

Rainer Buhtz
Jens Langendorf
Olaf Paetsch
Danuta Anna Buhtz

ZUGRIF F

Eine vereinheitlichte
Datenspezifikation
für graphische Darstellungen
und ihre graphische
Aufbereitung

Technical Report 87 - 6

Konrad-Zuse-Zentrum für Informationstechnik
Heilbronner Straße 10, D - 1000 Berlin 31

1) Anforderungen an die Dateischnittstelle

Durch die Arbeitsaufnahme der Forschungsabteilungen des ZIB entstand schnell der Bedarf an Softwarebausteinen zur graphischen Darstellung von Funktionsverläufen im Koordinatensystem. Diese im Grunde alltägliche Anwendung wird zwar durch die bereits im Hause vorhandene Graphiksoftware (GKS (1,5), BIZEPS (2)) auf allen Rechnern einheitlich abgedeckt; die Kommunikation der Anwendungsprogramme mit dem Graphik-System sollte jedoch nicht über Aufrufe von FORTRAN - Unterprogrammen erfolgen, sondern über eine Zwischendatei. Die Gründe sind :

- die Anwenderprogramme sind vielfach bereits sehr umfangreich und würden durch das Hinzubinden der Graphiksoftware noch unhandlicher werden (man denke an einen möglichen Einsatz auf PC oder an die Tatsache, daß der CRAY - Rechner kein virtuelles Betriebssystem besitzt).
- es sollen auch Kurvenverläufe in einer Zeichnung darstellbar sein, die nicht im gleichen Programmlauf berechnet wurden.

Neben der eigentlichen Darstellung der Kurven stellen die Anwendersysteme an das (oder die) Graphikprogramm(e) teilweise höchst unterschiedliche Anforderungen, die am besten anhand der z.Zt. aktuellen Anwendungen dargelegt werden:

- a) Darstellung von Trajektorien gewöhnlicher Differentialgleichungen (sogen. "LARKIN-Output" (3))

Bei dieser Anwendung fallen im allgemeinen folgende Daten an: je ein Vektor aufsteigend sortierter X-Werte (Stützstellen) sowie ein Vektor der zugehörigen Y-Werte (Lösungsvektor), wobei pro X-Wert N Y-Werte vorhanden sind, wenn N die Anzahl der darzustellenden Kurven ist. Da die Stützstellen oft über mehrere Berechnungen hinweg konstant bleiben, soll es erlaubt sein, sie im Datensatz fortzulassen, wobei dann die Werte aus dem letzten Durchgang weiter gelten sollen. Wichtig ist auf jeden Fall, daß Anzahl und Werte der Stützstellen innerhalb einer Zeichnung variieren dürfen.

- b) Effizienzuntersuchungen an Programmen

Bei dieser Anwendung wird berechnet, wieviel Zeit ein Programm benötigt, um eine vorgegebene Genauigkeit zu erreichen bzw. wie oft hierfür ein bestimmter Programmteil aufgerufen werden mußte etc. Hier werden die Werte immer paarig auftreten (also je ein X-Wert (" Toleranz ") und ein Y-Wert (" Zeit " oder " Anzahl der Aufrufe ")), da man wohl das Zeitverhalten des Programms über mehrere Messungen hinweg, jedoch nie das Zeitverhalten und Anzahl der Programmaufrufe einer Messung gemeinsam in einer Graphik darstellen wird (dazu sind schon die Größenordnungen zu verschieden). Eine Zusatzanforderung ist, daß Toleranzwerte typischerweise Zehnerpotenzen sind, also die X-Achse zweckmäßigerweise logarithmisch zu skalieren ist.

c) Lösungsdarstellungen partieller zeitabhängiger Differentialgleichungen

Hier wird zum ersten Mal die dreidimensionale Graphik "betreten", wobei die volle Allgemeinheit noch nicht zugelassen werden soll. Die Gleichungen sollen nur die Ableitungen nach X und T, nicht jedoch nach Y, enthalten, und es sollen nur Lösungskurven an gewissen Zeitschichten (also für festes T) dargestellt werden. Die Datenstruktur sieht daher so aus: zu je einem festen T-Wert ein X-Vektor mit den Stützstellen und Y-Vektor mit den N Lösungen (meist wird hier pro T-Wert nur je eine Lösungskurve ausgewählt werden). Anforderungen an dreidimensionale Graphik-Software sind lediglich die Möglichkeit der Darstellung unter verschiedenen Blickwinkeln (in Parallelprojektion). Ferner soll es zukünftig möglich sein, Punkte mit gleichen Stützwerten an verschiedenen Zeitschichten zu verbinden, um deren Wanderung zu beobachten.

2) Die Dateispezifikation ZUGRIFF

Als Versuch, die oben erwähnten drei Anwendungen über ein einheitliches Datenformat abzuhandeln, wurde die Dateischnittstelle ZUGRIFF (ZIB Universal Graphical Input File Format) definiert. Allgemeinere dreidimensionale Datenformate (z.B. für CAD-Anwendungen) blieben bewußt unberücksichtigt, um die Spezifikation nicht zu überfrachten und da hierfür auch bereits geeignete Datenformate existieren, z.B. die neutrale Netzdatei NND des TDLG - Systems der TU Berlin (6).

