

## 4 Festigkeitsberechnungen

### 4.1 Darstellung eines zusammengesetzten Biegeträgers

Leider sind die grafischen Darstellungsmöglichkeiten unter VBA sehr begrenzt. Eine Möglichkeit graphische Elemente darzustellen sind *Shapes*.

Mit der Anweisung

```
(Ausdruck).AddShape(Type, Left, Top, Width, Height)
```

und dem Type *msoShapeRectangle* lassen sich Rechtecke auf dem Arbeitsblatt platzieren.

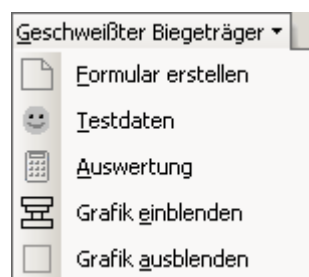
So erzeugt die Anweisung

```
Shapes.AddShape msoShapeRectangle, 200 , 100, 80, 60
```

ausgehend von der linken oberen Ecke, 200 Punkte nach rechts und 100 Punkte von oben, ein Rechteck mit der Breite von 80 Punkten und der Höhe von 60 Punkten.

Mit der *Visual Basic-Hilfe* finden Sie unter dem Index *Addshape* einen Hilfstext, in dem der Link *msoAutoShapeType* eine Vielzahl möglicher Shapeformen anbietet.

Zunächst wird die Symbolleiste um die beiden Menüpunkte Grafik einblenden und Grafik ausblenden ergänzt.



**Bild 4-1**  
Ergänzungen in der  
Symbolleiste

Das Tabellenblatt *tblTräger* erhält die nachfolgenden Prozeduren.

#### **Codeliste 4-1** Ergänzung der Prozeduren in *tblTräger*

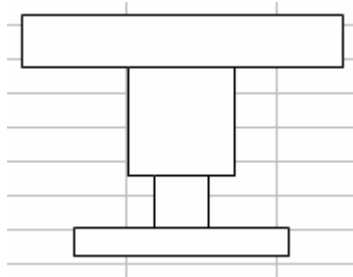
```
Sub Grafik_ein()
    Dim Zeile As Integer
    Dim x, b, h As Double
    Zeile = 3
    x = 200
    Do While Cells(Zeile, 3) > 0
```

```

    b = Cells(Zeile, 3) / 5
    h = Cells(Zeile, 4) / 5
    'Parameter: Type, Left, Top, Width, Height
    Shapes.AddShape msoShapeRectangle, 200 - (b / 2), x, b, h
    x = x + h
    Zeile = Zeile + 1
  Loop
End Sub
Sub Grafik_aus()
  Dim Shp As Shape
  For Each Shp In Shapes
    Shp.Delete
  Next
End Sub

```

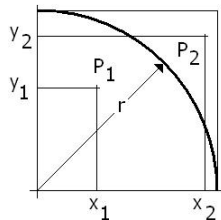
Für das im Buch dargestellte Trägerbeispiel *Laufschiene* ergibt sich die in Bild 4-2 dargestellte Form.



**Bild 4-2**  
Trägerquerschnitt des  
Berechnungsbeispiels

## 4.2 Berechnung eines Viertelkreises nach der Monte-Carlo-Methode

Benutzt man zwei Intervalle mit einer Gleichverteilung, dann erhalten die damit erzeugten Punkte in einer Ebene ebenfalls eine Gleichverteilung. Diese Eigenschaft nutzt diese Methode zur Flächenbestimmung.



**Bild 4-3**  
Flächenbestimmung nach der  
Monte-Carlo-Methode

In Bild 4-3 liegt der Punkt  $P_1$  in der gesuchten Fläche (Treffer), der Punkt  $P_2$  jedoch außerhalb.

Einen Punkt  $P_i$  erzeugt man durch zwei Zufallszahlen  $x_i$  und  $y_i$ . Ob  $P_i$  ein Treffer ist, ergibt sich aus der Formel des Pythagoras mit

$$\sqrt{x_i^2 + y_i^2} \leq r. \quad (4.1)$$

Setzt man  $r=1$ , dann können die im Intervall  $(0,1]$  erzeugten Zufallszahlen direkt als Koordinaten eines Punktes in der Ebene verwendet werden.

Erzeugt man hinreichend viele Punkte  $n$  und hat damit  $m$  Treffer ( $m < n$ ), und sind  $A_K$  der Flächeninhalt des Viertelkreises und  $A_Q$  der Flächeninhalt des Quadrats, dann gilt das Verhältnis

$$\frac{m}{n} = \frac{A_K}{A_Q}. \quad (4.2)$$

Und damit

$$A_K = \frac{m}{n} \cdot A_Q. \quad (4.3)$$

Die quadratische Fläche wird 1 gesetzt und damit bestimmt sich der exakte Wert des Viertelkreises aus dem Funktionswert  $\text{Atn}(1)$ .

**Tabelle 4.5** Struktogramm Monto-Carlo-Methode am Biegeträger

Eingabe n	
Randomize, m=0	
i=1, 1, n	
x = Rnd(x)	
y = Rnd(x)	
$r = \sqrt{x_i^2 + y_i^2}$	
Ist r < 1	
Ja	Nein
m=m+1	
Ausgabe der Fläche $A_k(n) = m/n$	

**Codeliste 4-2** Bestimmung der Fläche eines Viertelkreises nach der Monto-Carlo-Methode

```

Sub MonteCarlo()
  Dim i, j, m, n As Long
  Dim x, y, r As Double

  Randomize: x = 0

```

```
j = ActiveCell.Row
n = Cells(j, 1)
m = 0
For i = 1 To n
  x = Rnd(x)
  y = Rnd(x)
  If Sqr(x * x + y * y) < 1 Then
    m = m + 1
  End If
Next i
Cells(j, 2) = m / n
Cells(1, 3) = Atn(1)
End Sub
```

Bei dieser Methode kann der Fall auftreten, dass bei größerer Anzahl von Versuchen der Näherungswert wieder ungenauer wird. Jedoch bei hinreichender großer Anzahl von Versuchen wird der Näherungswert immer genauer.

	A	B	C
1	1000	0,781	0,78539816
2	5000	0,7882	
3	10000	0,7848	
4	50000	0,7849	
5	100000	0,78797	
6	500000	0,784404	
7	1000000	0,784612	
8	5000000	0,785494	

**Bild 4-4**  
Auswertung mit  
unterschiedlichen  
Laufvariablen