CHAOS*ultd* GEM

Version 6.40

1.1.1994

von

Th. Morus Walter

Inhaltsverzeichnis

1 Copyright **1**

2 Allgemeines **2**
2.1 Möglichkeiten und Grenzen . . . 2
2.2 Unterstützte Fraktaltypen . . . 3

3 Installation **3**
3.1 Hardwarevorraussetzungen . . . 3
3.2 Installation . . . 4
3.3 Lieferumfang . . . 4

4 Danksagung **5**

5 Entwicklungsstand **6**
5.1 Änderungen gegenüber früheren Versionen . . . 6
5.2 Helfer gesucht! . . . 6
5.3 Updates und Bezugsmöglichkeiten . . . 7

6 Fehler (und was man dafür halten könnte) **7**
6.1 eliminierte Fehler . . . 7
6.2 bekannte Fehler . . . 7
6.3 unbekannte Fehler . . . 8
6.4 Fehler in der Anleitung . . . 8

7 zu dieser Anleitung **9**

I CHAOS*ultd* GEM **10**

1 Allgemeine Bedienungshinweise **10**
1.1 Hinweise zum Betrieb . . . 10
1.2 Booten des Programmes . . . 10
1.3 Speicherverwaltung . . . 10
1.4 Dialoge . . . 11
1.5 CHAOS*ultd* und Multitasking . . . 12
1.6 Fenster . . . 13
1.7 Objekte . . . 13
1.8 Selektieren von Objekten im Objekt-Fenster . . . 14
1.9 Bedienungselemente zum Aufruf von Funktionen . . . 15
1.10 Online-Hilfe . . . 16

2 Bilder erzeugen **16**
2.1 neue Bilder – Eingabe der Parameter . . . 16
2.2 Eingabehilfen . . . 18
2.3 Berechnung von Bildern . . . 20
2.4 Neu berechnen . . . 20
2.5 extended Neu, Block, ... 21

3 Bilder verwalten **21**
3.1 Film edieren . . . 22
3.2 Bilder packen . . . 22
3.3 Bilder konvertieren . . . 23
3.4 Löschen . . . 23

4 Anzeigefunktionen **24**
4.1 Bilder anzeigen . . . 24
4.2 Bildvergleich . . . 25
4.3 Block zeigen . . . 25
4.4 Konstante anzeigen . . . 25
4.5 Koordinaten anzeigen . . . 26
4.6 Punktinfo . . . 26
4.7 neues Objekt . . . 26
4.8 Bild-Parameter anzeigen . . . 26
4.9 Farben . . . 27
4.10 Fraktal-Parameter anzeigen . . . 27
4.11 Info anzeigen . . . 27

5 Input/Output **27**
5.1 Speichern . . . 27
5.2 Laden . . . 28
5.3 Drucken . . . 28
5.4 Clipboard . . . 28

6 Einstellungen **29**
6.1 Einstellungen I . . . 29
6.2 Einstellungen II . . . 29
6.3 Parameter-Voreinstellungen . . . 30
6.4 Parameter speichern . . . 30

II die Fractal-Routinen **31**

1 der Algorithmus **31**

2 Parameter **31**
2.1 benötigte Parameter . . . 31
2.2 Eingabe der Parameter . . . 32
2.3 3d Fractale . . . 33
2.4 freie Fractale . . . 34

3 unterstützte Optionen **35**
3.1 Parameter anzeigen . . . 35
3.2 Bild-Info anzeigen . . . 35
3.3 Punktinfo . . . 35
3.4 Block/Konstante zeigen . . . 35
3.5 neu berechnen . . . 35

III die Hüpfer-Routinen **37**

1 der Algorithmus **37**

2 Parameter **37**

3 unterstützte Routinen **37**

IV die L-System-Routinen **39**

1 Definition der Parameter **39**

2 Eingabe der Parameter **40**

3 Berechnung von L-Systemen **41**

V die Feigenbaum-Routinen 42

VI die XCHAOS-Routinen 43

1 Barnsley-Fractale 44

2 seltsame Attraktoren 45

3 Diffusion 46

4 Popcorn 46

5 Kreis^2 46

VII Hardcopy-Routinen 47

1 24 Nadel-Matrixdrucker-Treiber 47

2 Gdos-Treiber 48

1 Copyright

CHAOS*ultd* GEM V6.40 geschrieben von Th. Morus Walter

CHAOS*ultd* GEM V6.40 (c) 1993/4 by NEVER READY SOFTWARE, Th. Morus Walter

Th. Morus Walter
Schulstr. 22
93080 (W-8401 Pentling)
(Federal Republic of Germany)
Tel. 0941 949802
e-Mail: morus.walter@rphs1.physik.uni-regensburg.de

Dieses Programm ist **Postcardware**!

Das Programm darf und soll frei kopiert und weitergegeben werden, aber nur vollständig. D.h. es müssen alle Dateien, die man mit CHAOS*ultd* erhalten hat, auch weitergegeben werden – bzw. die diese Dateien enthaltenden Archive. Komerzielle Nutzung ist ausgeschlossen, mit Ausnahme der für den Vertrieb von PD-Disketten üblichen Kopiergebüren. Die Aufnahme in eine Public-Domain Disketten-Serie bedarf der Zustimmung des Autors.
Die Aufnahme des Programmes in Internet-Fileserver oder in Mailboxen ist erlaubt und erwünscht, vorausgesetzt für das Abrufen des Programmes wird *keine* Gebür verlangt.
Wem das Programm gefällt, und wer es nutzt, ist verpflichtet, mir – dem Autor – eine Ansichtskarte als kleines Dankeschön zu schicken. So erfahre ich ein bißchen was über die User meines Programmes (womöglich gibt es gar keine) und habe etwas (hoffentliche positive) Resonanz, die Rückseite der Postkarte läßt sich ja für Anregungen und konstruktive Kritik (andere wird überlesen) nutzen (ich bitte allerdings um Verständnis, daß ich schriftliche Fragen nur bei frankiertem Rückumschlag beantworten kann). Ich denke, eine solche Postkarte ist nicht zu viel verlangt, für ein Programm wie CHAOS*ultd* GEM.