Eine ZUGRIFF-Datei enthält (zwingend vorgeschrieben) die folgenden drei Datentypen sowie (optional) Kommandos zur Gestaltung der Graphik, die stattdessen auch im Dialog mit dem Graphik-Programm eingegeben werden können.

a) Namensvereinbarung

&NAME name1,name2,...,namen oder

&NAME n

Dieser Kennsatz muß genau einmal vor der ersten Datendefinition vorhanden sein. Er spezifiziert die Anzahl und die Namen der in der Datei enthaltenen Kurven. Im ersten Fall wird die i-te Kurve mit dem i-ten Namen bezeichnet, im zweiten Fall werden Standardbezeichnungen verwendet.

Beispiele :

&NAME H2,02,H20,H*,0H*,H02

für 6 Kurven

&NAME 5

für 5 Kurven mit Standardbezeichnung

b) Datendefinition

&DEF X=a,Y=b,(/Z=c/)(/NOKEY/)

Durch diesen Kennsatz werden definiert :

- die Namen der horizontalen , vertikalen und räumlichen Achsen in dieser Reihenfolge (für die Beschriftung)
- die Anzahl der Werte in der entsprechenden Dimension.

Als "Anzahl" darf auch der Wert "N" angegeben werden (auch als Replikator). Er entspricht der im &NAME-Satz definierten Anzahl der Kurven.

Beispiele :

```
&DEF T=1,Y=N
```

Die horizontale Achse wird mit "T" beschriftet, die vertikale mit "Y". Im Datensatz stehen immer je ein T-Wert und N Y-Werte.

```
&DEF X=10,Y=10*N,T=1
```

Die räumliche Achse wird mit "T" beschriftet. Pro T-Wert stehen 10 X-Werte und N*10 Y-Werte im Datensatz.

Bemerkungen :

Gemäß 1c) kann die Anzahl der Z-Werte höchstens 1 sein.

Der Parameter NOKEY steuert, ob im Datensatz selbst auch die Achsenbezeichner (etwa "X=") stehen dürfen (wie es bei Fortran-Ausgabe über NAMELIST der Fall ist).

Kommt in der Datei der &DEF-Satz mehrmals vor (etwa, weil sich die Anzahl der Stützwerte ändert), so darf dabei nicht von 2D - auf 3D - Graphik umgeschaltet werden und umgekehrt.

c) Daten

Ein Datensatz wird eingeleitet durch

```
&DATA
```

Es folgen Zahlenwerte gemäß der &DEF-Spezifikation.

Der Datensatz darf (muß jedoch nicht) mit &END abgeschlossen werden. Jedes andere Schlüsselwort (&DATA oder &DEF) beendet den Datensatz ebenfalls.

Beispiel :

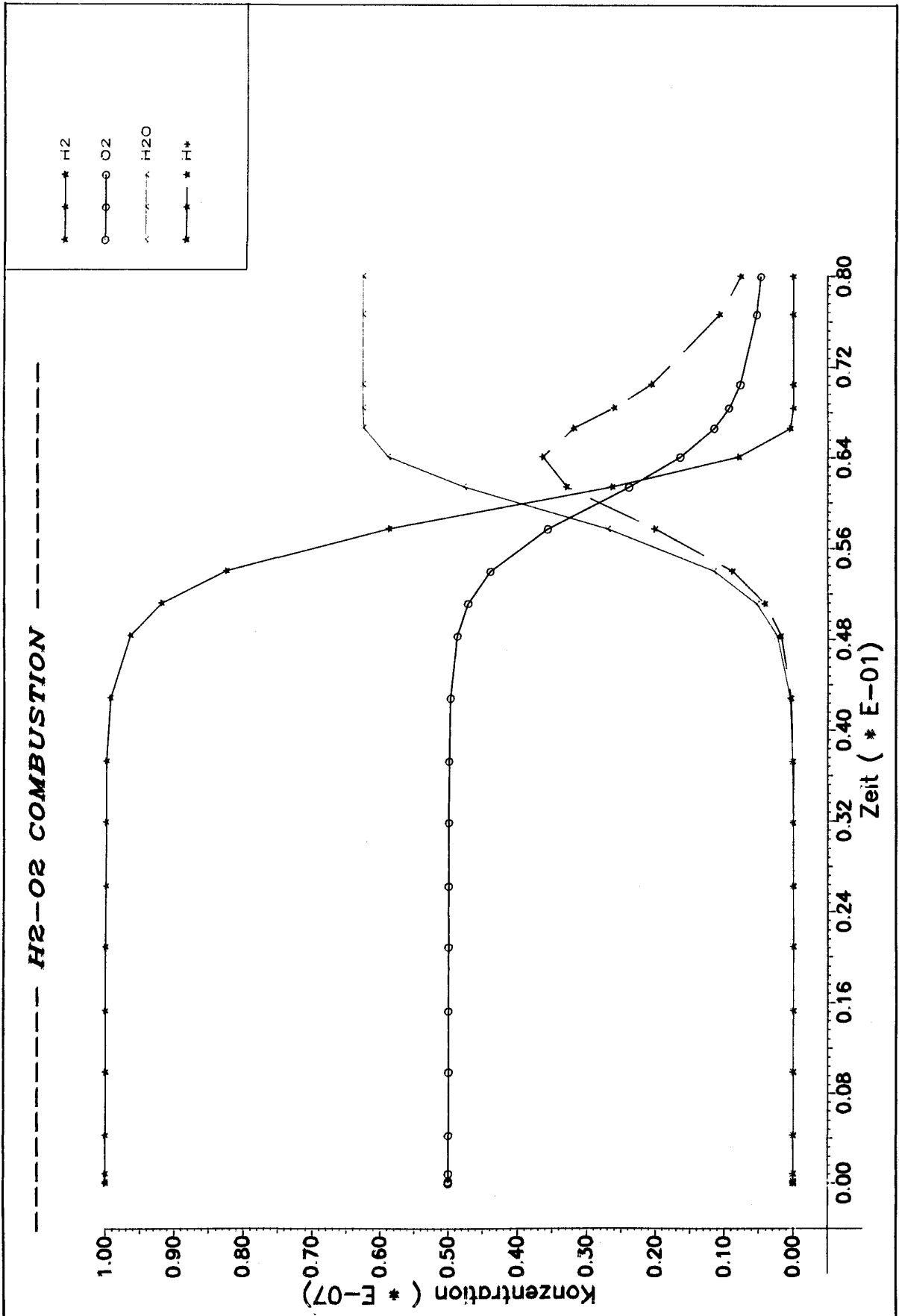
```
&DATA
X=0.0000E+00
Y=0.1000E-05, 0.2000E-05,...
```

