

Graphische Darstellungen im Analysisunterricht

Viele Konzepte der Analysis haben eine graphische Interpretation, die sich allerdings im konkreten Fall naturgemäß nur unter großem Aufwand auf Papier bringen läßt.

Wir zeigen in dieser Arbeit, wie das Computeralgebrasystem DERIVE [5] dazu verwendet werden kann, Konzepte der Analysis graphisch zu veranschaulichen. Dies kann wesentlich zum Verständnis der Unterrichtsinhalte beitragen.

1 Iteration und Fixpunkte

Jeder Schüler hat vermutlich schon folgende Situation erlebt: Man nehme einen Taschenrechner zur Hand und drücke ununterbrochen auf die COS-Taste. Man stellt dann fest, daß eine Zahlenkolonne berechnet wird, deren Ergebnis sich schließlich nicht mehr ändert.

Mit DERIVE kann man diese *Iteration* mit dem ITERATES-Befehl durchführen. Beim Aufruf von ITERATES(f, x, x_0, n) wird beginnend bei $x = x_0$ die Vorschrift $x_{k+1} = f(x_k)$ n -mal durchgeführt. Die Anwendung von `Simplify` auf ITERATES(COS(x), x , 0, 5) liefert also

$$[0, 1, \text{COS}(1), \text{COS}(\text{COS}(1)), \text{COS}(\text{COS}(\text{COS}(1))), \text{COS}(\text{COS}(\text{COS}(\text{COS}(1))))],$$

da mit `Simplify` immer exakte Rechnungen durchgeführt werden. Möchte man mit dezimalen Näherungswerten arbeiten, wodurch Konvergenz gegebenenfalls erst sichtbar wird, verwendet man `approx` und erhält

$$[0, 1, 0.540302, 0.857553, 0.654289, 0.79348].$$

Noch sieht man die Konvergenz nicht. Läßt man jedoch das vierte Argument von ITERATES weg, so wird die Iteration so lange durchgeführt, bis eine Zahl zum wiederholten Male als Iterationswert berechnet wird. Als Resultat von ITERATES(COS(x), x , 0) unter Verwendung von `approx` erhält man nun

$$[0, 1, 0.540302, 0.857553, \dots, 0.739085, 0.739084, 0.739085, 0.739084, 0.739085].$$

Man bedenke aber, daß `Simplify` hier in eine Endlosschleife gerät und nicht abbricht (warum?)!

Das Ergebnis von eben sieht so aus, als wäre das Verfahren zu spät abgebrochen worden. DERIVE arbeitet intern immer mit rationalen Näherungen. Wenden wir `Simplify` auf das eben berechnete Ergebnis an, erhalten wir¹

$$\left[0, 1, \frac{5396}{9987}, \frac{8362}{9751}, \dots, \frac{7838}{10605}, \frac{4943}{6688}, \frac{34111}{46153}, \frac{7770}{10513}, \frac{34111}{46153}\right],$$

und man sieht, daß das letzte Resultat tatsächlich das erste ist, das bereits in der Iterationsliste vorkam.

¹Dieses Ergebnis ist möglicherweise versionsabhängig.

Wir können auch schwierigere Iterationen durchführen, die man mit einem Taschenrechner nicht mehr so leicht nachvollziehen kann. Beispielsweise liefert die Approximation des Ausdrucks $\text{ITERATES}(\text{EXP}(-x), x, 0)$ die Liste

$[0, 1, 0.367879, 0.6922, \dots, 0.567143, 0.567143, 0.567143, 0.567143, 0.567143]$

(mit welchen Tastenkombinationen könnte man diese Liste mit einem Taschenrechner erzeugen?), und $\text{ITERATES}(x+\text{SIN}(x), x, 1)$ generiert eine Approximation der Kreiszahl π

$[1, 1.84147, 2.80506, 3.13527, 3.14159, 3.14159]$.

Mit **Options Precision Digits** können wir im übrigen bei größerer Rechenzeit auch mit größerer Rechengenauigkeit arbeiten. Man verwende dann gegebenenfalls als Startwert eine bereits berechnete gute Näherung!

Die gezeigten Berechnungen lassen sich nun zwar auch ohne weiteres mit einem programmierbaren Taschenrechner durchführen, aber sowohl die Frage, warum beispielsweise das letzte Resultat eine Näherung von π ist, als auch die Frage, warum die Konvergenz bei diesem Beispiel so gut ist, bleibt im Dunkeln.

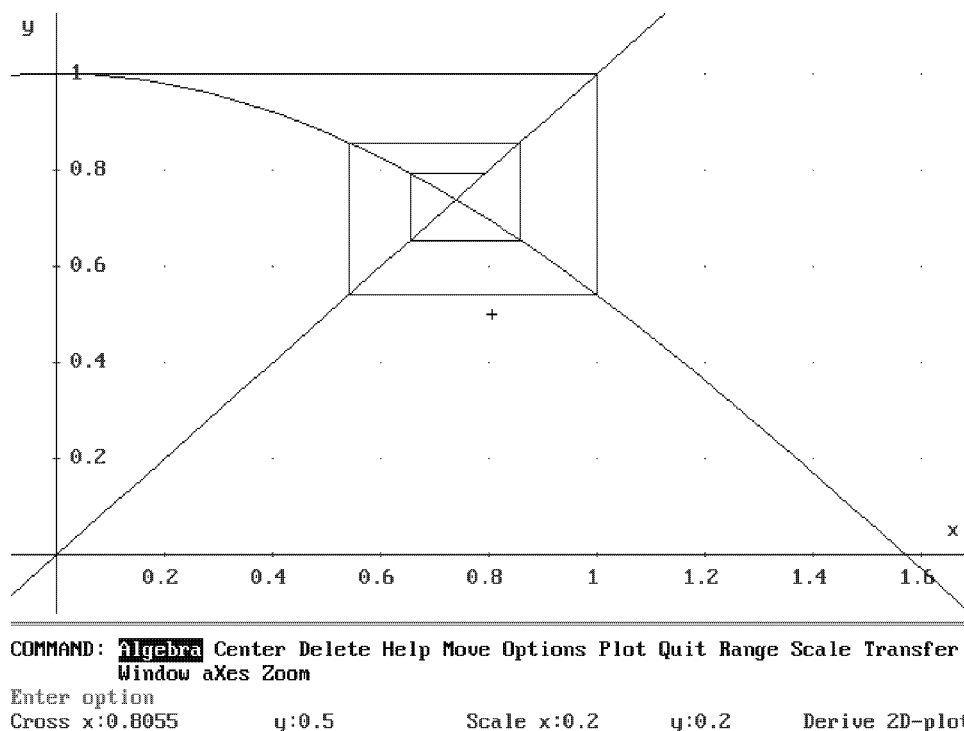


Abbildung 1: Graphische Darstellung von $\text{FIXPUNKT_GRAPH}(\text{COS}(x), x, 0, 5)$

Hier hilft nun eine graphische Darstellung der Situation weiter, die wir mit DERIVE erstellen können. Hierzu erklären wir die DERIVE-Funktionen²

²Der Schüler lade diese Funktionen mit **Transfer Load Utility** aus einer vom Lehrer präparierten Datei (zunächst) als Black Box!

```
FIXPUNKT_FOLGE1(f,x,x0,n):=ITERATES(f,x,x0,n)
```

