcaph[2]@(0)=>>{-1}}
  [dl] !{\xcaph[2]@(0)=<<{2}}
  [dl] !{\xcaph[2]@(0)==|{3}}
  [dl] !{\xcaph[2]@(0)|{4}}
}
```



2 Real world examples

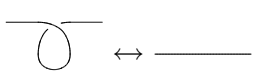
In the previous section, I showed simple examples. This section deals with complete examples and shows how to put everything together.

2.1 The Reidemeister moves

Every introduction in Knot theory notes the three Reidemeister moves, here you can see how to typeset them with XY-pic. It is also the first example made up of many knot pieces carefully arranged.

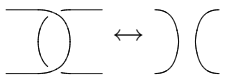
2.1.1 First Reidemeister Move

```
\xygraph{
$$ \xygraph{
     !{0;/r1.0pc/:}
     [u]
     !{\vover}
     !{\vcap-}
     [ul]!{\xcaph@(0)}
     [r]!{\xcaph@(0)}
    } \, \leftrightarrow \,
    \xygraph{
     !{0;/r1.0pc/:} [uu]
     !{\xcaph[3]@(0)}
  } $$
}
```



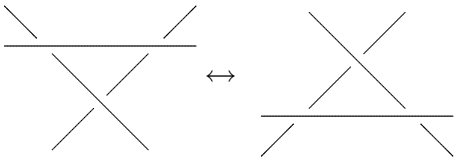
2.1.2 Second Reidemeister Move

```
$$ \xygraph{
     !{0;/r1.0pc/:}
     [u(0.8)]
     !{\xcaph@(0)}
     !{\vover}
     !{\vunder-}
     [l]!{\xcaph@(0)}
     [r]!{\xcaph@(0)}
     [uul]!{\xcaph@(0)}
   } \, \leftrightarrow \,
   \xygraph{
     !{0;/r1.0pc/:}
     [u(0.8)]!{\huncross[2]}
   } $$
}
```



2.1.3 Third Reidemeister Move

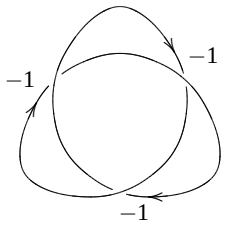
```
$$ \xygraph{
     !{0;/r1.0pc/:}
     [u(0.7)]
     !{\xoverh[3]}
     [ull][ul(0.5)]!{\sbendv@(0)}
     [rrrr]!{\sbendh@(0)}
     [llllll][d(1.25)]!{\xcaph[-6]@(0)}
   } \, \leftrightarrow \,
   \xygraph{
     !{0;/r1.0pc/:}
     [uu]
     !{\xoverh[3]}
     [llddddd][ld(0.5)]!{\sbendh@(0)}
     [rrrr]!{\sbendv@(0)}
     [llllll][u(1.25)]!{\xcaph[-6]@(0)}
   } $$
}
```



2.2 Trefoil is made up with polygons

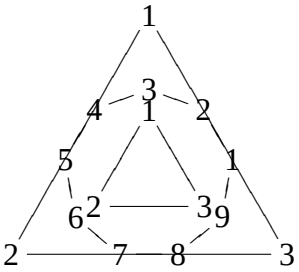
This section is based on the example given in [RM99]. It uses the polygon feature to arrange the knot pieces between the edges of stacked polygons. To do something similar you have to understand XY-pic's polygon feature described in [RM99], too. I have added orientation and labels, describing the writhe of the trefoil.