3) Die Verarbeitung einer ZUGRIFF-Datei im Dialog

Im vorangehenden Kapitel wurde spezifiziert, wie eine ZUGRIFF-Datei auszusehen hat. Es ist daher in erster Linie eine Schnittstellenfestlegung für den Anwendungsprogrammierer. Im späteren Produktionsbetrieb braucht dies den Programmbenutzer nicht mehr zu interessieren, da die Anwendungsprogramme (z.B. aus der Numerik) als Output automatisch den richtigen Input für die Graphikprogramme erzeugen.

Zur Zeit existiert ein Graphikprogramm namens GRAZIL (Graphical ZIB Language) zur Verarbeitung des LARKIN - Output (s.1a). Es

Fig. 1



wird derzeit auf ZUGRIFF - Format umgestellt (bleibt dabei aber aufwärtskompatibel) und kann danach auch die Anwendungen 1b) und 1c) bedienen.

Das Programm führt mit dem Benutzer einen Dialog, um die graphische Ausgabe der ZUGRIFF - Datei zu steuern.

Eine eingehende Beschreibung aller Kommandos würde den Rahmen dieses Reports sprengen. Es werden exemplarisch nur die wichtigsten Befehle skizziert.

Eine typische Graphik, wie sie bei der Verarbeitung einer ZUGRIFF-Datei entsteht, zeigt Fig.1.

Einige Graphik-Elemente entstammen der ZUGRIFF-Datei:

- die Achsenbeschriftungen "Zeit" bzw. "Konzentration" (s. &DEF) (die Skalierungsfaktoren werden automatisch gewählt, um die Zahlen an den Achsen-Ticks nicht zu lang werden zu lassen).
- die Legendenbeschriftungen "H2", "O2" etc. (s. &NAME)
- (natürlich) die Stützpunkte selbst (in der Graphik durch Markersymbole gekennzeichnet).

Für alle anderen Optionen (Farben, Linientypen, Überschrift etc.) gelten folgende Regeln:

- a) sie können im Dialog eingegeben werden, z.B.

```
HEAD '----- H2-O2 COMBUSTION -----'
```

- b) sie können in der ZUGRIFF-Datei stehen, z.B.

```
&HEAD '----- H2-O2 COMBUSTION -----'
```

- c) die aktuellen Werte aller Parameter, die während des Programmlaufes (woher auch immer) gesetzt worden sind, können auf eine Kommandodatei ausgegeben werden und später von dieser wieder eingelesen werden.