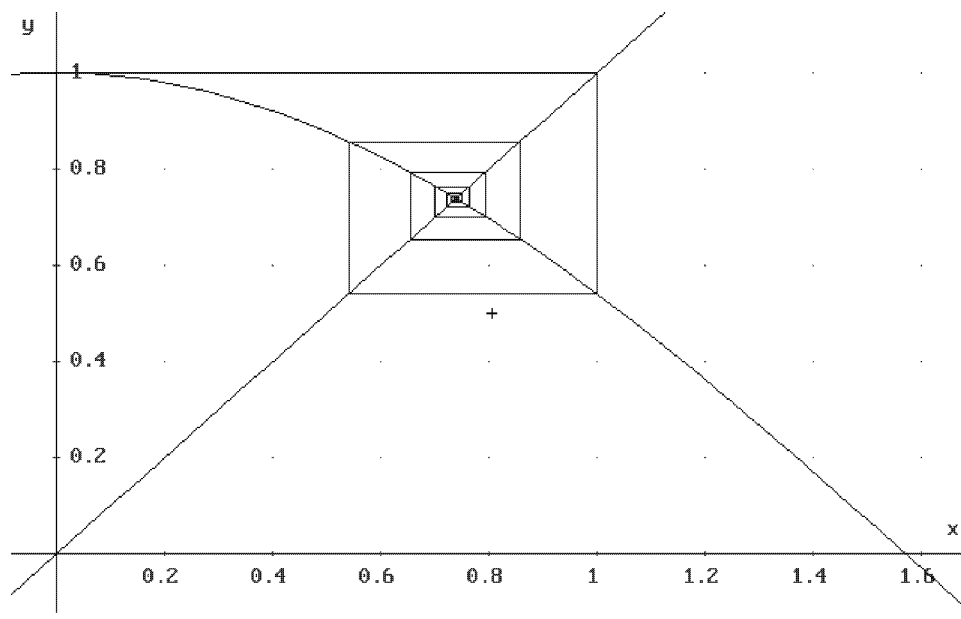
```
FIXPUNKT_FOLGE2(f,x,x0,n,aux):=VECTOR(IF(MOD(1_,2)=0,
[ELEMENT(aux,(1_+2)/2),ELEMENT(aux,(1_+2)/2)],
[ELEMENT(aux,(1_+1)/2),ELEMENT(aux,(1_+3)/2)]),1_,0,2*DIMENSION(aux)-2)
```

```
FIXPUNKT_FOLGE(f,x,x0,n):=FIXPUNKT_FOLGE2(f,x,x0,n,FIXPUNKT_FOLGE1(f,x,x0,n))
```

```
FIXPUNKT_GRAPH_AUX(f,x,x0,aux):=[x,f,aux]
```

```
FIXPUNKT_GRAPH(f,x,x0,n):=FIXPUNKT_GRAPH_AUX(f,x,x0,FIXPUNKT_FOLGE(f,x,x0,n))
```

Die Funktion `FIXPUNKT_GRAPH(f,x,x0,n)` berechnet dann die nötigen Informationen für eine graphische Darstellung der Iteration. Läßt man das letzte Argument weg, wird solange iteriert, bis ein Fixpunkt erzeugt wird, genauer, bis ein Ergebnis berechnet wird, das bereits als Zwischenergebnis vorkam. Das Ergebnis von `FIXPUNKT_GRAPH(f,x,x0,n)` ist ein Vektor, dessen erstes und zweites Element die Ausdrücke `x` und `f` sind, welche ja geplottet werden sollen, und dessen dritte Komponente ein Vektor ist, dessen Zeilen zu zeichnende Streckensegmente darstellen. Man mache sich die Wirkungsweise der erklärten Funktionen klar!



```
COMMAND: Algebra Center Delete Help Move Options Plot Quit Range Scale Transfer
Window axes Zoom
Enter option
Cross x:0.8055      y:0.5      Scale x:0.2      y:0.2      Derive 2D-plot
```

Abbildung 2: Graphische Darstellung von `FIXPUNKT_GRAPH(COS(x),x,0)`

Approximieren wir nun beispielsweise $\text{FIXPUNKT_GRAPH}(\cos(x), x, 0, 5)$, so erhalten wir

$$\left[\begin{array}{c} x, \cos(x), \\ \left[\begin{array}{cc} 0 & 0 \\ 0 & 1 \\ 1 & 1 \\ 1 & 0.540302 \\ 0.540302 & 0.540302 \\ 0.540302 & 0.857553 \\ 0.857553 & 0.857553 \\ 0.857553 & 0.654289 \\ 0.654289 & 0.654289 \\ 0.654289 & 0.79348 \\ 0.79348 & 0.79348 \end{array} \right] \end{array} \right],$$

und `Plot` mit `Options State Connected`³ liefert Abbildung 1. Noch schöner ist natürlich die Darstellung von $\text{FIXPUNKT_GRAPH}(\cos(x), x, 0)$, s. Abbildung 2.

Die Berechnung von $\text{FIXPUNKT_GRAPH}(x+\sin(x), x, 1)$ resultiert in Abbildung 3 .

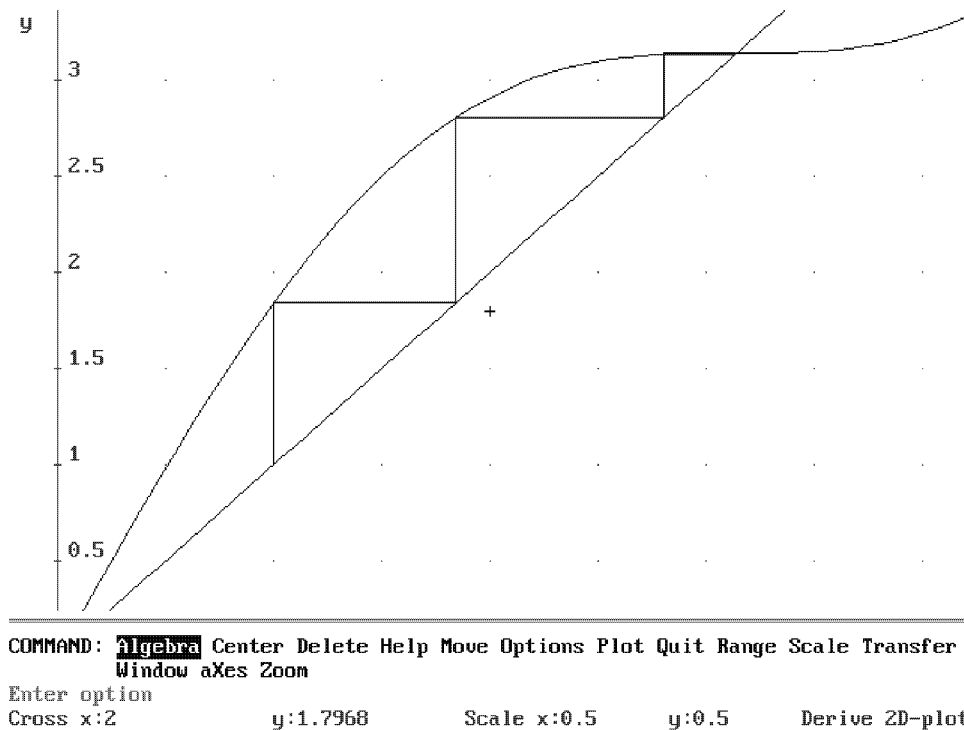


Abbildung 3: Graphische Darstellung von $\text{FIXPUNKT_GRAPH}(x+\sin(x), x, 1)$

Auf diese Weise steht dem Schüler die graphische Situation für jedes konkrete Beispiel in wenigen Sekunden zur Verfügung. Dies kann ganz wesentlich zum Verständnis der zugrundegelegten mathematischen Konzepte beitragen.

³Dies sollte die Standardeinstellung und mit `Transfer Save State` in der Datei `DERIVE.INI` gesichert sein, um den Schülern zu ersparen, diese Einstellung selbst vornehmen zu müssen.

Dadurch ändert sich auch die Unterrichtsmethodik gewaltig: Während bisher graphische Darstellungen – beispielsweise bei der Kurvendiskussion – am Ende eines Rechenprozesses standen und dessen Ziel darstellten oder – wie bei der Iteration – nur in Lehrbüchern zu finden waren, können diese nun am Anfang des Lernprozesses stehen und auf einfache Weise auf viele konkreten Probleme angewandt werden. Was der Schüler auf den erzeugten Graphiken erkennt, kann dann versucht werden zu beweisen. Der Erkenntnisprozeß aber findet beim Schüler selber statt.