Hinweis

Die Benutzung des Programmes geschieht auf *eigene* Gefahr. Der Autor garantiert weder für irgendwelche Schäden (am Rechner oder auch am Benutzer) noch für die korrekte Funktionsweise des Programmes. Insbesondere wird auch *nicht* garantiert, daß den erzeugten Bildern wirklich die beschriebenen Algorithmen zugrundeliegen, wenngleich ich natürlich davon ausgehe.

Wider die neue (und alte) Rechte!
Nie wieder Krieg! Nie wieder Faschismus!

2 Allgemeines

CHAOS*ultd* GEM V6.40 ist die neueste und vorläufig endgültige Version von CHAOS*ultd* GEM, das eine Weiterentwicklung von CHAOS*ultd* V5.0 ist.
CHAOS*ultd* GEM ist, wie das Suffix GEM schon andeutet, konsequent GEM-konform programmiert (von Programmfehlern abgesehen). Das Programm sollte deshalb in jeder Auflösung und auf jeder Graphikkarte laufen. Zur Ausgabe wird ausschließlich das VDI verwendet.
Trotz der Tatsache, daß die GEM-Konformität immer noch wenigstens zwei Nachteile gegenüber der Version 5.0 mit sich bringt, erkläre ich CHAOS*ultd* GEM V6.40 hiermit (anders als noch V6.00 und V6.01) zum Nachfolger der Versionen V5.0x.
Die Nachteile sind zum einen, die geringere Anzeigegeschwindigkeit, ein Effekt, der sich in Grenzen halten dürfte (nachgemessen hab ich's nicht) und (schlimmer) die Unmöglichkeit der Darstellung in ST-low durch Umschalten der Auflösung. Letzteres ist schon bitter, auf der anderen Seite arbeite ich eh meist in monochromer Auflösung.

2.1 Möglichkeiten und Grenzen

Abgesehen von der endlichen Rechengeschwindigkeit und dem endlichen Speicher ist CHAOS*ultd* GEM praktisch grenzenlos (deshalb auch das *ultd*, das für *unlimited* steht, im Namen). D.h. praktisch alle im normalen Betrieb auftretenden Einschränkungen sind durch Rechengeschwindigkeit und Speichergröße bedingt.
Daneben gibt es die folgenden programmbedingten Einschränkungen:

- maximal 9999 Objekte gleichzeitig im Speicher
- Bildgrößen von 64-32000 x 40-32000 Pixel

  im einzelnen:

  - minimale Bildbreite: 64 Pixel
  - minimale Bildhöhe: 40 Pixel
  - maximale Bildbreite und -höhe: 32000 Pixel
  - die Bildbreite muß durch 16, die Bildhöhe durch 4 teilbar sein
- 1-8 Bildebenen, d.h. 2-256 Farben; bei einer Bildebene (mono- bzw. genauer duochrom) sind noch 2x2 und 4x4 Raster möglich

  echte Einschränkung: keine Unterstützung von High-/True-Color (bei den meisten Bildtypen wäre das aber eh nicht sinnvoll)
- maximal 32767 Bilder pro Film

Anmerkungen:

nicht alle Bilderzeugungsroutinen unterstützen alle Bildformate, insbesondere gibt es Routinen, die nur 2 Farben (und auch keine Raster) zulassen

die theoretisch mögliche Bildgröße von 32000x32000 Pixeln bei 256 Farben (8 Bildebenen) hat einen Speicherplatzbedarf von 976.5625 Megabyte (also ca. 1 Gbyte). Zwar kann das Bild für die Bildverwaltung gepackt werden, um es zu berechnen oder anzuzeigen muß die Bitmap aber entpackt vorliegen. Ich möchte auch nicht wissen, wie lange es dauern würde ein solches Bild mit ca. 10^9 Punkten zu berechnen.

Noch eine weitere Anmerkung zur Speicherverwaltung: CHAOS*ultd* GEM verwendet an einigen Stellen in Zeigern -1l als Flag. Damit dürften sich Schwierigkeiten bei Verwendung von Speicher oberhalb von $2^{31} = 2$ GigaByte ergeben, insbesondere kann man[1] wohl nicht mehr als 2 GigaByte Speicher nutzen. Ich betrachte diese Einschränkung allerdings als ausgesprochen theoretisch (um 1 GigaByte Speicher auch nur zu löschen – wenn einer tatsächlich soviel Speicher bezahlen kann – braucht eine 8 Mhz 68000 CPU (mit sowas arbeite ich immer noch) ca. 1000 Sekunden (1 Sekunde pro Megabyte habe ich mal abgeschätzt, auch wenn es wohl sicher noch schneller geht) also eine viertel Stunde.

