

Einige kleine Beispiele mit Erläuterungen zu pstricks

12. März 2013

Inhaltsverzeichnis

1	Polygon	2
2	Kreise und Kreisbögen	3
3	multido	4
4	Beschriftung	5
5	Datenplot mit pst-plot	6
6	Quadratische Funktion	7
7	Grafik über Knoten	8

Vorwort

Abbildungen in Dokumenten nehmen immer eine gesonderte Rolle ein. Es gibt unzählige Möglichkeiten.

Die entsprechende Grafik kann mit einem Zeichenprogramm erstellt werden und durch Export in das erforderliche Grafikformat konvertiert und eingebunden werden. Für pdflatex können die Formate jpeg, pdf und png benutzt werden. Für latex ist das Format eps gut geeignet.

In manchen Fällen ist es ungünstig externe Bilder zu verwalten, z.B. große Sammlung von Übungsaufgaben mit vielen Bildern. In solchen Fällen, ist es wünschenswert die Grafik im latex-Quelltext (intern) zu haben. Auch dafür gibt es mehrere Möglichkeiten.

Einen besonderen Platz nimmt hier PSTricks ein. Durch viele vorhandene Pakete gibt es Vereinfachungen in der Grafikerstellung. Es gibt reichlich Information im Internet, z.B. www.pstricks.de oder das Buch von Herbert Voß „PSTricks Grafik mit PostScript für T_EX und L^AT_EX“. Mit dem Paket pst-pdf kann PSTricks auch mit pdflatex genutzt werden.

Im Folgenden sollen einige kleine Beispiele zeigen, wie einfach die Anwendung von PSTricks sein kann.

Es ist keine vollständige Dokumentation. Diese sollte den angegebenen Quellen und den Dokumentationen zu den Paketen entnommen werden.

1 Polygon

Das Gitter wird mit dem Befehl `\psgrid` in der Größe von `pspicture` (in ganzen Zahlen) erzeugt. Der Standard sieht wie folgt aus (die Label sind auch Standard).

`\pspolygon` zeichnet eine geschlossene Linie.

(für offene Kurven: `\psline`)

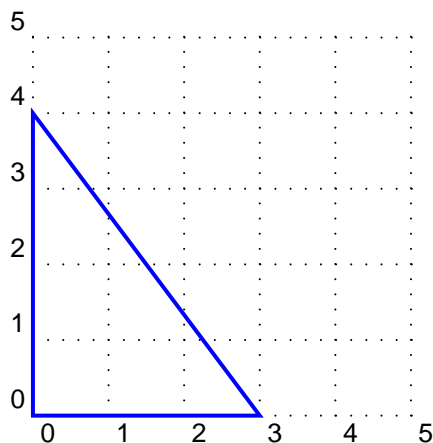
Syntax:

`\pspolygon[Optionen](x1,y1)(x2,y2)...(xn,yn)`

`\pspolygon*[Optionen](x1,y1)(x2,y2)...(xn,yn)`

(Sternversion: das Innere der Kurve wird mit aktueller Linienfarbe gefüllt)

Beispiel:

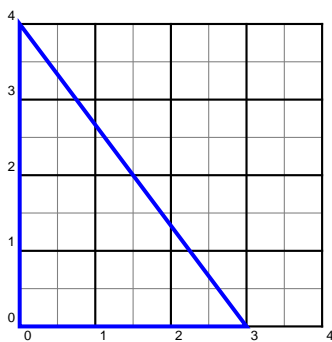


Syntax zu Beispiel Polygon - *benötigt* `\usepackage{pstricks}`

```
\begin{pspicture}(5,5.2)\psgrid
\pstVerb{2 setlinejoin} %Linien laufen rund aus
\pspolygon[linecolor=blue,linewidth=1.5pt](0,4)(3,0)
\end{pspicture}
```

Das Gitter kann über Optionen verändert werden:

Beispiel:



Syntax zu Beispiel Polygon - *benötigt* `\usepackage{pstricks}`

```
\begin{pspicture}(4,4.2)\psgrid[griddots=0,gridlabels=5pt,subgriddiv=2]
\pstVerb{2 setlinejoin}
\pspolygon[linecolor=blue,linewidth=1.5pt](0,4)(3,0)
\end{pspicture}
```

2 Kreise und Kreisbögen

Ein Kreis wird mit dem Befehl `\pscircle` erzeugt. Über Angabe von Optionen können Linienstärke, Farbe, Füllung usw. definiert werden.