Unter Zuhilfenahme der erzeugten Bilder können unsere oben gestellten Fragen z. B. leicht beantwortet werden: Konvergiert das Iterationsverfahren, so konvergiert es offenbar gegen eine Lösung der Gleichung $x = f(x)$, also gegen einen Schnittpunkt des Graphen von f mit der ersten Winkelhalbierenden, und dies offenbar umso schneller, je flacher der Graph am Schnittpunkt ist. Die Funktion $f(x) = x + \sin x$ hat an der Stelle π eine horizontale Tangente, was offenbar optimal ist.

2 Das Newtonverfahren

Das Newtonverfahren baut letztlich auf der eben gewonnenen Erkenntnis auf. Es liefert ein Iterationsverfahren für die numerische Lösung der Gleichung

$$f(x) = 0 ,$$

das fast immer die angesprochene optimale Eigenschaft besitzt. Newtons Idee besteht darin, eine Folge von Näherungswerten dadurch zu berechnen, daß man ausgehend von einer Näherung x_k eine (hoffentlich bessere) Näherung x_{k+1} durch Auflösen der *linearisierten* Hilfs-gleichung

$$y = f(x_k) + (x - x_k)f'(x_k) = 0$$

nach x , also

$$x_{k+1} = x_k - \frac{f(x_k)}{f'(x_k)} \quad (1)$$

erhält. Linearisierung bedeutet somit, daß man statt der Gleichung von f die Tangentengleichung im Punkt $(x_k, f(x_k))$ verwendet. Geometrisch heißt dies dann, daß man den Graphen von f durch die Tangente durch den Punkt $(x_k, f(x_k))$ ersetzt.

Mit den DERIVE-Funktionen

```
NEWTONS(f, x, x0, n) := ITERATES(x-f/DIF(f, x), x, x0, n)
```

```
NEWTON_AUX(f, x, aux, n) := VECTOR(IF(MOD(1_, 2)=0,
[ELEMENT(aux, (1_+2)/2), 0],
[ELEMENT(aux, (1_+1)/2), LIM(f, x, ELEMENT(aux, (1_+1)/2))]), 1_, 0, 2*DIMENSION(aux)-2)
```

```
NEWTON_GRAPH(f, x, x0, n) := [f, NEWTON_AUX(f, x, NEWTONS(f, x, x0, n), n)]
```

implementiere ich das Newtonverfahren mitsamt seiner graphischen Bedeutung. Hierbei berechnet `NEWTONS(f, x, x0, n)` eine n -fache Iteration des Newtonverfahrens, und `NEWTON_GRAPH(f, x, x0, n)` liefert wieder einen zugehörigen Vektor von Streckensegmenten.

Approximieren wir beispielsweise $\text{NEWTON_GRAPH}(\text{COS}(x), x, 5/2, 2)$, so erhalten wir

$$\left[\begin{array}{c} \text{COS}(x), \\ \left[\begin{array}{cc} 2.5 & 0 \\ 2.5 & -0.801143 \\ 1.16135 & 0 \\ 1.16135 & 0.398099 \\ 1.59532 & 0 \end{array} \right] \end{array} \right],$$

und `Plot` liefert Abbildung 4. Die rasante Konvergenzgeschwindigkeit ist graphisch sehr gut zu erkennen. Schon nach 2 Iterationsschritten hat man die Nullstelle gut approximiert und nach einem weiteren Iterationsschritt kann der iterierte Wert mit dem Auge von der Nullstelle nicht mehr unterschieden werden.

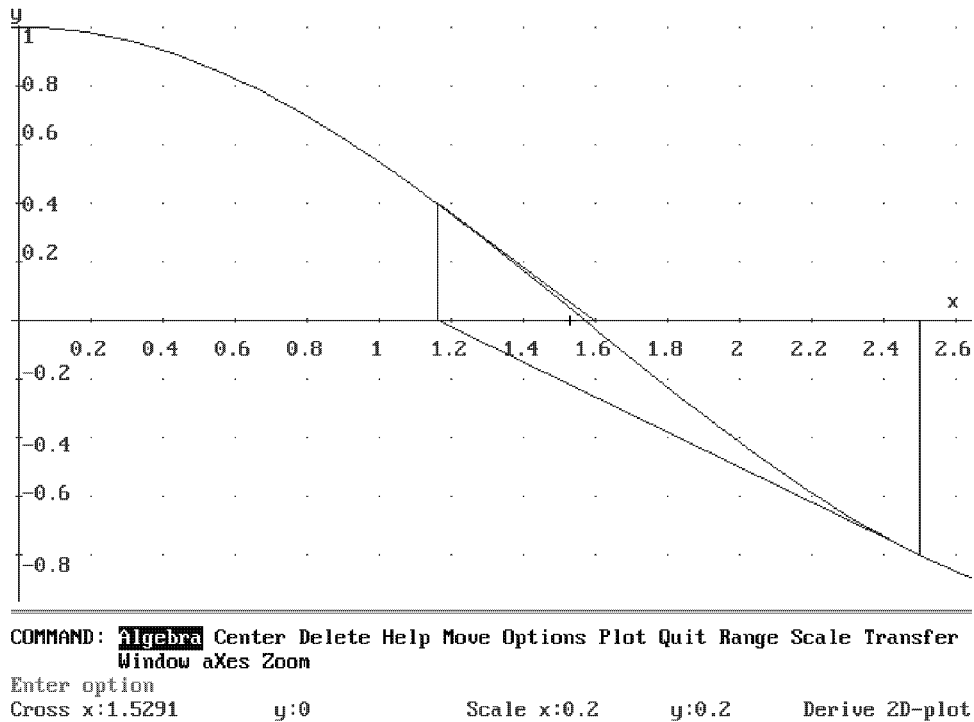


Abbildung 4: Graphische Darstellung von $\text{NEWTON_GRAPH}(\text{COS}(x), x, 5/2, 2)$

3 Divergente und chaotische Iteration

Aber das Newtonverfahren kann auch divergieren. Dies sieht man schön an der Beispielfunktion $f(x) = x^{1/3}$, deren Nullstelle $x_0 = 0$ man mit dem Newtonverfahren wegen $f'(x_0) = \infty$ nicht erzeugen kann. Die graphische Darstellung von $\text{NEWTON_GRAPH}(x^{1/3}, x, 0.1, 5)$ (man verwende `Manage Branch Real`!⁴ Warum?) ergibt Abbildung 5 und verdeutlicht diesen Sachverhalt.

⁴Wieder sollte dies als Standardeinstellung in der Datei `DERIVE.INI` zur Verfügung stehen.