```
   \xygraph{
     !{0;/r2.0pc/:}
     !P3"a"{~>{}}
     !P9"b"{~:{(1.3288,0):}~>{}}
     !P3"c"{~:{(2.5,0):}~>{}}
     !{\vunder~{"b2"}{"b1"}{"a1"}{"a3"}<{-1}}
     !{\vcap~{"c1"}{"c1"}{"b4"}{"b2"}=<}
     !{\vunder~{"b5"}{"b4"}{"a2"}{"a1"}<{-1}}
     !{\vcap~{"c2"}{"c2"}{"b7"}{"b5"}=<}
     !{\vunder~{"b8"}{"b7"}{"a3"}{"a2"}<{-1}}
     !{\vcap~{"c3"}{"c3"}{"b1"}{"b8"}=<}
   }
```



At first three polygons are drawn, a triangle with the edges $a1,\ldots,a3$, a polygon with 9 edges $b1,\ldots,b9$ and at last a triangle with edges $c1,\ldots,c3$. Then every knot piece is placed between 4 edges. In the next example the polygons are made visible and the numbering of the edges is shown:

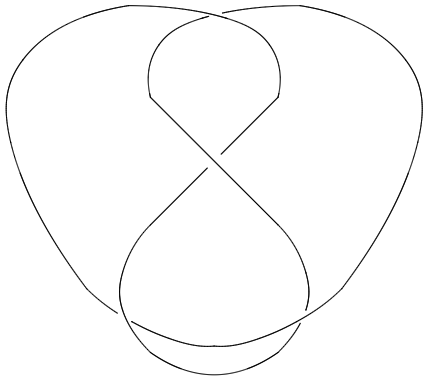
```
\xygraph{
  !{0;/r2.0pc/:}
  !P3"a"{~*{\xypolynode}>{}}
  !P9"b"{~:{(1.3288,0):}~*{\xypolynode}>{}}
  !P3"c"{~:{(2.5,0):}~*{\xypolynode}>{}}
}
```



2.3 The figure-8 knot

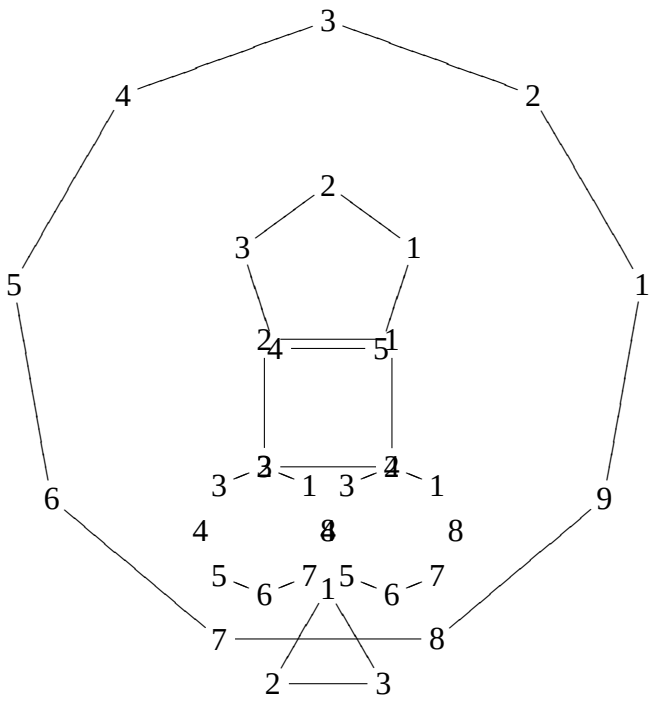
Like the trefoil, the figure-8 knot is drawn using stacked polygons. The arrangement of the polygons is tricky and I cannot give you any general advice.