Syntax:

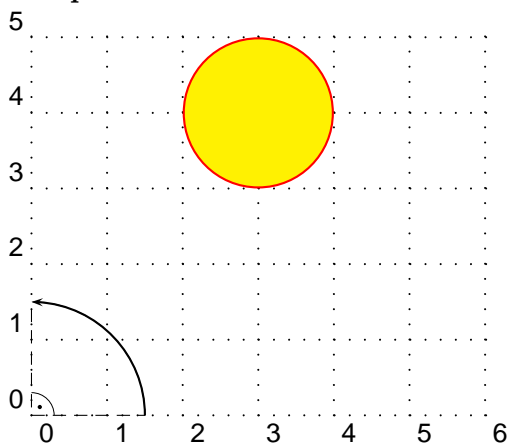
```
\pscircle[Optionen](xM,yM){Radius}  
\pscircle*[Optionen](xM,yM){Radius}  
    (wenn Mittelpunkt nicht angegeben wird, wird automatisch (0,0) genommen)  
    (Sternversion: das Innere des Kreises wird mit aktueller Linienfarbe gefüllt)
```

Mit dem Befehl `\psarc` können Kreisbögen (math. positiv) bzw. Kreisabschnitte gezeichnet werden.

Syntax:

```
\psarc[Optionen]{Pfeil}(xM,yM){Radius}{Winkel1}{Winkel2}  
    (Sternversion: die Fläche wird mit aktueller Linienfarbe gefüllt)
```

Beispiel:



Kreise und Kreisbögen - *benötigt* `\usepackage{pstricks}`

```
\begin{pspicture}(6,5)\psgrid  
\pscircle[linecolor=red,fillcolor=yellow,fillstyle=solid](3,4){1}  
\psarc[showpoints=true]{->}{1.5}{0}{90}  
\psarc[linewidth=0.2pt]{-}{0.3}{0}{90}  
\pscircle* (.11,.11){.8pt} %\psdot(0.15,0.15)  
\end{pspicture}
```

... und `\psdot` ist für einen rechten Winkelpunkt zu groß, daher `\pscircle` da man mit `\pscircle` die Größe selbst definieren kann.

... und `showpoints=true` schaltet die gestrichelten Linien von (xM,yM) des Kreisbogens zu den Endpunkten ein.

3 multido

Mit dem Befehl `\multido` können stetige Wiederholungen per Variable definiert, und somit vermieden werden. Im unteren Beispiel wird mit `r` ein reeller Wert und mit `i` ein ganzer Wert als Variable definiert (weitere Möglichkeiten sind der Anleitung zu `multido` zu entnehmen).

Syntax:

`\multido{Variable=Start+Schrittweite}{Wiederholungen}{Was wiederholt werden soll}`



`multido` - *benötigt* `\usepackage{pstricks}` und `\usepackage{multido}`

```
\begin{pspicture}(0,-.5)(8,1)
\psgrid[subgriddiv=1,gridlabels=0,griddots=0](8,1) %Gitter zeichnen
\multido{\ia=0+2,\ib=1+2}{4}{\psframe*(\ia,0)(\ib,1)} %schwarze Kästen
\multido{\ra=.5+1,\ib=1+1}{8}{\rput(\ra,-.4){\ib}} %Beschriftung
\end{pspicture}
```

4 Beschriftung

Für Bemaßungen wird der Befehl `\psline` verwendet. Durch Angabe der Option kann das Aussehen verändert werden. In der Dokumentation `pstricks-add-doc.pdf` befinden sich die möglichen Styles. Die Angabe der Maßzahl erfolgt dann über `\rput`.

Syntax: Maßpfeil

```
\psline[Optionen]{Style}(x1,y1)...(xn,yn)
```

Syntax: Maßzahl

```
\rput*Drehung(x,y)Maß
```

(Sternversion: ermöglicht Überschreiben, was in diesem Fall erforderlich ist)

In diesem Beispiel ist noch die Anwendung von `\psbezier` gezeigt.