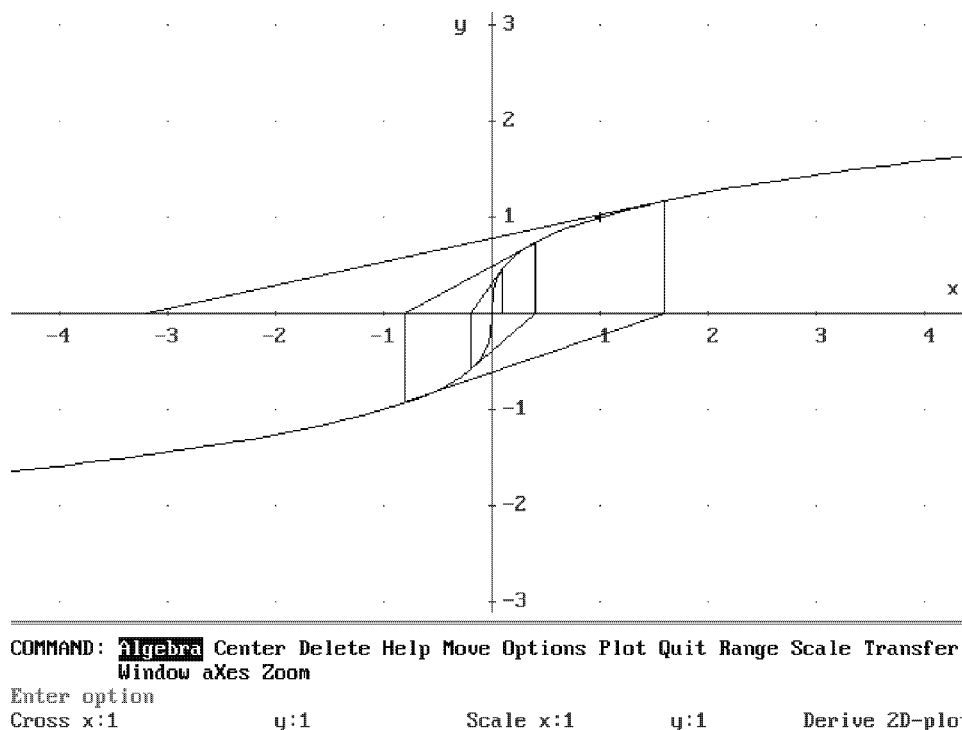


Abbildung 5: Graphische Darstellung von $\text{NEWTON_GRAPH}(x^{1/3}, x, 0.1, 5)$

Nachdem die Schüler die Divergenz aus der Graphik abgelesen haben, ist es ein Leichtes für sie, die Newtonfolge (1)

$$x_{k+1} = x_k - \frac{f(x_k)}{f'(x_k)} = x_k - \frac{x_k^{1/3}}{\frac{3}{3}x_k^{-2/3}} = -2x_k$$

zu berechnen. Die Berechnung zeigt, daß sich der Abstand zur Nullstelle $x_0 = 0$ bei jedem Iterationsschritt verdoppelt. Ein ähnliches Beispiel kann herangezogen werden, um zu zeigen, daß das Newtonverfahren nicht immer schnell konvergieren muß. Man betrachte $f(x) = x^{2/3}$, wo sich $x_{k+1} = -x_k/2$ ergibt.

Die DERIVE-Funktion $\text{NEWTONFOLGE}(f, x) := x - f/\text{DIF}(f, x)$ berechnet die rechte Seite der Formel (1) des Newtonverfahrens. Um diese Berechnung automatisch durchführen zu lassen, kommen uns also die symbolischen Fähigkeiten von DERIVE zur Hilfe. Beispielsweise liefert die Vereinfachung von $\text{NEWTONFOLGE}(x^{1/3}, x)$ wieder $-2x$. Es zeigt sich auch, daß die sonst in der Analysis so berühmte Funktion $f(x) = e^{-1/x^2}$ (s. [3], S. 172 und S. 355) langsam (nämlich linear) gegen ihre „Nullstelle“ $x_0 = 0$ konvergiert. Hier ergibt sich $\text{NEWTONFOLGE}(\text{EXP}(-1/x^2), x)$ zu $x - x^3/2$.

Man kann in diesem Zusammenhang gegebenenfalls den Spieß auch umdrehen und fragen, für welche Funktionen f das Newtonverfahren divergiert bzw. nur langsam konvergiert. Genauer fragen wir: Für welche Funktion f ergibt sich

$$x - \frac{f(x)}{f'(x)} = g(x) ? \tag{2}$$

Aus (2) erhält man durch Integration

$$\frac{f'(x)}{f(x)} = \frac{1}{x - g(x)} \quad \text{oder} \quad \ln f(x) = \int \frac{1}{x - g(x)} dx ,$$

und schließlich

$$f(x) = \exp \left(\int \frac{1}{x - g(x)} dx \right) . \quad (3)$$

Diese Berechnung wird von der DERIVE-Funktion

`NEWTONFUNKTION(g, x) := EXP(INT(1/(x-g), x))`

erledigt, und `NEWTONFUNKTION(-2x, x)` bzw. `NEWTONFUNKTION(-x/2, x)` liefern uns die Funktionen $x^{1/3}$ bzw. $x^{2/3}$ wieder. Deklariert man mittels `F(x) :=` und `G(x) :=` die beiden Funktionen $f(x)$ und $g(x)$, so kann man die obige Herleitung von (3) aus (2) mit DERIVE durchführen: Man gibt die Gleichung `x-F(x)/DIF(F(x), x)=G(x)` ein, zieht x ab (`<F4>-x`), multipliziert mit -1 (`<-F4>`), integriert (`INT(<F4>, x)`, `Simplify`), und erhält das Ergebnis (3) durch Anwendung der Exponentialfunktion (`EXP <F4>`, `Simplify`).

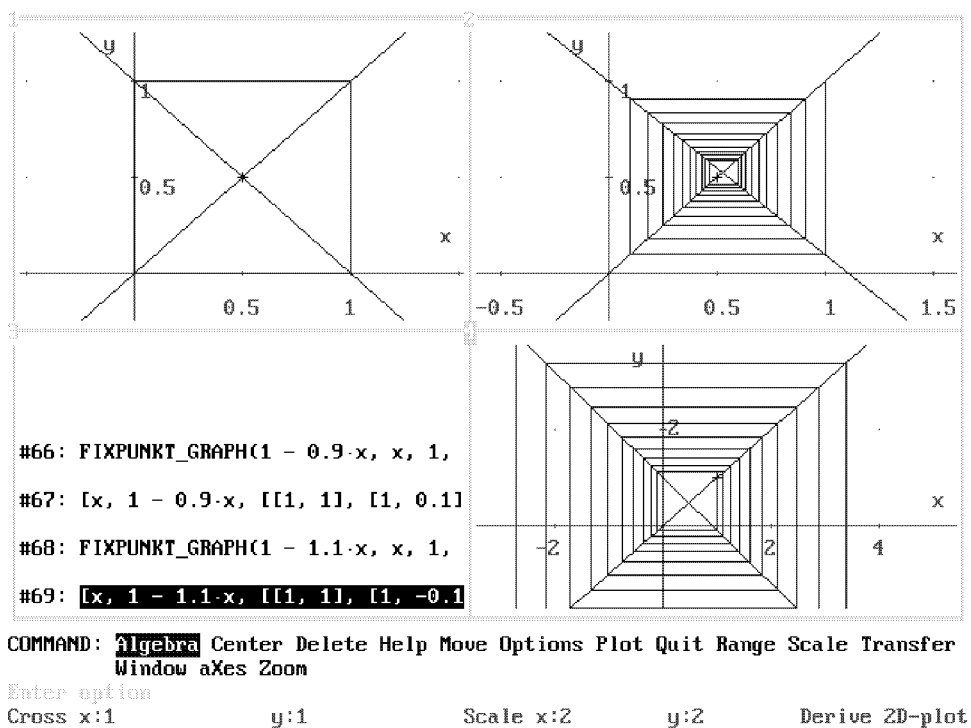


Abbildung 6: Iteration für $f(x) = 1 - x, 1 - 0.9x, 1 - 1.01x$

Erst recht konvergiert das Fixpunktverfahren nicht immer. Unter welchen Voraussetzungen aber liegt Konvergenz vor? Wir hatten bereits erkannt, daß es besonders gut konvergiert, wenn die Funktion f an der betreffenden Stelle x_0 eine horizontale Tangente besitzt, also $f'(x_0) = 0$ gilt. Man sieht nun auch leicht graphisch ein, daß im Fall $|f'(x_0)| < 1$ (lokal) immer Konvergenz vorliegt, während wir im Fall $|f'(x_0)| > 1$ Divergenz haben. Besonders interessant sind also Fälle, bei denen entweder $|f'(x_0)| = 1$ oder jedenfalls $|f'(x_0)| \approx 1$ ist.

Die Betrachtung solcher Situationen führt schnell zu sehr interessanten und überraschenden Iterationen.

