

Asymptote in L^AT_EX verwenden

M. Kremer, S. Stock, S. Ulbricht

12.06.2015

Gliederung

- 1 Das Programm
- 2 Konfigurationen am TeXmaker
- 3 Grundbefehle
 - Größe festlegen
 - pair, path
 - draw, pen
 - Label, Arrows
 - nützlich: dot, fill, arc
- 4 Transformationen und Funktionen
 - rotate, shift, scale
 - Funktionen definieren
- 5 Schnittpunkte
- 6 Schleifen



Homepage: <http://asymptote.sourceforge.net/>

Bitte downloaden und installieren!

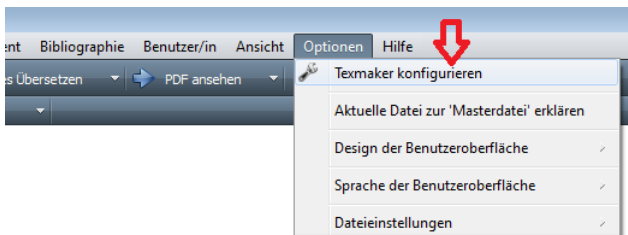
Gliederung

- 1 Das Programm
- 2 Konfigurationen am TeXmaker
- 3 Grundbefehle
 - Größe festlegen
 - pair, path
 - draw, pen
 - Label, Arrows
 - nützlich: dot, fill, arc
- 4 Transformationen und Funktionen
 - rotate, shift, scale
 - Funktionen definieren
- 5 Schnittpunkte
- 6 Schleifen

Neues Package einbinden: `\usepackage{asymptote}`

Quelltext für Asymptote direkt in \LaTeX -Code integrieren:

```
\begin{asy}  
...  
\end{asy}
```



Gliederung

- 1 Das Programm
- 2 Konfigurationen am TeXmaker
- 3 Grundbefehle**
 - Größe festlegen
 - pair, path
 - draw, pen
 - Label, Arrows
 - nützlich: dot, fill, arc
- 4 Transformationen und Funktionen
 - rotate, shift, scale
 - Funktionen definieren
- 5 Schnittpunkte
- 6 Schleifen

Anderer Quellcode als in \LaTeX .

Befehle werden nicht mit `\` begonnen, müssen aber mit `;` beendet werden.

Auskommentiert wird jetzt mit `//`

Gliederung

- 1 Das Programm
- 2 Konfigurationen am TexMaker
- 3 Grundbefehle**
 - Größe festlegen
 - pair, path
 - draw, pen
 - Label, Arrows
 - nützlich: dot, fill, arc
- 4 Transformationen und Funktionen
 - rotate, shift, scale
 - Funktionen definieren
- 5 Schnittpunkte
- 6 Schleifen

Umgebung im \LaTeX -Code öffnen und mit dem `size`-Befehl Größe festlegen:

```
\begin{asy}  
  size(4cm,4cm);  
  ...  
\end{asy}
```

Gliederung

- 1 Das Programm
- 2 Konfigurationen am TeXmaker
- 3 Grundbefehle**
 - Größe festlegen
 - **pair, path**
 - draw, pen
 - Label, Arrows
 - nützlich: dot, fill, arc
- 4 Transformationen und Funktionen
 - rotate, shift, scale
 - Funktionen definieren
- 5 Schnittpunkte
- 6 Schleifen

Definierte Wege von A nach B

```
pair A = (0,1);  
pair B = (2,0);  
path P = A -- B;
```

Die Komponenten von A und B können auch einzeln angewählt werden:

```
path Q = (A.x,B.y) -- A..(B.x,A.y)..B -- cycle;
```

- sind gerade Verbindungslinien
- .. legt einen Spline von A nach B
- cycle schließt den Pfad

Einschub **Datentypen:**

<code>pair</code>	erzeugt einen Punkt in der (x, y) -Ebene
<code>real</code>	reellwertige Größen
<code>int</code>	ganze Zahlen
<code>tripples</code>	Punkte im Raum
<code>string</code>	Zeichenfolge

Gliederung

- 1 Das Programm
- 2 Konfigurationen am TexMaker
- 3 Grundbefehle
 - Größe festlegen
 - pair, path
 - **draw, pen**
 - Label, Arrows
 - nützlich: dot, fill, arc
- 4 Transformationen und Funktionen
 - rotate, shift, scale
 - Funktionen definieren
- 5 Schnittpunkte
- 6 Schleifen

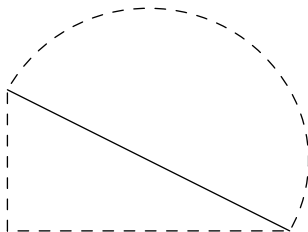
Denkbar Sempel:

```
draw(P);
```

```
currentpen=dashed;  
draw(Q);
```