[1] wer will, kann sich jeder Stelle, an der ‚man' steht, ‚frau' (dazu) denken; der Verzicht auf diese Formen (ebenso wie auf den studentischen Plural (wieso eigentlich studentisch, müßte das nicht studentInisch oder studierendenisch heißen, konsequenterweise) ist *nicht* Ausdruck der Geringschätzung dieser Problematik, auch wenn es jetzt vielleicht so klingt, und der Autor zugibt nicht immer in der Lage zu sein, die Problematik so ernst zu nehmen, wie es viele andere tun; der Verzicht ist vielmehr Ausdruck der Tatsache, daß der Autor diese Formen nicht von Anfang an berücksichtigt hat und der Bequemlichkeit des Autors, die ihn daran hindert, jetzt nochmal alles daraufhin durchzusehen und zu ändern; der Autor wendet sich im Übrigen gegen die Einführung von ‚ders' als männlichen Artikel Plural (statt ‚die Männer' würde es dann ‚ders Männer' heißen)

2.2 Unterstützte Fraktaltypen

CHAOS*ultd* GEM ist modular aufgebaut – analog zu CHAOS*ultd* V5.0x, nur konsequenter, es gibt keine internen Berechnungsroutinen mehr. Das Programm besteht aus einem Hauptteil, der die Verwaltung der Bilder, das Anzeigen, Abspeichern etc. übernimmt, und mehreren Modulen, die Routinen zur Berechnung von Fraktalen bestimmter Typen zur Verfügung stellen. Maximal sind 32 Fraktal-Typen möglich (wobei eine Routine auch mehrere Typen umfassen kann).
Derzeit existieren Module für die folgenden Bildtypen:

- Fractale

  Unter Fractalen im engeren Sinn verstehe ich hier Mandelbrot-, Julia- und Pickover-Mengen, letztere mit frei definierbarer Berechnungsformel[2].

  Mandelbrot- und Julia-Mengen können auch in Pseudo 3d dargestellt werden.

  Das Modul Fractale umfaßt also genau die Bildtypen, die es schon in Fractal 4.3 (als einzige) und dann genauso in CHAOS*ultd* V5.0 gegeben hat.

- Hüpfer (nach Barry Martin)

- Feigenbaum-Diagramme

- L-Systeme

- XCHAOS

  XCHAOS ist ein Modul, das verschiedenen Berechungsroutinen gemeinsame Funktionen zur Parametereingabe etc. zur Verfügung stellt. Bei der Realisation einer solchen Berechungsroutine wird dann praktisch nur noch angegeben, welche Parameter gebraucht werden. Das ist zwar nicht so flexibel wie spezielle Eingaberoutinen, reduziert aber das Schreiben neuer Routinen im wesentlichen auf die eigentliche Zeichenroutine, so daß ich weniger zu tun habe und sich vielleicht auch mal andere daran wagen, Routinen für CHAOS*ultd* zu realisieren.

  Unter XCHAOS gibt es zur Zeit:

  - Barnsley-Fraktale
  - seltsame Attraktoren (Lorenz, Rössler, etc.)
  - Diffusion
  - Popcorn
  - Kreis[2]

- Iterierte Funktionen-Systeme, linear und nichtlinear

  Die Iterierten Funktionen-System sind bisher undokumentiert! Und zwar nicht, weil sie trivial zu bedienen sind, sondern weil ich nicht mehr mag (sorry). Wer weiß, was IFS sind (siehe z.B. einen recht guten Artikel in der ST-Computer 11/91) und seiner Intuition folgt, hat sicherlich gute Chancen, die Routinen zu verstehen, sie folgen gleichen Konzepten wie die anderen Routinen auch. Einziger weiterer Hinweis: bei den Kopierern (linear/nichtlinear) muß man nach dem Parameterdialog ein Bild-Objekt auswählen, das als Startbild verwendet wird. Man hüte sich ansonsten vor Speichermangel (da kann ich für nichts garantieren).

  Sogesehen sind die IFS-Routinen bislang *freak-only*!

  (Bitte keine Anfragen zu diesem Thema, ich schreibe die Anleitung schon noch irgendwann, und dann kann man weitersehen. Ich glaube es ist trotz dieses Anleitungsmangels nicht ganz unsinnig, die Routinen mitzugeben.)

3 Installation

3.1 Hardwarevorraussetzungen

CHAOS*ultd* GEM wurde auf einem Atari 520 ST$^+$ mit TOS1.4, 4 MByte Speicher und einer 85 MByte Festplatte entwickelt und getestet.
Das Programm sollte auf allen Atari ST, STE, TT, Falcon und Eagle Rechnern laufen – mangels Hardware wurde das Programm aber nur auf der oben genannten Konfiguration getestet.

[2] ich unterscheide in dieser Anleitung zwischen Fraktalen, als Oberbegriff für alle berechneten Bilder (selbst wenn sie gar nicht fraktal sind), und Fractalen im genannten engeren Sinn

CHAOS*ultd* GEM sollte mit jeder Auflösung und Graphikkarte zusammenarbeiten, solange ein VDI-Treiber existiert (mit Ausnahme von High-/True-Color-Auflösungen). Mindestauflösung ist allerdings 640x200 Punkte, wobei sich bei 640x200 Punkten (ST-Mid) ein paar kleinere Unschönheiten ergeben (siehe Fehler, 6.2).
Auf der Festplatte werden etwa vier- bis fünfhundert Kilobyte Platz benötigt und zwar für das Programm an sich. Prinzipiell kann man das Programm also auch ohne Festplatte auf Diskette verwenden. Die Datenmenge berechneter Bilder ist nach oben offen.
Speicher braucht das Programm natürlich auch, unter 1 MByte dürfte wenig gehen. CHAOS*ultd* GEM belegt – abhängig von der Zahl der Module die man verwendet – ca. 250-500 kByte Speicher für Programm, Resourcedateien etc.. Dazu kommt noch der Speicherbedarf für die Bilder. Minimum einige hundert Kilobyte, nach oben offen.
Zum Ausdrucken von Bildern gibt es eine Schnittstelle für ein externes Hardcopy-Modul. Bisher existiert nur ein Modul für 24 Nadel-Matrixdrucker.