Wählen wir zunächst $f(x) = 1 - x$, so liegt ein *Zyklus* vor, s. Abbildung 6 oben links. Hier ist $x_0 = 1/2$ und $f'(x_0) = 1$. Geringfügige Änderung an $f'(x_0)$ liefert die (langsam) konvergente Situation von Abbildung 6 oben rechts sowie die divergente Situation aus Abbildung 6 unten rechts. Da DERIVE diese Bilder streckensegmentweise aufbaut, sieht man die Iteration noch eindrucksvoller als auf dem Bild, da die Zeit beim Bildaufbau die Funktion der Variablen k übernimmt.

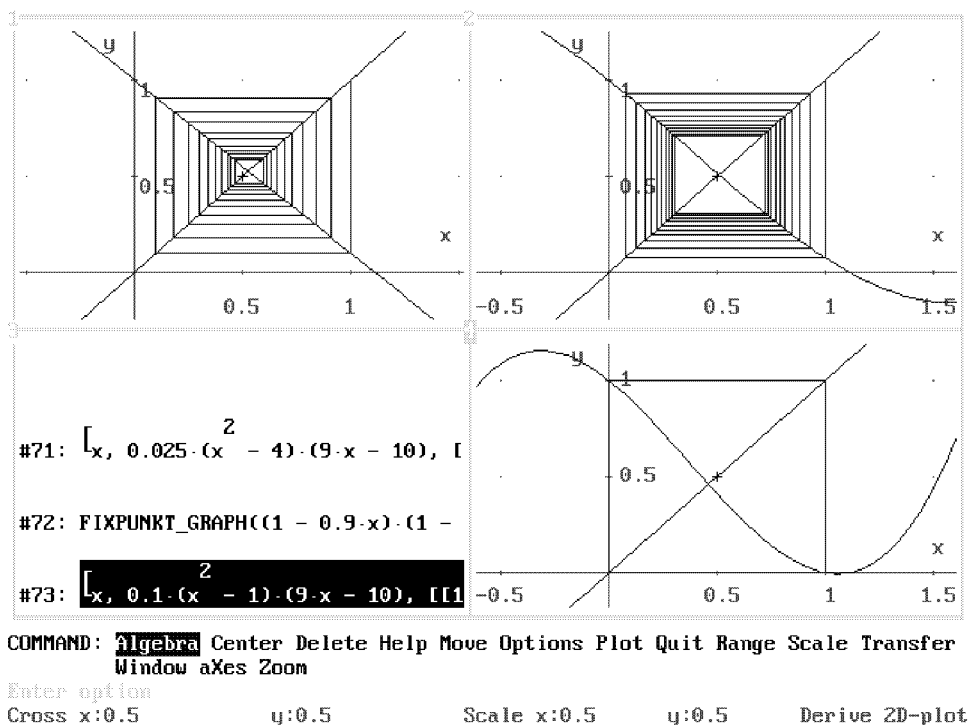
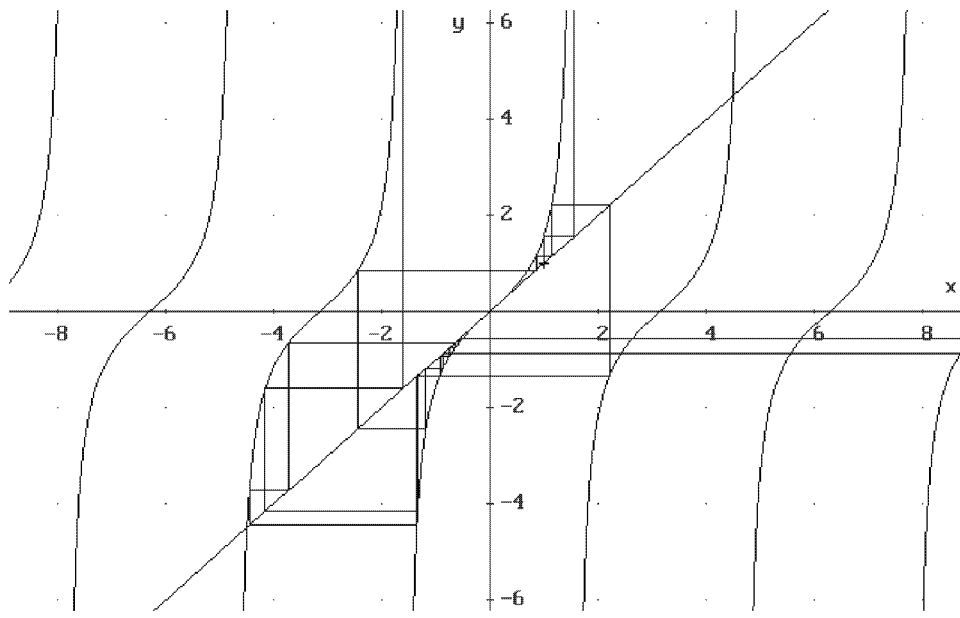


Abbildung 7: Iteration für $f(x) = 1 - 0.9x$, $(1 - 0.9x)(1 - x^2/4)$, $(1 - 0.9x)(1 - x^2)$

Ein anderes Experiment sehen wir in Abbildung 7. Ändert man die lineare Iterationsfunktion $f(x) = 1 - 0.9x$ zu der (in der Umgebung von $x_0 = 1/2$ geringfügig) nichtlinearen Iterationsfunktion $f(x) = (1 - 0.9x)(1 - x^2/4)$, so liegt weiter Konvergenz vor, die sich aber verlangsamt. Die stärkere Veränderung von f zu $f(x) = (1 - 0.9x)(1 - x^2)$ führt schließlich dazu, daß die Konvergenz des Verfahrens ausbleibt und sich wieder ein Zyklus bildet. Es fällt dem Schüler nicht schwer und es macht Spaß, sich Zwischenzustände anzusehen sowie die Situation für $f(x) = (1 - 0.9x)(1 - ax^2)$ mit $a > 1$ zu untersuchen.

Ein weiteres interessantes Beispiel betrifft die Tangensfunktion. Da diese jeden reellen Wert unendlich oft annimmt, gibt es hier ein recht chaotisches Verhalten. Die graphische Darstellung von $\text{FIXPUNKT_GRAPH}(\text{TAN}(x), x, 1, 20)$ ist in Abbildung 8 dargestellt.

Da $\text{NEWTONFUNKTION}(x - \text{TAN}(x), x)$ die Funktion $\sin x$ erzeugt, ist dies im wesentlichen gleichwertig zu dem chaotischen Verhalten des Newtonverfahrens für $\sin x$, das ich für die 3 nahe beieinanderliegenden Anfangswerte 1.3, 1.4 und 1.5 in Abbildung 9 dargestellt haben.



COMMAND: **Algebra** Center Delete Help Move Options Plot Quit Range Scale Transfer
Window axes Zoom

Enter option

Cross x:1

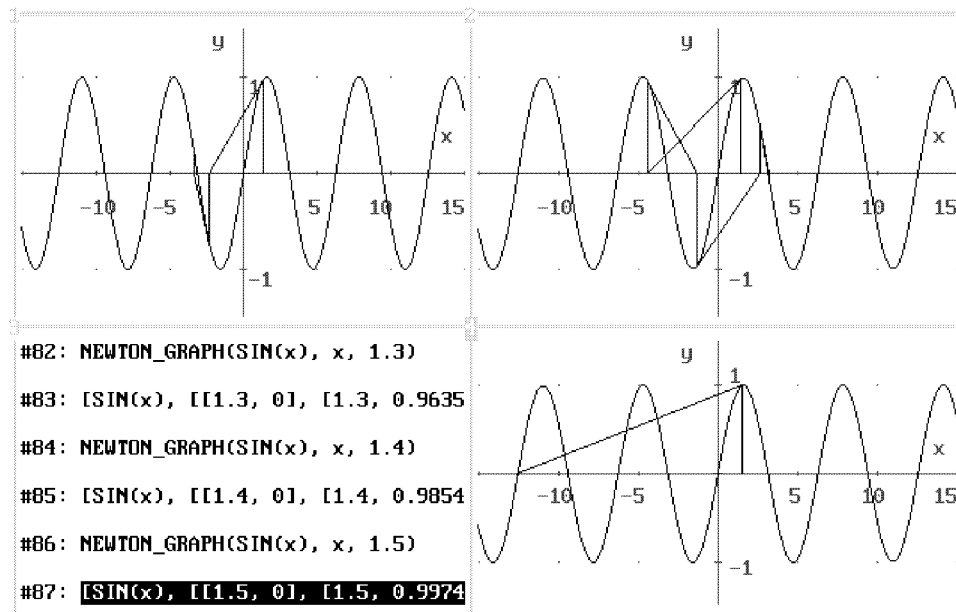
y:1

Scale x:2

y:2

Derive 2D-plot

Abbildung 8: Graphische Darstellung von $\text{FIXPUNKT_GRAPH}(\text{TAN}(x), x, 1, 20)$