```
\xygraph{
  !{0;/r2.0pc/:}
  !P9"e"{~:{(5,0):}~>{}}[u]
  !P5"d"{~:{(1.41421,0):}~>{}}[dd]
  !P4"a"{~:{(1.41421,0):}~>{}}[ddl]
  !P8"b"{~={45}~>{}} [rr]
  !P8"c"{~={45}~>{}} [ddl][u(0.1)]
  !P3"f"{~>{}}
  !{\xoverh~{"a2"}{"a1"}{"a3"}{"a4"}}
  !{\vover~{"b6"}{"b4"}{"f1"}{"b2"}}
  !{\vunder~{"c6"}{"c8"}{"f1"}{"c2"}}
  !{\vover~{"d3"}{"d1"}{"a2"}{"a1"}}
  !{\xcapv~{"c6"}{"f2"}{"b6"}{"f3"}}
  !{\hcap~{"d3"}{"e5"}{"b4"}{"e5"}}
  !{\hcap~{"d1"}{"e1"}{"c8"}{"e1"}}
}
```



Here is the arrangement of the polygons:

```
\xygraph{
  !{0;/r2.0pc/:}
  !P9"e"{~:{(5,0):}~*{
   \xypolynode}~>{-}}[u]
  !P5"d"{~:{(1.41421,0):}~*{
   \xypolynode}~>{-}}[dd]
  !P4"a"{~:{(1.41421,0):}~*{
   \xypolynode}~>{-}}[ddl]
  !P8"b"{~={45}~*{
   \xypolynode}~>{-}}[rr]
  !P8"c"{~={45}~*{
   \xypolynode}~>{-}}
  [ddl][u(0.1)]
  !P3"f"{~*{
   \xypolynode}~>{-}}
}
```



3 Troubleshooting

3.1 TEX-capacity exceeded

If there are many XY-pic diagrams on a single page, one may get an LATEX error message about insufficient memory. You have to enlarge main-memory in your `texmf.cnf`. In my SuSE 8.1 distribution the config file is located in `/etc/texmf/texmf.cnf`. Edit this file, look for a constant called `main_memory` and enlarge its value. I use `main_memory = 800000`. Save and run the command `texconfig init`.

A Tables

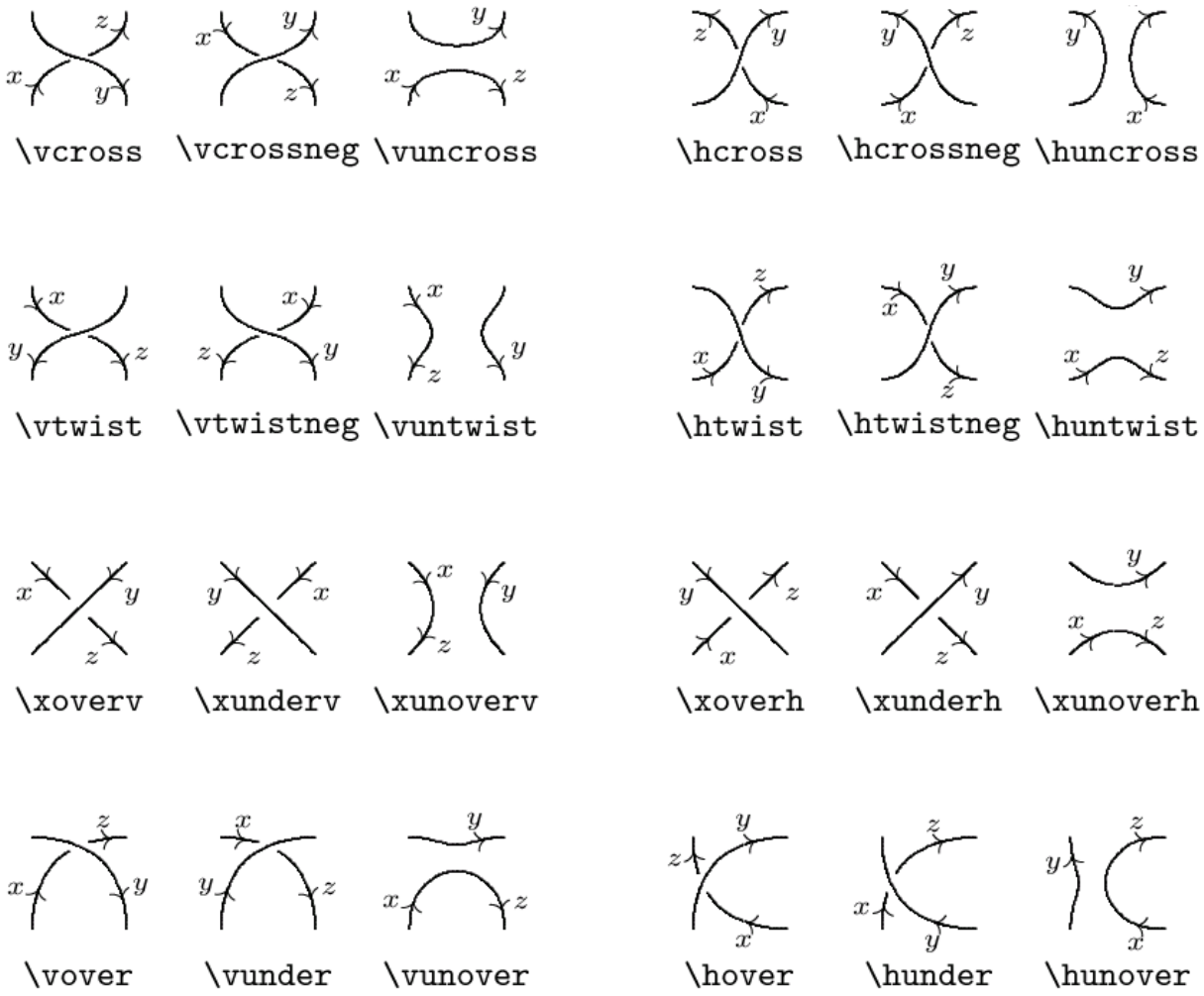


Figure 2: knot crossings with orientation and label positions

The tables shown in figure 2 and 3 are copied from [RM99].