3.2 Installation

Bei CHAOS*ultd* GEM von ‚Installieren' zu reden ist eigentlich eine Übertreibung[3].
Das Schlimmste hat man schon hinter sich, wenn man in der Lage ist, diesen Text zu lesen, nämlich das Auspacken der Archive. Hat man dies geschafft, dann kann man entweder *CHAOS.APP* gleich starten, oder aber – wenn einem der Platz, an dem das Programm jetzt steht, nicht gefällt – man kopiert das Programm *CHAOS.APP* und den Ordner *CHS_ULTD.GEM* in ein beliebiges Verzeichnis auf der Festplatte (oder auf eine Diskette, was aber natürlich weniger empfehlenswert ist).
Vielleicht noch kurz ein Wort zur Erklärung. Der Ordner *CHS_ULTD.GEM* enthält die Resource-Dateien, die Einstellungsdatei, die externen Module und im Ordner *HELP* die Hilfs-Texte für das Hilfs-System.
Die externen (Berechnungs-)Module besitzen die Dateikennung (Extension) *.XCH*, was natürlich für eXdended CHaos steht. Möchte man nicht alle externen Module verwenden, was ja auch Speicher kostet, so kann man die entsprechenden Module ohne weiteres löschen (nebst zugehöriger Resourcedatei; man sollte natürlich noch irgendwo eine vollständige Kopie haben) oder durch Ändern der Dateikennung deaktivieren.
Die richtige Zuordnung zwischen den Modulen und den zugehörigen Dateinamen wird dem geneigten Leser als Übungsaufgabe überlassen.
Das Hardcopy-Modul besitzt die Kennung *.HCP*.

3.3 Lieferumfang

CHAOS*ultd* GEM V6.40 umfasst die folgenden Dateien und Ordner (nach vollständigem Entpacken aller Archive; hat man CHAOS*ultd* via Internet bezogen, so hat man unter Umständen nicht alle Dateien, wenn man sich nicht alle Archive geholt hat (was ja durchaus sinnvoll sein kann, vgl. auch *README.TXT*):

`CHS_ULTD.APP`	das Programm
`READ_ME.TXT`	
`GRRRRRRR.TXT`	READ IT!
`CHANGES.TXT`	aktuelle Änderungen
`CHNG_ALT.TXT`	ältere Änderungen (in der Anleitung enthalten)
`ERRORS.TXT`	vorhandene Fehler
`CHAOS.FAQ`	Fragen und Antworten zu CHAOS*ultd*
`*.TXT`	was ich womöglich sonst noch zu sagen habe
`CHS_ULTD.GEM\`	
`CHS_ULTD.RSC`	die Resourcedatei
`CHS_ULTD.SET`	die Einstellungen
`24NADEL.HCP`	die Hardcopy-Routine
	und diverse Module
`FRACTAL.XCH`	
`HUEPFER.XCH`	
`IFS.XCH`	
`LSYSTEM.XCH`	
`FEIGENBM.XCH`	
`XCHAOS.XCH`	
`*.XCH`	neue Module?

[3] auf PCs ist das Installieren von Programmen ein beliebtes Unterhaltungsspiel für DiskJockeys nach dem Motto, sie haben ein Dutzend Disketten und schieben die in bestimmter Reihenfolge in das Laufwerk; bei größeren Programmen hat man dann wieder nur eine Scheibe, dafür aber ein CD-Rom, dessen Inhalt die Festplatte auch problemlos (über)füllen kann

	sowie deren Resourcen
24NADEL.RSC	
FRACTAL.RSC	
HUEPFER.RSC	
IFS.RSC	
LSYSTEM.RSC	
FEIGENBM.RSC	
XCHAOS.RSC	
*.RSC	neue Resource-Files?
HELP\	
*.HLP	die Hilfstexte
ANLEITNG\	
CHS_ULTD.DVI, -.TXT, -.PS	die Anleitung im TEX , ASCII und Postscript-Format
BEISPIEL\	Ordner mit Beispielen (unberechnet)
*.CHS	
PARAM\	Ordner mit Parameterdateien
*.L	für L-Systeme
*.IFS	und IFS
CHAOS_PG\	Ornder mit der Entwicklerdokumentation

4 Danksagung

CHAOS*ultd* GEM V6.40 wurde im wesentlichen mit Hilfe folgender Programme entwickelt:

- Pure C 1.0, PureDebugger
- PASM Assembler
- Interface Resource Construction Set
- Tempus Editor
- Gemini (als Grundlage für alles andere)
- Lindner-TEX (für diese Anleitung)

CHAOS*ultd* GEM verwendet MyDials von Olaf Meisiek (Interface) mit dem NKCC Tastaturhandler von Harald Siegmund.

Koautor (in spe): Renè Fritz
(Farbselektor, Farbrotation)

besonderen Dank an

- die Programmierer von FractInt:

  the Stone Soup Group:

  Main authors (in historical order):

  Bert Tyler
  Timothy Wegner
  Mark Peterson
  Pieter Branderhorst

  Contributing authors (in alphabetic order);

  Joseph A Albrecht, Kevin C Allen, Rob Beyer, John W. Bridges, Michael D. Burkey, Robin Bussell, Lee Daniel Crocker, Monte Davis, David Guenther, Michael L. Kaufman, Wesley Loewer, **Adrian Mariano**, Joe McLain, Bob Montgomery, Roy Murphy, Ethan Nagel, Jonathan Osuch, Marc Reinig, Prof. JM Richard-Collard, Lee H. Skinner, Dean Souleles Chris J Lusby Taylor, Scott Taylor, Paul Varner, Phil Wilson

(ich hoffe, ich habe nicht zufällig einen vergessen)

FractInt ist ein PC-Programm zur Erzeugung von Fractalen, das vor allem durch die Vielzahl möglicher Fractal-Typen und die große Rechengeschwindigkeit besticht.
Leider gibt es für den ST nur eine partielle Implementation der Version 12, die nochdazu sehr instabil ist (beim PC ist FractInt nunmehr bei Version 18 angelangt).
Angesichts erheblicher Mengen an 80(x)xx-Assembler-Code läßt sich das Programm auch nur sehr schwer auf den ST umsetzen. (das manuelle Übersetzen von 80(x)xx Code in 68000 Code ist grausig, besonders, wenn man 80(x)xx Assembler überhaupt nicht beherrscht)
Allerdings muß es ja nicht gleich das ganze Programm sein, so diente FractInt (dessen Sourcen frei und frei zugänglich sind) als Quelle zweier Fractaltypen für CHAOSultd. (FractInt stand ja schon Pate bei Barnsley-Fractalen und den Attraktor-Routinen, bei den L-Systemen und der Diffusion habe ich auch auf den Code von FractInt zurückgegriffen, weshalb die Autoren der Original-Routinen auch im Programminfo als Co-Autoren erscheinen).