#82: **NEWTON_GRAPH**(SIN(x), x, 1.3)
#83: [SIN(x), [[1.3, 0], [1.3, 0.9635]
#84: **NEWTON_GRAPH**(SIN(x), x, 1.4)
#85: [SIN(x), [[1.4, 0], [1.4, 0.9854]
#86: **NEWTON_GRAPH**(SIN(x), x, 1.5)
#87: [SIN(x), [[1.5, 0], [1.5, 0.9974]

COMMAND: **Algebra** Center Delete Help Move Options Plot Quit Range Scale Transfer
Window axes Zoom

Enter option

Cross x:0

y:0

Scale x:5

y:1

Derive 2D-plot

Abbildung 9: Chaotisches Newtonverfahren

4 Das Bisektionsverfahren

Auch das Bisektionsverfahren, welches ja im allgemeinen viel langsamer konvergiert als das Newtonverfahren, kann ebenfalls mit DERIVE graphisch dargestellt werden. Hierdurch kann das verschiedene Konvergenzverhalten durch den Schüler viel besser erfahren werden.

Hierzu verwenden wir beispielsweise die DERIVE-Funktionen

```
BISEKTIONEN_AUX(f,x,a,b,n):=ITERATES(  
IF(LIM(f,x,(ELEMENT(g_,1)+ELEMENT(g_,2))/2)=0,  
[(ELEMENT(g_,1)+ELEMENT(g_,2))/2,(ELEMENT(g_,1)+ELEMENT(g_,2))/2],  
IF(LIM(f,x,(ELEMENT(g_,1)+ELEMENT(g_,2))/2)*LIM(f,x,ELEMENT(g_,1))<0,  
[ELEMENT(g_,1),(ELEMENT(g_,1)+ELEMENT(g_,2))/2],  
[(ELEMENT(g_,1)+ELEMENT(g_,2))/2,ELEMENT(g_,2)])),  
g_,[a,b],n)
```

```
BISEKTIONEN(f,x,a,b,n):=IF(LIM(f,x,b)*LIM(f,x,a)<0,  
BISEKTIONEN_AUX(f,x,a,b,n),  
"Verfahren nicht anwendbar",  
"Randbedingungen nicht verifizierbar")
```

```
BISEKTION_GRAPH_AUX(f,x,a,b,aux):=  
[CHI(a,x,b)*f,VECTOR([[ELEMENT(ELEMENT(aux,k_),1),1-(k_-1)/10],  
[ELEMENT(ELEMENT(aux,k_),2),1-(k_-1)/10]],k_,1,DIMENSION(aux))]
```

```
BISEKTION_GRAPH(f,x,a,b):=BISEKTION_GRAPH_AUX(f,x,a,b,BISEKTIONEN(f,x,a,b,10))
```

Hierbei berechnet die DERIVE-Funktion $\text{BISEKTIONEN}(f, x, a, b, n)$ die numerische Approximation einer Lösung der Gleichung $f(x) = 0$ im Intervall $[a, b]$ mit dem Bisektionsverfahren unter der Voraussetzung, daß f an den Stellen a und b verschiedenes Vorzeichen hat. Die DERIVE-Funktion $\text{BISEKTION_GRAPH}(f, x, a, b)$ gibt eine graphische Darstellung der ersten 10 Iterationsschritte.

In Abbildung 10 sind das Newtonverfahren $\text{NEWTON_GRAPH}(\cos(x), x, 5/2, 3)$ und das Bisektionsverfahren $\text{BISEKTION_GRAPH}(\cos(x), x, 0, 3)$ zur Bestimmung der ersten positiven Nullstelle der Kosinusfunktion gegenübergestellt. Es ist deutlich zu erkennen, um wieviel langsamer das Bisektionsverfahren gegen $\pi/2$ konvergiert.

Genau wie beim Newtonverfahren verhält sich das Bisektionsverfahren bei der Sinusfunktion chaotisch: Man kann nicht vorhersagen, gegen welche der vielen Nullstellen das Verfahren konvergiert. Einige Beispiele sind in Abbildung 11 dargestellt.

Da das Bisektionsverfahren für f immer konvergiert, sofern $f(a)$ und $f(b)$ verschiedenes Vorzeichen haben, kann Divergenz wie in Abbildung 5 nicht auftreten, s. Abbildung 12. Aber: Die Konvergenz ist eben nie besonders schnell.

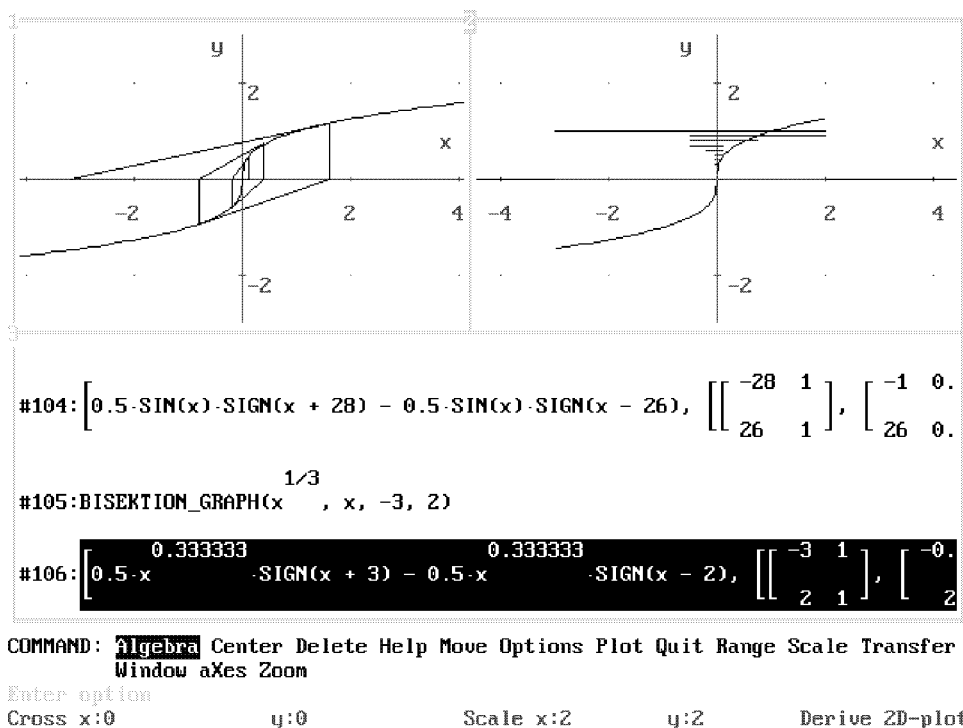


Abbildung 12: Konvergentes Bisektionsverfahren bei divergentem Newtonverfahren

5 Integrationsverfahren

Bei der numerischen Integration werden von Graphen berandete Flächen durch Rechtecke, Trapeze oder Kurven höherer Ordnung angenähert. Um die Güte dieser Approximationen besser beurteilen zu können, ist es hilfreich, auf graphische Darstellungen der Näherungsgebiete zurückzugreifen. Über die Implementierung verschiedener Integrationstechniken habe ich an anderer Stelle berichtet ([1]–[2]).