Zeichnet den Pfad Q gestrichelt. In kurzer Form lassen sich Optionen auch direkt im draw-Befehl einstellen:

```
draw(Q,dashed);
```



```
pair A = (0,1);  
pair B = (2,0);  
path P = A -- B;  
path Q = (A.x,B.y) -- A..(B.x,A.y)..B -- cycle;  
draw(P);  
draw(Q,dashed);
```


Gliederung

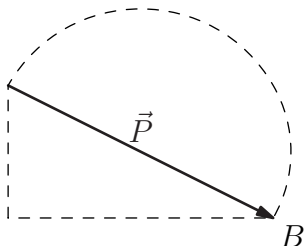
- 1 Das Programm
- 2 Konfigurationen am TeXmaker
- 3 Grundbefehle**
 - Größe festlegen
 - pair, path
 - draw, pen
 - Label, Arrows**
 - nützlich: dot, fill, arc
- 4 Transformationen und Funktionen
 - rotate, shift, scale
 - Funktionen definieren
- 5 Schnittpunkte
- 6 Schleifen

```
currentpen=Black+1;  
draw(Label("$\vec{P}$",MidPoint,N),P,EndArrow(6.0));
```

Die Position des Labels im Pfad lässt sich durch `BeginPoint`, `MidPoint`, `EndPoint` oder kontinuierlich mit `Relative(0.3)` einstellen.

Die Orientierung des Labels wird in Himmelsrichtungen N,S,E,W aber auch NE, SW angegeben.

`EndArrow`, `MidArrow` und `BeginArrow` sind selbsterklärend. `Arrows` setzt Pfeile an Anfang und Ende des Pfades. Die Größe der Pfeile lässt sich ebenfalls einstellen.



```
draw(Label("$\vec{P}$",MidPoint,N),P,EndArrow(6.0));
```

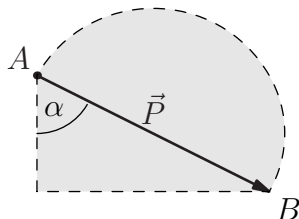
Label lassen sich auch frei im Raum platzieren:

```
label("$B$",B,SE);
```

Gliederung

- 1 Das Programm
- 2 Konfigurationen am TeXmaker
- 3 Grundbefehle**
 - Größe festlegen
 - pair, path
 - draw, pen
 - Label, Arrows
 - **nützlich: dot, fill, arc**
- 4 Transformationen und Funktionen
 - rotate, shift, scale
 - Funktionen definieren
- 5 Schnittpunkte
- 6 Schleifen

```
dot(Label("$A$",MidPoint,NW),A,Black+3);  
fill(Q,lightgray);  
draw(Label(...),arc(A,0.5,-90,-30));
```



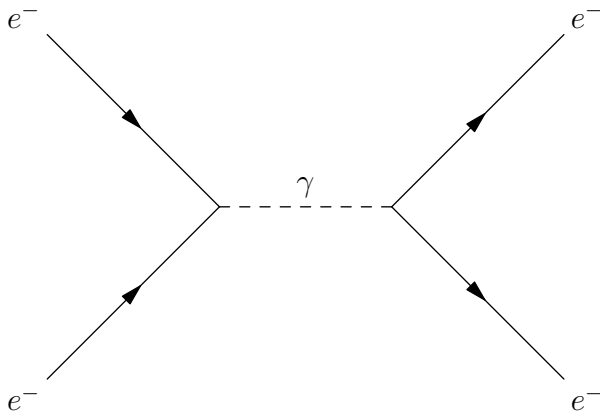


Abbildung 1: Elektron-Elektron-Streuung

Gliederung

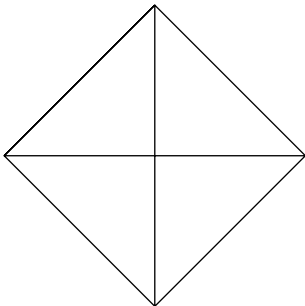
- 1 Das Programm
- 2 Konfigurationen am TeXmaker
- 3 Grundbefehle
 - Größe festlegen
 - pair, path
 - draw, pen
 - Label, Arrows
 - nützlich: dot, fill, arc
- 4 Transformationen und Funktionen
 - rotate, shift, scale
 - Funktionen definieren
- 5 Schnittpunkte
- 6 Schleifen