- Michael G.

  für die Fließkommaroutinen der freidefinierten Fractale
  und einige (in früheren Versionen) gefundene Fehler

- Wolfgang Ley

  für die Betreuung des FTP-Servers in Clausthal

- Stefan Haake

  für die Betreuung des FTP-Servers in Stuttgart

5 Entwicklungsstand

5.1 Änderungen gegenüber früheren Versionen

siehe hierzu die Dateien *CHNG_ALT.TXT* und *CHANGES.TXT*.
CHNG_ALT.TXT enthält alle Änderungen der verschiedenen Versionen von V6.31 bis zurück zu V6.00. Alle diese Änderungen sind in dieser Version der Anleitung berücksichtigt.
Die Datei *CHANGES.TXT* enthält gegebenfalls alle Änderungen, die über den Stand dieser Anleitung hinausgehen.

5.2 Helfer gesucht!

CHAOS*ultd* GEM ist nicht fertig!
Es wird auch nie fertig werden; natürlich kann es irgendwann passieren, daß ich keine Lust mehr habe, an CHAOS*ultd* weiterzuarbeiten[4].
Ich denke allerdings, daß die Entwicklung des Hauptprogrammes inzwischen weitgehend abgeschlossen ist, das heißt, ich plane momentan keine größeren Erweiterungen mehr. Was noch bleibt ist das Entfernen von Fehlern, die sicher immer wieder auftauchen werden, sowie verschiedene interne Bereinigungen und Optimierungen. In diesem Zusammenhang wird es wohl auch noch zu kleineren Änderungen an Funktionen kommen.
Was die externen Routinen angeht, so sind dort noch einige Macken zu beseitigen und Funktionen zu ergänzen.
Daneben gäbe es natürlich noch jede Menge Fraktal- und andere Bildtypen, für die man Berechnungsroutinen schreiben könnte.
Die Entwicklung an CHAOS*ultd* geht also weiter. Leider fehlt mir die Zeit, auch nur das zu realisieren, was unmittelbar ansteht. An neue Routinen ist momentan nicht zu denken (ich brauche einfach mal wieder eine Rechner-Pause). Insbesondere sollte ich mich eigentlich mit Quarks, Mesonen und Baryonen im NJL-Modell befassen, um endlich Diplom-Physiker zu werden.
Sollte irgendwer daran interessiert sein, an CHAOS*ultd* V6.xx mitzuwirken, so ist er (oder sie) herzlich eingeladen. Am einfachsten ist natürlich die Entwicklung neuer Berechnungsroutinen, da man hier eine definierte Schnittstelle hat. Aber auch eine Übertragung des Progammes in eine andere Sprache wäre denkbar, die Text-Strings des Hauptprogrammes sind bereits weitgehend in eine eigene Datei ausgelagert – bei den Berechnungsroutinen muß ich dies noch machen, aber das ist kein prinzipielles Problem – der Rest der Texte (eigentlich der Hauptteil) steht wie es sich gehört in der Resource-Datei.
Hier stellt man natürlich gleich fest, so einfach ist es auch wieder nicht, Programmieren und zwar in C sollte man schon können. Außerdem sollte man über Pure C (andere C Compiler setzen noch mehr Programmiererfahrung

[4] nein, die Jammerei, daß ich keine Resonanz auf CHAOS*ultd* bekomme werde ich jetzt nicht wiederholen (ausnahmsweise)

voraus, weil man dann die Kompatibilität testen muß und die ist schlecht, weil Pure C Parameter in Registern übergibt) und Interface (RCS) verfügen. Sind diese Vorraussetzungen gegeben, ist das Angebot ernst gemeint. Nach der überragenden Resonanz auf CHAOS*ultd* V5.0x (zwei Anrufer, die mit dem selbstenpackenden Archiv nicht zurechtkamen) habe ich allerdings kaum noch Illusionen.
Trotzdem ist die Schnittstelle mittlerweile dokumentiert. Die Dokumentation findet sich – nebst Beispielen und der nötigen Header-Datei – in der Entwicklerdokumentation zu CHAOS*ultd*; diese findet man entweder in einem eigenen Archiv *CHAOS_PG.LZH* (Internet) oder im selbstentpackenden Archiv *CHAOS_PG.TOS*.
Sollte jemand tatsächlich Probleme haben, interessante neue Fraktal-Typen für CHAOS*ultd* zu finden, dann sollte er – (pardon) sie – sich mal FractInt anschauen, ein PC (leider) Programm, das jede Menge Fraktal-Typen bietet.