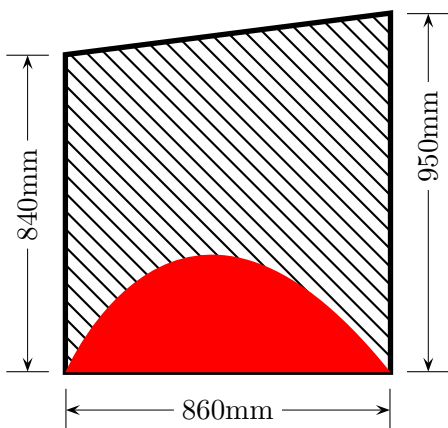
Syntax:

```
\psbezier[Optionen]{Style}(x0,y0)(x1,y1)(x2,y2)(x3,y3)
```

(Sternversion: Fläche wird mit aktueller Linienfarbe gefüllt)

(werden nur 3 Punkte gesetzt, wird standardmäßig der Koordinatenursprung (0,0) gesetzt)

Beispiel:



Beschriftung - *benötigt* `\usepackage{pstricks}`

```
\psset{unit=0.05mm}% 1:20, or =0.1mm for 1:10
\begin{pspicture}(-150,-150)(1000,1000)
  \pspolygon[linewidth=2pt,fillstyle=vlines](0,0)(0,840)(860,950)(860,0)
  \psbezier*[linecolor=red](0,0)(300,700)(860,0)
  \psset{linewidth=0.2pt,arrowscale=2,tbarssize=8pt}
  \psline{|<->|}(0,-100)(860,-100)\rput*(430,-100){860mm}
  \psline{|<->|}(960,0)(960,950)\rput*{90}(960,625){950mm}
  \psline{|<->|}(-100,0)(-100,840)\rput*{90}(-100,470){840mm}
\end{pspicture}
```

Mit `\psset` können Optionen für alle nachfolgenden Befehle gesetzt werden. Zu Beginn von `pspicture` ist der Maßstab gesetzt worden und in der Zeichnung sind mit `psset` die Optionen für die Maßlinien gesetzt worden, in diesem Falle Strichstärke, Pfeilgröße, Länge der Maßhilfslinien.

Wie in diesem Beispiel noch zu sehen ist, muss der Koordinatenursprung nicht bei (0,0) liegen. Es ist zum Zeichnen einfach, die Grafik bei (0,0) beginnen zu lassen und die Bemaßung im Anschluß auszuführen. Dazu kann `\begin{pspicture}(-x0,-y0)(x1,y1)` gesetzt werden, damit die Bemaßung dann innerhalb der Grafik Platz findet.

5 Datenplot mit pst-plot

Pst-plot bietet mehrere Plotfunktionen. In diesem Beispiel wird gezeigt, wie eine fertige Datendatei ausgegeben werden kann. Die Datenstruktur muss folgendermaßen aussehen: x y *oder* x,y *oder* (x,y) *oder* $\{x,y\}$

Syntax:

```
\fileplot[Optionen]{Dateiname}
```

In diesem Beispiel werden auch Koordinatenachsen mit dem Befehl `\psaxes` (mit Paket `pstricks-add`) verwendet.

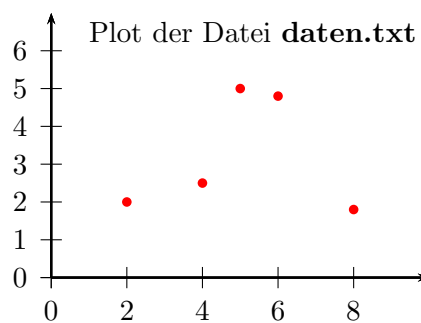
Syntax:

```
\psaxes[Optionen]{Style}(x1,y1)
```

Beispiel:

Inhalt der Datei **daten.txt**:

```
2 2
4 2.5
5 5
6 4.8
8 1.8
```



Datenplot - *benötigt* `\usepackage{pstricks}` und `\usepackage{pst-plot}`

```
\psset{unit=5mm}
\begin{pspicture}(-1.2,-1)(10,6)
\fileplot[linecolor=red,plotstyle=dots]{daten.txt}
\psaxes[Dx=2]{->}(10,7)
\rput[1](1,6.5){Plot der Datei \textbf{daten.txt}}
\end{pspicture}
```

Die möglichen Styles für Label befinden sich in der Dokumentation `pstricks-add-doc.pdf`. Die Option `Dx=2` gibt an, dass auf der x-Achse nur jeder 2. Label gesetzt werden soll.