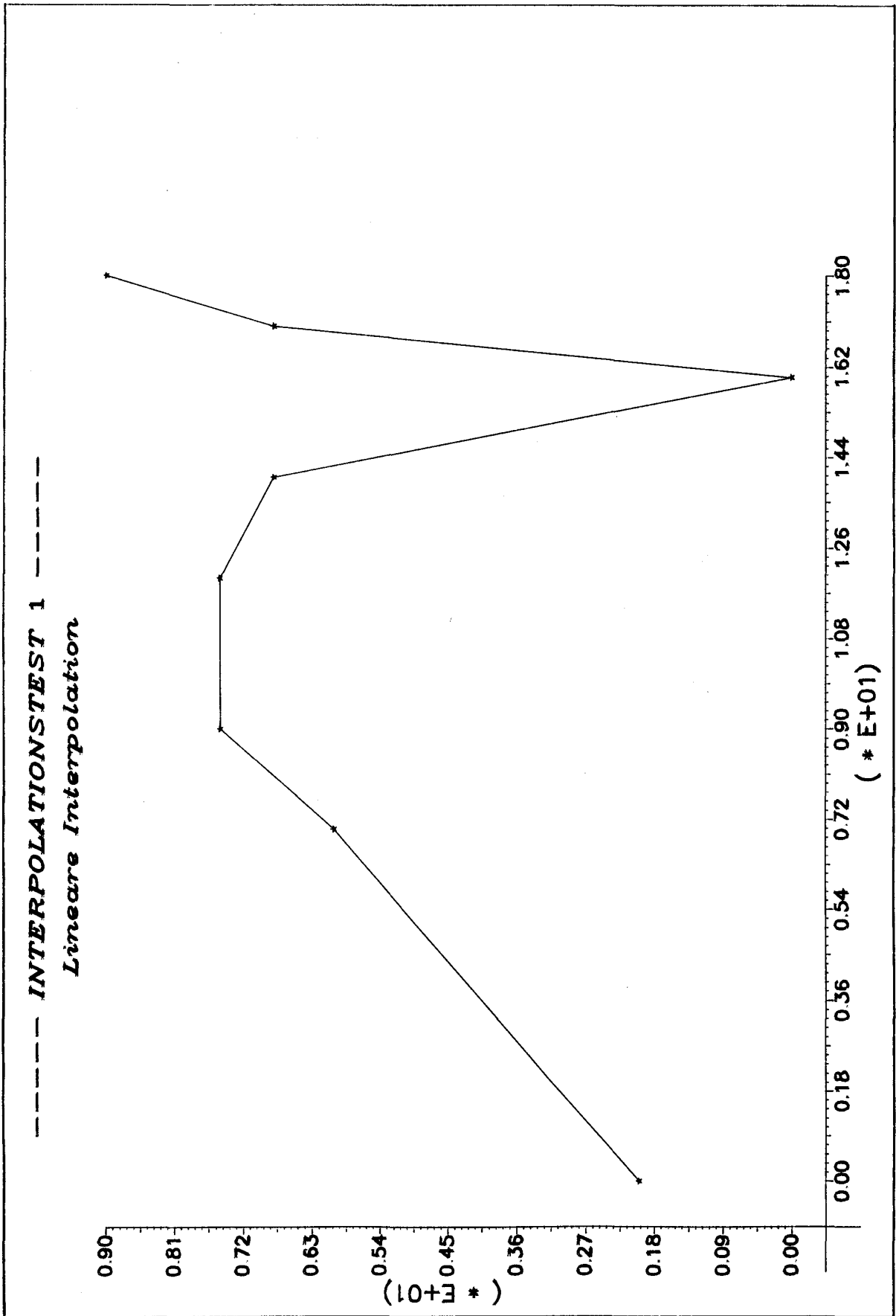
Es gilt die folgende Hierarchie:

Zunächst setzt das Graphikprogramm Standardwerte fest. Diese können von Kommandos in der ZUGRIFF-Datei überschrieben werden. Danach kann durch ein Spezialkommando eine früher erstellte Kommandodatei eingelesen werden. Die höchste Priorität haben schließlich Eingaben "von Hand", also im Dialog.

Folgende Optionen waren für die Graphik in Fig.1 noch gesetzt:

- es wurde spezifiziert, daß die Kurven in verschiedenen Farben gezeichnet werden sollen.
- es wurde spezifiziert, daß die Kurven in verschiedenen Linientypen dargestellt werden sollen.
- es wurde spezifiziert, daß die Stützpunkte durch Marker gekennzeichnet werden sollen.

Fig. 2



Weitere wichtige Parameter:

- Angabe, ob die Achsen linear oder logarithmisch skaliert werden sollen.
- Workstation Type des Gerätes, auf dem die Ausgabe erfolgen soll (dieser Parameter kann während eines Programmlaufs im Dialog geändert werden. So kann man sich z.B. das Bild zuerst am Terminal ansehen und dann auf einem Plotter ausgeben).

In der Beispielgraphik ist deutlich zu erkennen, daß die Stützpunkte der Kurven durch Geradenstücke verbunden wurden (lineare Interpolation, in GKS-Sprechweise "Polyline"). Ein solches Bild ist in vielen Fällen zwar zum schnellen Sichten der Graphik auf dem Terminal ausreichend, für Zeichnungen zur Veröffentlichung jedoch unbefriedigend (s. Fig.2).

Greift man jedoch das Thema Interpolation auf, so scheiden sich die Geister: es gibt wohl kein Interpolationsverfahren, das für alle Anwendungsfälle befriedigende Ergebnisse liefert, wie wir anhand von Beispielen darlegen werden. Problemzonen sind dabei erfahrungsgemäß Stellen, an denen die Kurve plötzlich stark ansteigt oder abfällt, oder wenn sie nach einem starken Anstieg oder Abfall plötzlich sehr flach wird.

Das Graphik-Programm GRAZIL bietet daher - zusätzlich zur linearen Interpolation - drei Interpolationsverfahren zur Auswahl an, von denen eines wohl immer optisch befriedigend arbeitet.