5.3 Updates und Bezugsmöglichkeiten

Updates wird es also sicher geben – irgendwann.
Ich sehe mich außerstande, selbst Updates zu vertreiben, auch nicht gegen Leerdiskette und frankierten Rückumschlag, ich habe schon genug zu tun – sorry. Ausnahmen mache ich natürlich bei Leuten, die sich aktiv um CHAOS*ultd* bemühen, sei es durch Fehlermeldungen, Routinen, verbesserte Dokumentation (Hilfstexte) usw.
Wie also kommt man an eventuelle neue Versionen von CHAOS*ultd*.
Erste Möglichkeit sind PD-Versender, ich gehe davon aus, daß CHAOS*ultd* GEM wie CHAOS*ultd* V5.0x Aufnahme in die ST-Computer PD-Sammlung findet, und werde dort auch Updates abliefern.
Günstiger ist die zweite von mir selbst unterstützte Möglichkeit, CHAOS*ultd* GEM und kommende Updates via Internet zu beziehen. Günstiger, weil es nichts kostet und weil Updates dort wesentlich schneller verfügbar sein werden (V6.01 ist noch immer nicht in der PD-Sammlung veröffentlicht). Internet ist ein Internationales Netzwerk, an das bespielsweise die Universitäten angeschlossen sind, und in dem unter anderem FTP (File-Transfer-Protocoll) - Server existieren, von denen man PD-Software beziehen kann.
Ich selbst werde CHAOS*ultd* GEM wie auch Updates auf den Servern der Uni Stuttgart (ftp.uni-stuttgart.de) und der TU Clausthal (ftp.tu-clausthal.de) deponieren.[5] In Stuttgart findet man das Programm (bisher) im Verzeichnis *SOFT/ATARI/APPLICATIONS/MATH* in Clausthal in */PUB/ATARI/GRAPHICS/FRACTALS*.
Als dritte Möglichkeit böte sich die Verteilung via Mailboxen (z.B. Maus-Netz) an, da ich aber kein Modem besitze, kann ich hier nur dazu auffordern, daß irgendwer (oder die Betreiber der Boxen) das Programm dort ablegt.

6 Fehler (und was man dafür halten könnte)

„Programme laufen nicht“ lautet bekanntlich eine mögliche Formulierung von Murphy's Gesetz für Programmierer. Nun am Ende der Programm-Entwicklung hat man es meist geschafft, diesen Satz im konkreten Einzelfall zu widerlegen (oder man hat kein Programm und braucht keine Anleitung zu schreiben) – jedenfalls mehr oder weniger. So ist es auch hier.
Andererseits gibt es eine weitere, schwächere Folgerung aus Murphy's Gesetz die besagt, daß jedes nicht-triviale Programm mindestens einen Fehler enthält. Mit an Sicherheit grenzender Wahrscheinlichkeit trifft dies auch hier zu. Bevor Sie jetzt entsetzt das Programm wegwerfen, sollten Sie sich überlegen, daß Sie das dann konsequenterweise mit allen anderen Programmen (einschließlich des Betriebssystems) auch tun müßten.
Ich habe CHAOS*ultd* soweit möglich getestet[6] und es läuft. Das Programm ist soweit möglich fehlertolerant, sollte also auch bei Fehlbedienung nicht gleich schlappmachen[7].

6.1 eliminierte Fehler

siehe hierzu ebenfalls die Dateien *CHANGES.TXT* und *CHNG_ALT.TXT*.

6.2 bekannte Fehler

Einige kleinere Macken, die sich nicht so einfach beheben lassen, sind mir schon bekannt:

[5] Die beiden Server habe ich ausgewählt, weil Stuttgart eine der schnellsten Verbindungen nach Regensburg hat, und weil Clausthal das – soweit mir bekannt – bestorganisierte PD-Angebot für Atari hat, sowohl was den Umfang betrifft, vor allem aber wegen der Kommentierung der Programme – an dieser Stelle herzlichen Dank an die Betreuer der beiden Server Stefan Haake (Stuttgart) und Wolfgang Ley (Clausthal).
P.S. Suttgart mußte zuletzt auf Updates verzichten, weil der File-Transfer irgendwie gestört war; jedenfalls ist mir der Rechner, von dem aus ich übertragen habe nach einiger Zeit abgestürzt (natürlich ein DOSen-Rechner). Ich hoffe aber, daß ich das demnächst doch auch noch hinkriege, und dann liegt auch in Stuttgart wieder eine neue Version.

[6] naja man sollte das nicht zu wörtlich nehmen, die weiteren Tests könnt ja jetzt ihr übernehmen

[7] allerdings ist es bei Parametern von Berechnungsroutinen teilweise möglich ist, unsinnige Angaben zu machen, die nicht von der Eingaberoutine zurückgewiesen werden, dadurch aber auch nicht sinnvoll werden

- Startet man CHAOS*ultd* in mittlerer-ST-Auflösung (bzw. irgendeiner Auflösung die einen 8x8-Font verwendet), so gibt es drei kleinere Probleme mit den Dialogboxen.

  Das kleinste liegt im Programminfo; die erste Dialogbox, in der Bit-Image-Objekte angezeigt werden gerät etwas durcheinander. Aber die dient eh nur meiner Angeberei und ist nicht so wichtig.

  Die beiden anderen Probleme befinden sich in der Anzeige der Farbcodierung von Fractalen (in der Haupteingabe-Dialogbox für Fractalparameter) und in der Anzeige der Trafos linearer IFS in der Haupteingabe-Dialogbox für IFS-Parameter (linear). In beiden Fällen werden die Informationen nicht korrekt angezeigt, entweder überdecken sich die Zeilen partiell oder einzelne Zeilen werden ganz verdeckt. Es ist leider nicht so viel Platz in den beiden Dialogen vorhanden, um dieses Problem zu beseitigen. (Die Daten werden aber korrekt verwaltet und können bei Aufruf der Unter-Dialoge für ihre Eingabe gesichtet werden.)

  Dies gilt gegebenenfalls auch an anderen Stellen, an denen Ausgaben im kleinen Zeichensatz gemacht werden.