Zur graphischen Darstellung arithmetischer Riemann-Summen, der Trapezregel sowie der Simpsonregel benutzen wir die DERIVE-Funktionen

```
LINKS_GRAPH(f, x, a, b, n) := [f, VECTOR([
[a+(k_1)*(b-a)/n, 0], [a+(k_1)*(b-a)/n, LIM(f, x, a+(k_1)*(b-a)/n)],
[a+k_*(b-a)/n, LIM(f, x, a+(k_1)*(b-a)/n)], [a+k_*(b-a)/n, 0],
[a+(k_1)*(b-a)/n, 0]], k_, 1, n)]
```

```
RECHTS_GRAPH(f, x, a, b, n) := [f, VECTOR([
[a+(k_1)*(b-a)/n, 0], [a+(k_1)*(b-a)/n, LIM(f, x, a+k_*(b-a)/n)],
[a+k_*(b-a)/n, LIM(f, x, a+k_*(b-a)/n)], [a+k_*(b-a)/n, 0],
[a+(k_1)*(b-a)/n, 0]], k_, 1, n)]
```

```

TRAPEZ_GRAPH(f, x, a, b, n) := [f, VECTOR([
[a+(k_-1)*(b-a)/n, 0], [a+(k_-1)*(b-a)/n, LIM(f, x, a+(k_-1)*(b-a)/n)],
[a+k_*(b-a)/n, LIM(f, x, a+k_*(b-a)/n)], [a+k_*(b-a)/n, 0],
[a+(k_-1)*(b-a)/n, 0]], k_, 1, n)]

```

```

SIMPSON_GRAPH(f, x, a, b, n) := [[f], SUM(CHI(a+(k_-1)*(b-a)/n, x, a+k_*(b-a)/n)*(
(-2*n^2*(2*LIM(f, x, a+(k_-1/2)*(b-a)/n)-LIM(f, x, a+(k_-1)*(b-a)/n)-
LIM(f, x, a+k_*(b-a)/n))/(a-b)^2)*x^2+
(-n*(4*(a*(2*k_-2*n-1)+b*(1-2*k_))*LIM(f, x, a+(k_-1/2)*(b-a)/n)-
(a*(4*k_-4*n-1)+b*(1-4*k_))*LIM(f, x, a+(k_-1)*(b-a)/n)-
(a*(4*k_-4*n-3)+b*(3-4*k_))*LIM(f, x, a+k_*(b-a)/n))/(a-b)^2)*x+
((a*(2*k_-2*n-1)+b*(1-2*k_))*((a*(k_-n)-b*k_)*LIM(f, x, a+(k_-1)*(b-a)/n)+
(a*(k_-n-1)+b*(1-k_))*LIM(f, x, a+k_*(b-a)/n))-
4*(a*(k_-n)-b*k_)*(a*(k_-n-1)+b*(1-k_))*LIM(f, x, a+(k_-1/2)*(b-a)/n)))/
(a-b)^2), k_, 1, n)]

```

Die Funktionen `LINKS_GRAPH(f, x, a, b, n)` bzw. `RECHTS_GRAPH(f, x, a, b, n)` generieren hierbei eine graphische Darstellung des Flächeninhalts zwischen dem Graphen von f und der x -Achse im Intervall $[a, b]$ durch Approximation durch n gleich breite Rechtecke, während `TRAPEZ_GRAPH(f, x, a, b, n)` eine analoge Approximation mit Trapezen darstellt. Da Rechtecke wie Trapeze wieder von Streckensegmenten berandet sind, geschieht ihre Erzeugung wieder durch Angabe ihrer Endpunkte. Die Funktion `SIMPSON_GRAPH(f, x, a, b, n)` erzeugt eine graphische Darstellung der Approximation des Flächeninhalts durch stückweise quadratische Funktionen in jedem der n Teilintervalle gemäß der Simpsonregel. Die gegebene Definition habe ich mit Hilfe von `DERIVE` berechnet.

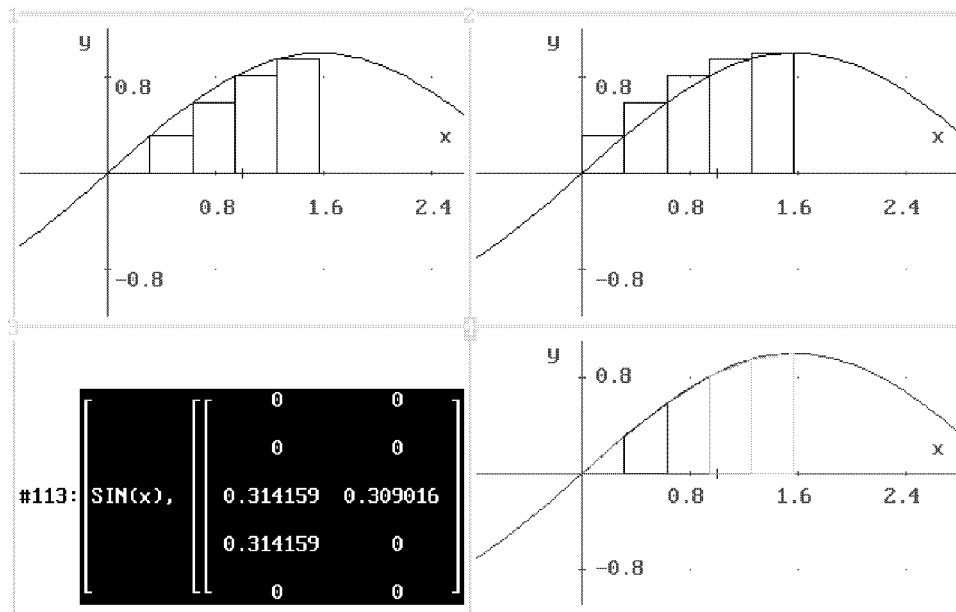
In Abbildung 13 wird das Integral

$$\int_0^{\pi/2} \sin x \, dx ,$$

d. h., der Flächeninhalt, der vom Graphen der Sinusfunktion, der x -Achse und den vertikalen Geraden $x = 0$ und $x = \pi/2$ berandet ist, durch Rechteckapproximationen von links und rechts sowie durch eine Trapezapproximation angenähert. Man sieht sehr schön, daß aufgrund der Monotonie der Sinusfunktion in dem betrachteten Intervall die linken Rechtecke allesamt zu klein und die rechten Rechtecke allesamt zu groß sind, so daß der Integralwert die linken bzw. rechten Riemannsummen als untere bzw. obere Schranken besitzt.

Es macht auch keine Mühe, sich Approximationen für größere Werte von n zu beschaffen, s. Abbildung 14.

Ferner sieht man, daß die Qualität der Trapezapproximation bedeutend besser ist, daß sie aber generell einen zu kleinen Wert liefert (welche Eigenschaft von f ist hierfür verantwortlich?). Die Simpsonapproximation ist bei diesem Beispiel so gut, daß man die Approximationsfunktion schon bei $n = 5$ nicht mehr von f unterscheiden kann, so daß wir auf eine Darstellung verzichten können, vgl. [1].