Zunächst bietet das im ZIB verwendete GKS über die Funktion "Generalized Drawing Primitive" (GDP) eine Kurveninterpolation an. Sie arbeitet mit einer lokalen 4-Punkte-Interpolation durch Polynome 3. Grades nach G.Renner und V.Pochop (7). Die entstehende Kurve ist stetig differenzierbar und hat die Eigenschaft, bei drei oder mehr auf einer Linie liegenden Stützpunkten nicht zu oszillieren, sondern wirklich eine Gerade zu liefern. Diese Kurven zeigen stets ein "schönes Bild", nur handelt es sich leider um ebene Kurven, keine Funktionen. Der Funktionsbegriff als solcher ist dem GKS fremd. Es kennt nur Punktserien $(X(i), Y(i), i=1, \dots, n)$ in der Ebene.

Daher kann es passieren (bei steilem Anstieg oder Abfall der Kurve, s.o.), daß die Kurve einen "Bauch" bekommt, d.h. sie schwingt ein wenig zur Seite aus, um den Abfall abzuschwächen. Dadurch geht in solchen Fällen der Funktionscharakter verloren (s.Fig.3).

Das im ZIB verfügbare Graphikpaket BIZEPS (2) enthält eine Funktionsinterpolationsroutine namens BFLINE. Sie berechnet zu je vier Stützpunkten ein Polynom 5. Grades aus je einer Parabel durch den 1. bis 3. resp. 2. bis 4. Punkt sowie einer Gewichtsfunktion. Das Verfahren liefert einen stetig differenzierbaren Funktionsgraphen. Leider oszilliert die Kurve an sehr flachen Stellen (s. Fig. 4).

Das sicherste Verfahren ist schließlich die stückweise kubische hermitesche Interpolation mit Hilfe des Programmpaketes PCHIP (4). Das Verfahren erzeugt Funktionsgraphen ohne Oszillationen oder zu starke Ausschläge. Man beachte in diesem Zusammenhang, daß dies