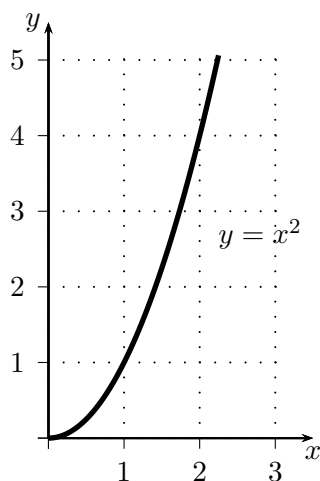
6 Funktion x^2

Bei der Eingabe der Funktion ist die UPN-schreibweise zu verwenden, also erst die Variablen angeben, dann erst wird die mathematische Funktion aufgerufen. Funktionen werden mit dem Befehl `\psplot` erstellt.

Syntax:

`\psplot[Optionen]{Definitionsintervall Startwert}{Definitionsintervall Endwert}{Funktion f(x)}`

Beispiel:



Funktionen - *benötigen* `\usepackage{pstricks}` und `\usepackage{pst-plot}`

```
\psset{unit=1cm}
\begin{pspicture}(-0.5,-0.5)(3.5,5.6)
\psgrid[gridlabels=0pt](0,0)(3,5) %label von psgrid ausgeschaltet, label mit psaxes gesetzt
\psaxes[showorigin=false, tickstyle=bottom, labelsep=5pt]{->}(3.5,5.5)
\rput(3.5,-.2){$x$}
\rput(-.2,5.5){$y$}
\rput(2.8,2.7){$y=x^2$}
\psplot[plotpoints=200,linewidth=2pt]{0}{2.25}{x^2 exp}
\end{pspicture}
```

Verwendete Optionen:

`gridlabels=0pt` Standardlabel von `psgrid` werden ausgeschaltet, da `psaxes` verwendet wird, können die label auch darüber gesetzt werden

`showorigin=false` Nullpunkt wird nicht ausgegeben

7 Grafik über Knoten

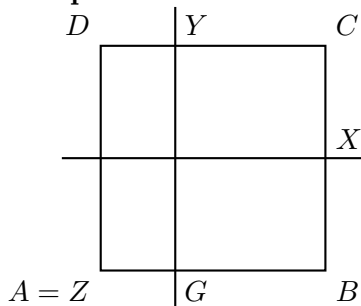
Mit dem Paket `pst-node` können Knoten definiert werden, welche dann beschriftet und beliebig verbunden werden können. `pnode` definiert einen Knoten mit Radius Null. Weitere Knotendefinitionen sind in der Dokumentation zu `pst-node` beschrieben.

Syntax:

`\pnode{Knotenname}` bzw. `\pnode(x,y){Knotenname}`

Die Nutzung der Knoten kann nun über `psframe`, `ncline`, ..., s. Doku erfolgen.

Beispiel:



Beispiel zu Knoten - *benötigt* `\usepackage{pst-node}`

```
\psset{unit=1cm}
\begin{pspicture}(-1.2,-.5)(3.5,3.5)\SpecialCoor
\pnode(0,0){A}\nput{225}{A}{A=Z$}
\pnode(3,0){B}\nput{315}{B}{B$}
\pnode(3,3){C}\nput{45}{C}{C$}
\pnode(0,3){D}\nput{135}{D}{D$}
\psframe(A)(C)
\pnode(1,0){G}\nput{315}{G}{G$}
\pnode(1,3){Y}\nput{45}{Y}{Y$}
\ncline[nodesep=-5mm]{G}{Y} \pnode(0,1.5){P}
\pnode(0,1.5){P}
\pnode(3,1.5){X}\nput{45}{X}{X$}
\ncline[nodesep=-5mm]{P}{X}
\end{pspicture}
```

Verwendete Optionen:

`nodesep=-5mm` die Verbindungslinie wird 5 mm über den Knoten hinaus gezeichnet