References

[RM99] Kristoffer H. Rose and Ross R. Moore. Xy-pic reference manual, 1999. Available at `http://www.brics.dk/~krisrose/Xy-pic.html`.

[Ros] Kristoffer H. Rose. The Xy-pic homepage. `http://www.brics.dk/~krisrose/Xy-pic.html`.

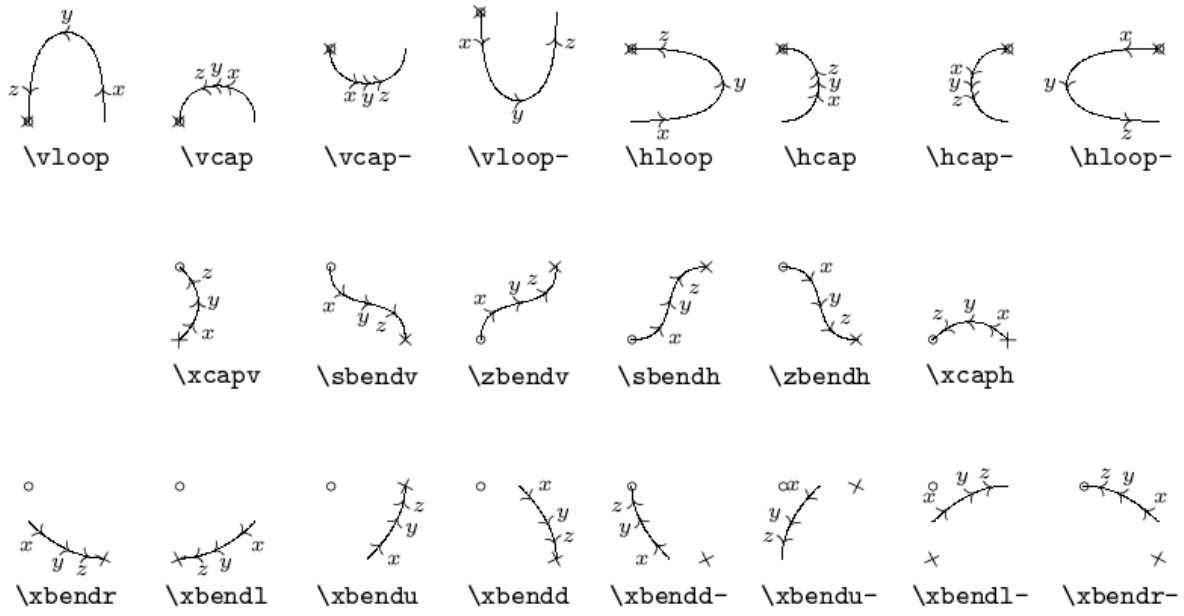


Figure 3: knot joins with orientation and label positions

[Ros99] Kristoffer H. Rose. Xy-pic user's guide, 1999. Available at `http://www.brics.dk/~krisrose/Xy-pic.html`.