- wird die (angezeigte) Berechung eines Bildes für ein Berechnungsinfo unterbrochen (`Space`-Taste) und kann für den Info-Dialog der Hintergrund nicht gespeichert werden, so wird das Bild danach *nicht* wieder angezeigt sondern der Bildschirm gelöscht. Die korrekte Erzeugung des Bildes ist davon nicht betroffen, nur die unmittelbare Ausgabe auf den Bildschirm während des Berechnens.

- bereits seit V5.0x gibt es einen Fehler bei den frei-definierten Fractalen, der auch in CHAOS*ultd* GEM V6.40 erhalten blieb (obwohl das jetzt eigentlich kein Fehler in CHAOS*ultd* sondern einer in einer der Berechnungsroutinen ist):

  Berechnet man freie Fractale und verwendet in der Abbruchbedingung die 'xor'-Verknüpfung, so erhält man (total) unterschiedliche Ergebnisse, je nachdem ob man mit der schnellen oder der exakten Berechnung arbeitet.

  This is *not* a feature, this is a bug!

  Das Ganze hat *vermutlich* folgenden Hintergrund: die Routinen zur Berechnung freier Fractale ignorieren Fließkommafehler jeglicher Art, insbesondere auch Überläufe[8].

  Verwendet man nun die 'xor'-Verknüpfung, so wird die Berechnung nicht abgebrochen, falls beide Bedingungen erfüllt sind[9], so daß es leicht zu einem Überlauf kommen kann (das kann man sich im Debugger auch anschauen). Anscheinend reagieren die beiden verschiedenen Fließkomma-Routinen nun bei und nach dem Überlauf anders, so daß sich verschiedene Resultate ergeben. Dabei ist das Ergebnis der schnellen Berechnungsroutine dasjenige, das (logischen Überlegungen folgend, d.h. vielleicht) richtiger ist – richtiger insofern, als hier trotz der Fließkomma-Überläufe die Punkte als unendlich tief erscheinen.

Weitere bekannte Fehler sind gegebenenfalls in der Datei *ERRORS.TXT* beschrieben.

6.3 unbekannte Fehler

gibt es nach obigen allgemeinen Überlegungen sicherlich jede Menge. Ich hoffe, es treten wenigstens keine schwerwiegenden Fehler mehr auf.
Kommt es bei der Benutzung von CHAOS*ultd* zu irgendwelchen kritischen Situationen so empfehle ich die Bemerkung in der rechten oberen Bildschirmecke zu beherzigen. Sie wurde unter anderem zu diesem Zweck dort installiert[10].
Wer einen Fehler findet, darf ihn zunächst einmal behalten.
Sollte der Fehler reproduzierbar sein, so wäre ich darüberhinaus für eine ausführliche Beschreibung des Fehlers und seiner Erzeugung dankbar (behalten muß man ihn natürlich trotzdem erst mal, es besteht dann aber eine Chance, daß ich ihn eliminiere).

6.4 Fehler in der Anleitung

Die Anleitung soll das Programm beschreiben. Tut sie das an irgendeiner Stelle nicht, so irrt die Anleitung und nicht das Programm[11].

Für Druk-Feeler in Anleitung gilt die Devise:

[8] als ich die Routinen geschrieben habe, bin ich davon ausgegangen, daß keine Fließkommafehler auftreten; eine aufwendige Behandlung der Fließkommafehler schien also überflüssig und würde nur Zeit kosten (beim Entwickeln der Routine wie beim Berechnen der Bilder); sie jetzt einzufügen ist mir im Moment zu viel Arbeit

[9] wenn man an dieser Stelle über den Sinn der Abbruchbedingung nachdenkt wird einem natürlich klar, daß 'xor' hier a priori nicht besonders sinnvoll ist, aber der Vollständigkeit gehört es halt auch dazu

[10] darüberhinaus soll diese Bemerkung natürlich zur Verbreitung von CHAOS*ultd* beitragen, D. Adams zufolge ist sie dazu bestens geeignet

[11] an sich wäre es ja umgekehrt auch ganz nett, aber das Risiko ist mir zu groß; die Anleitung anzupassen ist in jedem Fall das kleinere Übel

This is *not* a bug, this is a feature.
Druckfehler in der Anleitung dienen der Auflockerung des Textes und sollen insbesondere denen, die sonst nichts verstehen, Gelegenheit zu Erfolgserlebnissen geben, wenn sie welche finden. Insbesondere fehlen wahrscheinlich jede Menge Kommata.
Der Autor behält sich darüberhinaus von Rechtschreibregeln abweichende (also falsche) Schreibweisen vor, soweit die Worterkennung dadurch nicht beeinträchtigt wird[12].
Wiederholungen in der Anleitung mögen den einen oder anderen aufmerksamen Leser (gibt's solche etwa) verwundern, sie entstanden einfach, weil ich die Anleitung nicht von vorn bis hinten geschrieben habe, und man eigentlich alles überall und auf einmal erklären müßte.

7 zu dieser Anleitung

Real programmers don't read manuals heißt es, und ich mache das häufig genauso.
Dennoch möchte ich an dieser Stelle (was jetzt erst, nach so vielen Seiten) jedem die Lektüre der Anleitung nahelegen (auch wenn sie recht lang geworden ist). Man sollte sie wenigstens mal überflogen haben, da das Programm zwar mehr oder weniger konsequent in GEM eingebunden ist, viele der Möglichkeiten und Details des Programmes aber *nicht* sofort ersichtlich sind. (Insbesondere verletzte ich mit Vorliebe GEM-Benutzerführungs-Richtlinien, belege Menüs doppelt und dreifach usw.).
Die Anleitung entstand unter Verwendung von Teilen der Anleitung von CHAOS*ultd* V5.0x, teilweise wörtlich wie etwa der letzte oder der folgende Absatz. Wer CHAOS*ultd* V5.0x kennt, kann die Anleitung sicherlich sehr viel schneller lesen, sollte die Anleitung aber trotzdem durchsehen. Den letzte Teil der Anleitung von V5.0x – Dateiformate und Schnittstelle für die externen Routinen – gibt es jetzt als eigenen Text, siehe Abschnitt 5.2. Außerdem wurde die Anleitung ursprünglich für Version 6.00 geschrieben und dann für Version 6.10 und jetzt V6.31 überarbeitet (wer also schon mit V6.01 oder V6.10 gearbeitet hat, muß nicht die ganze Anleitung lesen, für den ist die Datei *CHNG_ALT.TXT* sicher günstiger.
Im Gegensatz zu früheren Versionen ist diese Version papiersparend gesetzt (Leser der Ascii-Version merken davon natürlich nichts). Ich hoffe, das wirkt sich nicht zu negativ auf die Lesbarkeit aus.