Fig. 3

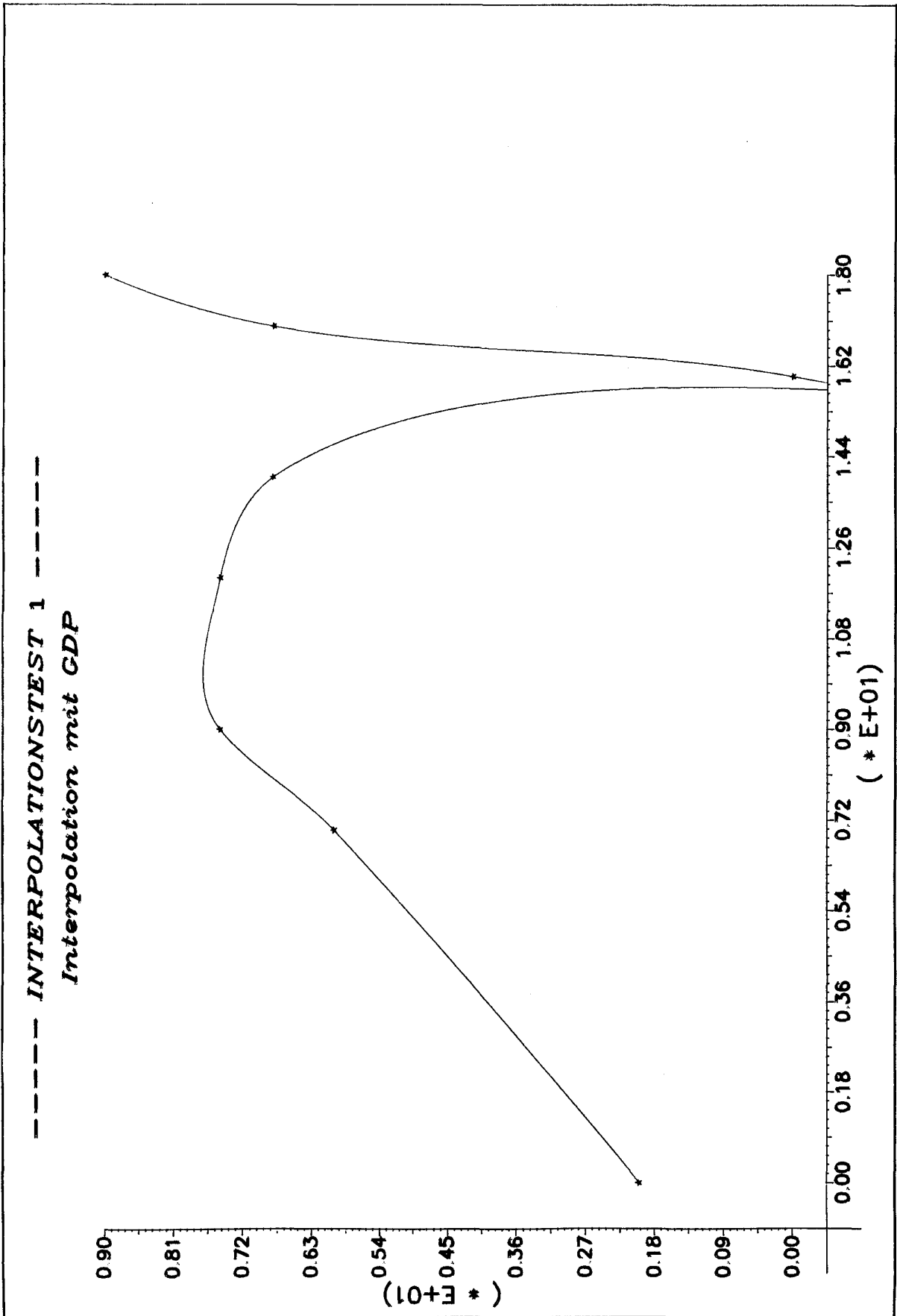


Fig. 4