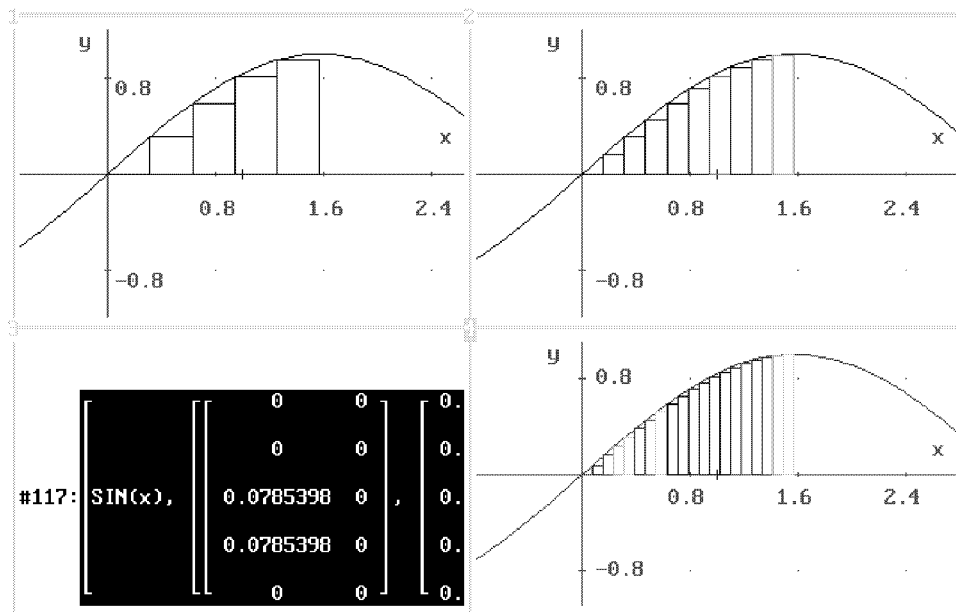


COMMAND: **Algebra** Center Delete Help Move Options Plot Quit Range Scale Transfer
Window axes Zoom

Enter option

Cross x:1 y:0 Scale x:0.8 y:0.8 Derive 2D-plot

Abbildung 13: Linke und rechte Rechteck- und Trapezapproximation



COMMAND: **Algebra** Center Delete Help Move Options Plot Quit Range Scale Transfer
Window axes Zoom

Enter option

Cross x:1 y:0 Scale x:0.8 y:0.8 Derive 2D-plot

Abbildung 14: Rechteckapproximationen für verschiedene n

Dies ist natürlich nicht bei allen Funktionen so. In welchen Fällen sind die gegebene Approximationen besonders schlecht? Dies sieht man schön bei der der Betrachtung des Integrals

$$\int_0^1 \sqrt{1-x^2} dx ,$$

s. Abbildung 15.

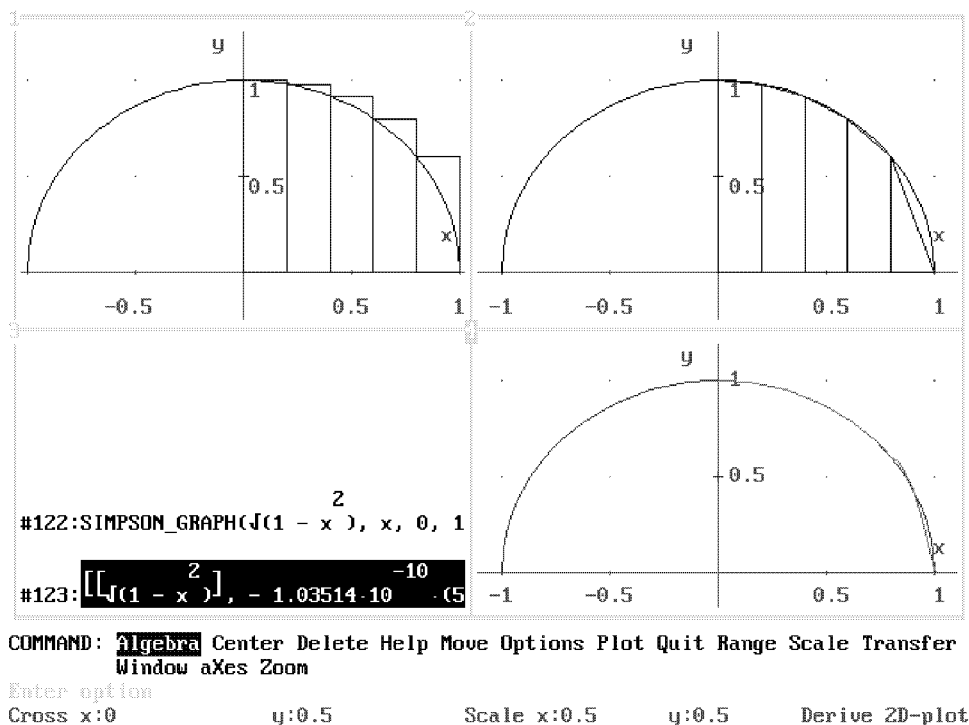


Abbildung 15: Approximationen zur Flächenberechnung des Kreises

Besonders schlecht geeignet zur numerischen Integration sind offenbar Stellen, an denen die Steigung des Integranden f besonders groß wird, und am schlimmsten ist die Situation, wenn wie in unserem Beispiel f eine vertikale Tangente besitzt. Nicht einmal das Simpsonverfahren kann einen solchen Kurvenverlauf korrekt modellieren, da ja quadratische Funktionen bekanntlich niemals vertikale Tangenten haben. In solchen Fällen müssen Programme wie DERIVE zur numerischen Approximation gegebenenfalls andere Verfahren anwenden. Man berechne dieses (und das folgende) Integral einmal mit DERIVEs `approx` Kommando numerisch!

Völlig chaotisch wird die Situation bei schnell oszillierenden Integranden wie beispielsweise $f(x) = \sin \frac{1}{x}$. Man stelle einmal einige Approximationen des (Riemann-konvergenten) Integrals

$$\int_0^{2/\pi} \sin \frac{1}{x} dx$$

dar!

6 Schlußbemerkung

Ich bin der Meinung, daß graphische Vorstellungen für den Analysis-Unterricht und für den Mathematikunterricht ganz allgemein, ungemein wichtig sind. Im Zeitalter der Computeralgebrasysteme sollten deren Fähigkeiten dazu verwendet werden, dem Schüler diese Vorstellungen zu konkretisieren. An einigen wenigen Beispielen habe ich versucht zu verdeutlichen, wie dies geschehen kann. Die gegebene Darstellung ist keineswegs vollständig, vielmehr können noch viele weitere mathematische Konzepte von Graphiken profitieren, die sich mit DERIVE erzeugen lassen.

Diese Arbeit ist ein Nebenprodukt der Bücher [3]–[4], in denen erstmals im deutschsprachigen Raum der gesamte Stoff einer Analysis-Vorlesung mit Hilfe von DERIVE präsentiert wird.

Da es sehr mühsam sein kann, die gegebenen DERIVE-Funktionen alle einzutippen (insbesondere die Funktion SIMPSON, da diese ja in einem längeren Prozeß mit Hilfe von DERIVE erzeugt wurde), bietet der Verfasser an, eine entsprechende DERIVE-Datei über email zuzusenden.

Anschrift des Verfassers: Dr. Wolfram Koepf, Seelingstr. 21, 14059 Berlin

email: koepf@zib-berlin.de

Literatur

- [1] Koepf, W., Ben-Israel, A.: Integration mit DERIVE. *Didaktik der Mathematik* **21**, 1993, 40–50.
- [2] Koepf, W., Ben-Israel, A.: The definite nature of indefinite integrals. *The International DERIVE Journal* **1**, 1994, 115–131.
- [3] Koepf, W., Ben-Israel, A., Gilbert, R. P.: *Mathematik mit DERIVE*. Vieweg-Verlag, 1993, ISBN 3-528-06549-4.
- [4] Koepf, W.: *Höhere Analysis mit DERIVE*. Vieweg-Verlag, 1994, ISBN 3-528-06594-X.
- [5] Rich, A., Rich, J. und Stoutemyer, D.: *DERIVE User Manual*, Version 2, Soft Warehouse, Inc., 3660 Waiialae Avenue, Suite 304, Honolulu, Hawaii, 96816-3236.