Gliederung

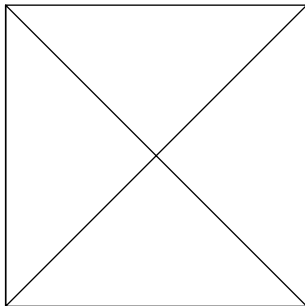
- 1 Das Programm
- 2 Konfigurationen am TeXmaker
- 3 Grundbefehle
 - Größe festlegen
 - pair, path
 - draw, pen
 - Label, Arrows
 - nützlich: dot, fill, arc
- 4 Transformationen und Funktionen
 - rotate, shift, scale
 - Funktionen definieren
- 5 Schnittpunkte
- 6 Schleifen

jetzt unser Spielzeug:

```
path F = S-W-N-S-E-W-N-E;
```



```
draw(F);
```



```
draw(rotate(45)*F);
```

```
rotate(real);
```

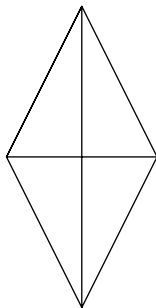
```
rotate(pair,real);
```

```
xscale(real);
```

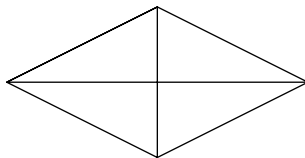
```
yscale(real);
```

```
scale(real,real);
```

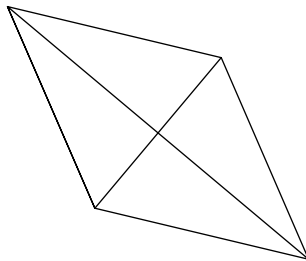
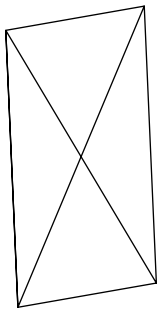
```
scale(pair);
```



```
draw(xscale(0.5)*F);
```



```
draw(yscale(0.5)*F);
```



```
draw(xscale(0.5)*rotate(50)*F); draw(rotate(50)*xscale(0.5)*F);
```

Gliederung

- 1 Das Programm
- 2 Konfigurationen am TeXmaker
- 3 Grundbefehle
 - Größe festlegen
 - pair, path
 - draw, pen
 - Label, Arrows
 - nützlich: dot, fill, arc
- 4 Transformationen und Funktionen
 - rotate, shift, scale
 - Funktionen definieren
- 5 Schnittpunkte
- 6 Schleifen

Zum Definieren von Funktionsverläufen als `path` muss vorher die Funktion

```
import graph;
```

übergeben werden.

```
real f(real x){  
return 2*x-x^2+3*x^3;  
}
```

```
path g = graph(f,0,1);  
draw(g);
```

Beispiel:

```
real SincQuadrat(real x){  
  return 10*(sin(x))^2/x^2;  
}
```

```
draw(graph(SincQuadrat,-10,-0.001));  
draw(graph(SincQuadrat, 0.001, 10));
```

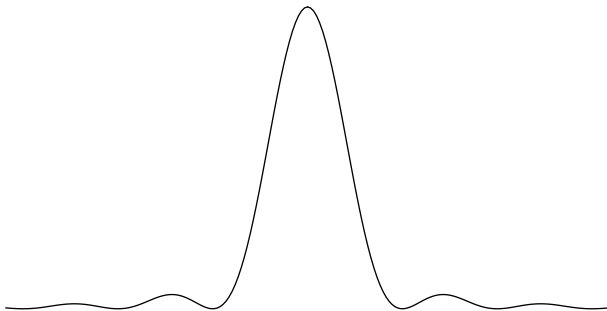
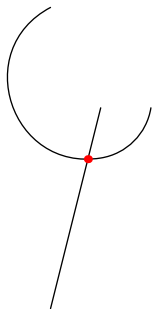


Abbildung 2: Intensitätsverteilung

Gliederung

- 1 Das Programm
- 2 Konfigurationen am TeXmaker
- 3 Grundbefehle
 - Größe festlegen
 - pair, path
 - draw, pen
 - Label, Arrows
 - nützlich: dot, fill, arc
- 4 Transformationen und Funktionen
 - rotate, shift, scale
 - Funktionen definieren
- 5 Schnittpunkte
- 6 Schleifen


```
path A = (0,0)-(1,4);  
path B = (2,4)..(1,3)..(0,6);  
  
draw(A);  
draw(B);  
  
pair C = intersectionpoint(A,B);  
  
dot(C,red);
```



Gliederung

- 1 Das Programm
- 2 Konfigurationen am TeXmaker
- 3 Grundbefehle
 - Größe festlegen
 - pair, path
 - draw, pen
 - Label, Arrows
 - nützlich: dot, fill, arc
- 4 Transformationen und Funktionen
 - rotate, shift, scale
 - Funktionen definieren
- 5 Schnittpunkte
- 6 Schleifen

```
int k = 42;

for (int i=0; i<k; ++i) {
    int j= i+1;
    [tu irgendwas mit i];
}
```