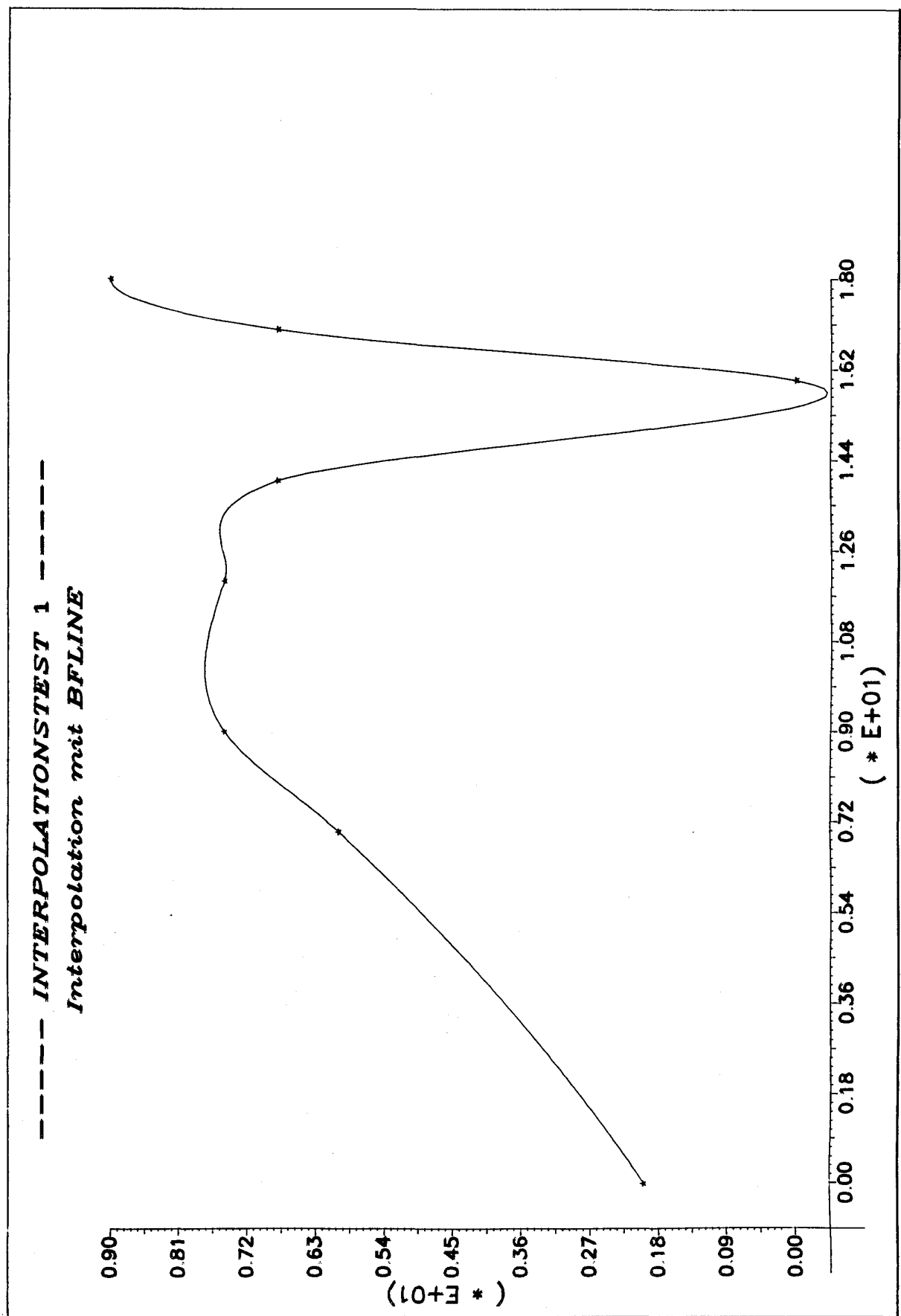
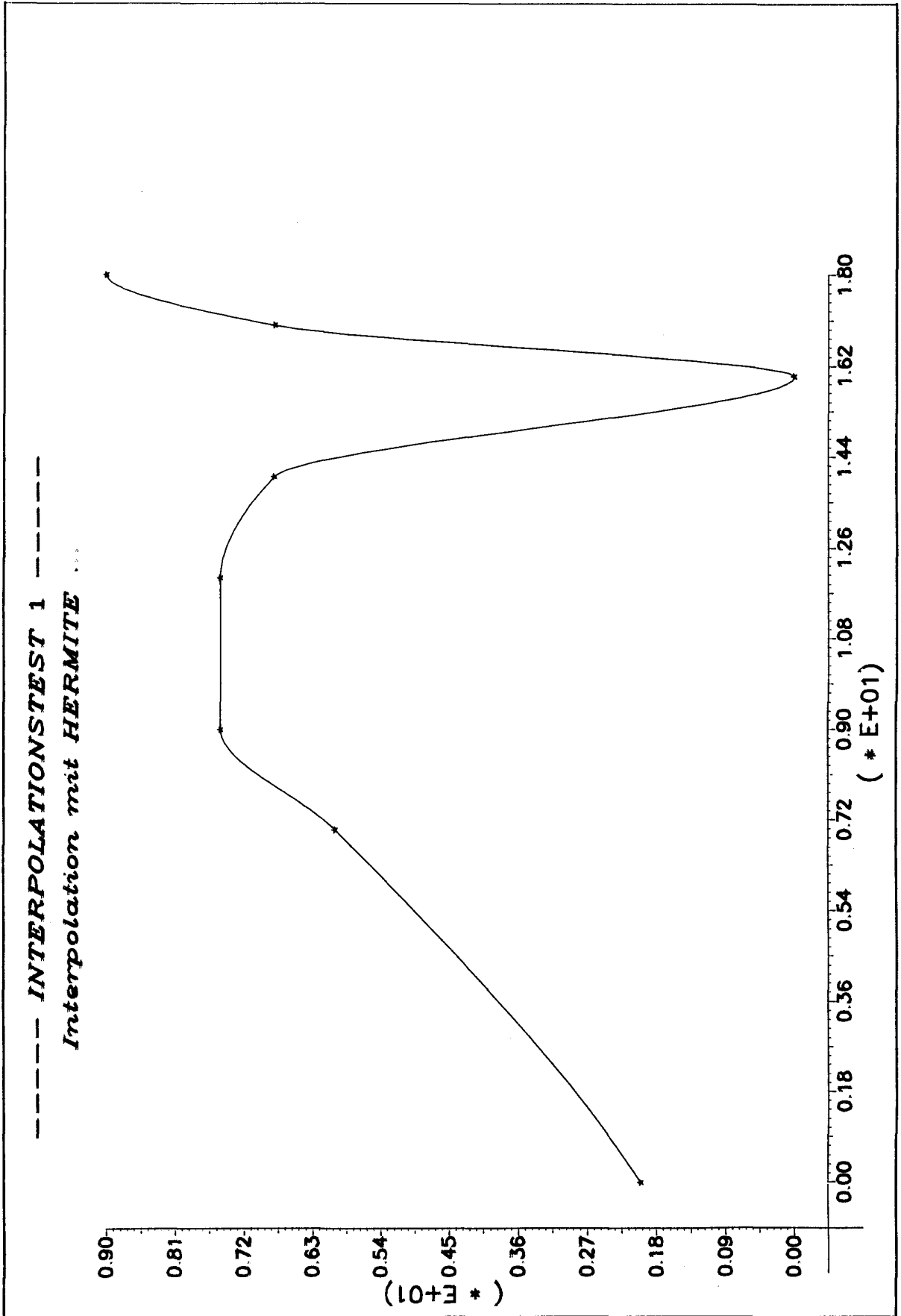


Fig. 5



das einzige Verfahren ist, bei dem (im gewählten Beispiel) die Funktion ihr Minimum genau an der Stützstelle mit dem kleinsten Y-Wert annimmt (s. Fig.5).

Allerdings sind die Kurven nicht mehr stetig differenzierbar, was sie manchmal etwas eckig aussehen läßt.

Zusammenfassend ist zu sagen: im allgemeinen liefert die GDP-Funktion des GKS die ansprechendsten Ergebnisse. Zudem bleiben diese Kurven auch bei nachträglicher Vergrößerung glatt, da hier das GKS selbst nur die Stützpunkte speichert.

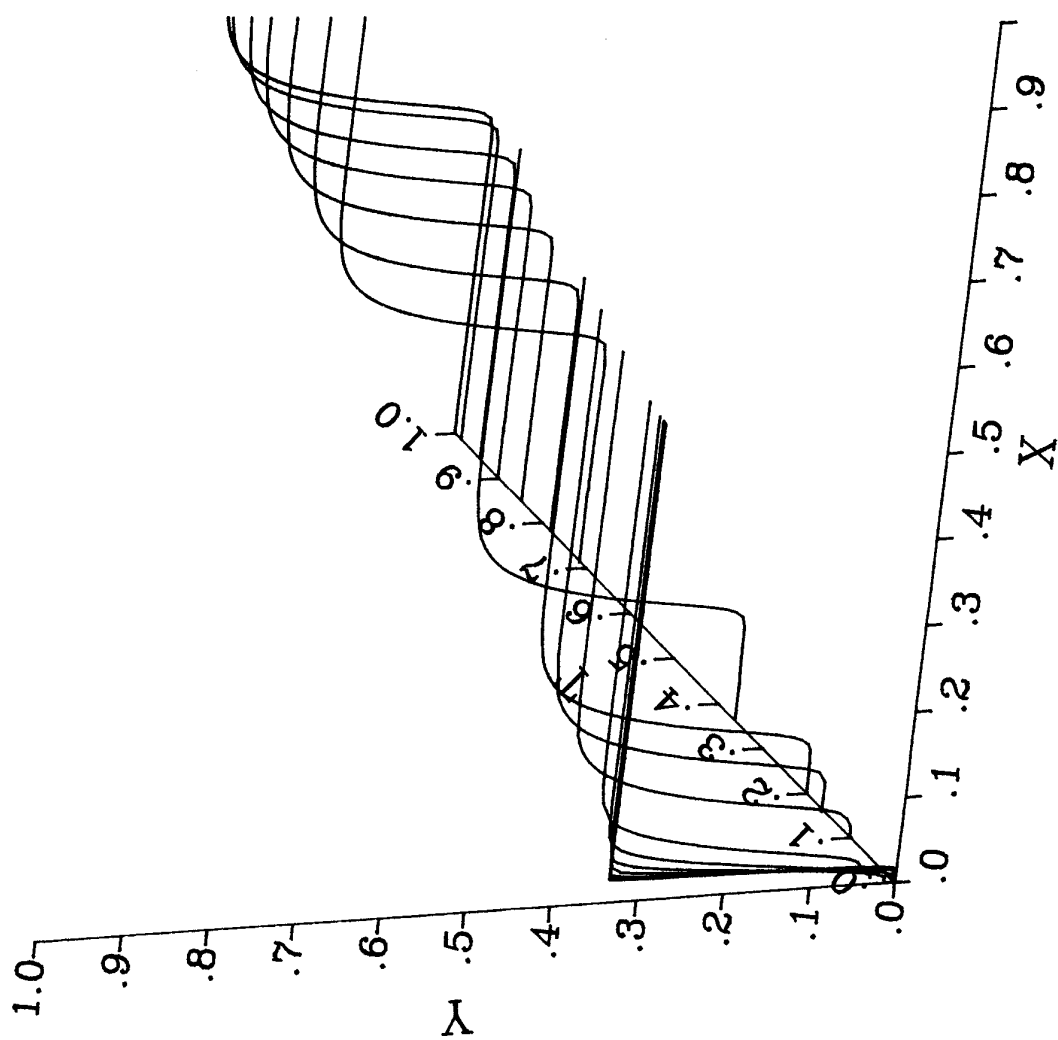
In den Fällen, wo die GDP-Funktion versagt, muß man auf BFLINE oder auf die hermitesche Interpolation ausweichen. Hier sollte für die Auswahl des Verfahrens der Augenschein ausschlaggebend sein. Eine automatische Auswahl des Interpolationsverfahrens durch das Graphik-Programm ist z.Zt. nicht vorgesehen. Man ist jedoch bei Benutzung des hermiteschen Verfahrens stets auf der sicheren Seite.

Interpolationsverfahren bei 3D-Darstellungen sind noch nicht implementiert. Es ist auch z.Zt. noch unklar, ob eine ebene Interpolation an jeder Zeitschicht ausreichend wäre.

Der gesamte 3D-Bereich des GRAZIL-Programms befindet sich noch in der Konzeptionsphase. Er orientiert sich im wesentlichen an einem synthetischen Kameramodell (8). Wichtig ist hierbei, daß der Benutzer das "Sichtvolumen" (als Äquivalent zum zweidimensionalen Window) beliebig ausrichten kann und dadurch die Kurvenschar aus verschiedenen Blickwinkeln betrachten kann. Eine Beispielgraphik zeigt Fig.6.

Perspektivische Projektionen werden nicht unterstützt, da die Vergleichbarkeit der Kurven darunter leiden würde. Der gesamte benötigte Funktionsumfang wird sich auf diesem Gebiet sicher erst im Verlauf der praktischen Anwendung herauskristallisieren.

Fig. 6



Referenzen

- (1) J.Bechlars, R.Buhtz: GKS in der Praxis, Springer-Verlag Berlin 1986
- (2) J.Bechlars, R.Buhtz, W.Hecht: BIZEPS Version 2.1 Benutzerhandbuch, Freie Universität Berlin, Zentraleinrichtung für Datenverarbeitung 1984
- (3) P.Deuflhard, G.Bader, U.Nowak: LARKIN - A Software Package For The Numerical Simulation Of Large Systems Arising In Chemical Reaction Kinetics. In: K.H.Ebert, P.Deuflhard, W.Jaeger (ed.): Modeling Of Chemical Reaction Systems. Springer Ser. Chem. Phys. 18, 1981
- (4) F.N.Fritsch: Piecewise Cubic Hermite Interpolation Package, Mathematics and Statistics Division, Lawrence Livermore National Laboratory, Livermore California, 1982
- (5) International Standard ISO 7942
Information Processing Systems
Computer Graphics
Graphical Kernel System (GKS)
Functional Description
ISO 7942/1985
- (6) H.Nowacki, R.Hammerschmidt, K.Parlar, K.Tsantas, H.Wetzel: Nutzeranleitung für das Programmsystem TDLG (Three Dimensional Layer On GKS), Technische Universität Berlin, Institut für Schiffs- und Meerestechnik, Berlin 1986
- (7) G.Renner, V.Pochop: A New Method For Local Smooth Interpolation, Eurographics Conference Proceedings, Darmstadt 1981
- (8) Status Report Of The Graphical Standards Planning Committee, in: Computer Graphics Siggraph - ACM, Vol 13, No. 3, August 1979