Die Anleitung ist – entsprechend der Modularität des Programmes – in mehrere Abschnitte aufgeteilt. Im ersten Abschnitt wird lediglich die allgemeine Benutzung des Verwaltungsteiles, im wesentlichen ohne konkreten Bezug auf irgendwelche Berechnungsroutinen, beschrieben. Die Beschreibung der Berechnungsroutinen für Fractale, Hüpfer, Feigenbaumdiagramme etc. folgt dann in den weiteren Abschnitten.
Bei der Beschreibung von CHAOS*ultd* gehe ich im folgenden in etwa in der Reihenfolge vor, wie man sie auch bei der Benutzung des Programmes gebraucht. Da keine konkreten Bilderzeugungsroutinen beschrieben werden, sollte man eventuell parallel entsprechende Abschnitte der folgenden Teile lesen, wobei die Fractal-Routinen wohl am ausführlichsten beschrieben sind. Bei den anderen Routinen wird ihre Kenntnis teilweise vorausgesetzt.
Auch ist es natürlich sinnvoll, die diversen Möglichkeiten gleich auszuprobieren; möglicherweise wird erst an Hand des Programmes selbst, klar, wie die eine oder andere Bemerkung in der Anleitung gemeint ist (sorry, aber noch ausführlicher kann ich das Programm wirklich nicht beschreiben).

Wird in der Anleitung auf Menüeinträge oder Texte von Button verwiesen, so sind diese in Sans Serif gesetzt. Tasten erscheinen in `Typewriter`, auch *DATEIEN* werden hervorgehoben. (Leser der Ascii-Version dieser Anleitung haben davon leider garnichts.)

Weiterführende Literatur

Die zugrundeliegenden Algorithmen und die sich daraus ergebenden Formen werden nur ansatzweise beschrieben. Ich möchte deshalb hier auf die einschlägige Literatur verweisen:
die meisten Anregungen habe ich den Sonderheften *Computerkurzweil I-IV* der Zeitschrift *Spektrum der Wissenschaft* entnommen. Sie sind zwar soweit ich weiß einzeln nicht mehr erhältlich, dafür gemeinsam in Buchform. Dort finden sich auch weitere Literaturangaben.
In einem Artikel in der ST-Computer (11/91) über Iterierte Funktionensysteme wird unter anderem noch auf die folgenden beiden Werke verwiesen:
H.-O. Peitgen, D. Saupe (Hrsg), The Science of Fractal Images, Springer Verlag 1988
Benoit B. Mandelbrot, Die fractale Geometrie der Natur, Birkhäuser 1987

[12] ich schreibe Standart weiter mit t auch wenns noch so falsch ist. (Doof, nicht?)

Teil I
CHAOS*ultd* GEM

1 Allgemeine Bedienungshinweise

1.1 Hinweise zum Betrieb

Es ist wenig ratsam, CHAOS*ultd* GEM auf einem 8Mhz ST ohne Bildschirmbeschleuniger einzusetzen. (Es ist überhaupt wenig ratsam, einen ST ohne Bildschirmbeschleuniger zu betreiben.) Wer es doch tut ist selber schuld und muß eben geduldig sein. Ich selbst verwende NVDI, und mußte (leider) CHAOS*ultd* speziell daran anpassen, da NVDI (V2.01) offenbar das Arbeiten mit Bitmaps, die mehr Bildebenen besitzen als der Bildschirm, nicht mag.
Man sollte darüberhinaus stets sicherstellen, daß dem System noch Speicher übrigbleibt (kann man in den Einstellungen eintragen), da die Dialogboxen dann den Bildschirmhintergrund sichern können und dieser nach Verlassen der Dialogbox nicht neu gezeichnet werden muß.
Wenn man sehr große Bilder (z.B. 3200x2000 Punkte in 2 Farben) erzeugt, dann kann schon das bloße Anzeigen von Bildern mehrere Sekunden vom Aufrufen der Funktion bis zum Öffnen des Fensters (oder Umschalten des Bildschirms) dauern. Schließlich muß das Bild erst entpackt werden etc.. Um solche Aktionen anzuzeigen verwendet CHAOS*ultd* neben Info-Dialogen (beim Laden, Speichern, Drucken, Konvertieren und (verdeckten) Berechnen) auch den Mauszeiger: beim Laden und Speichern wird eine stilisierte Diskette, beim Drucken ein (weniger gut erkennbarer) Drucker, beim Berechnen ein Taschenrechner, beim Holen und Ablegen von Bildern eine Uhr und beim Konvertieren ein Rechteck mit Graustufen angezeigt. In allen diesen Fällen ist mehr oder weniger viel Geduld nötig.
Generell ist es ratsam, erstmal kleinere Bilder zu erzeugen und dann (eventuell) schrittweise zu großen Bildern überzugehen. Dabei sollte man stets bedenken, daß der Speicherbedarf und (meist) auch der Rechenaufwand mit der Fläche des Bildes (mal Farbtiefe) und nicht mit der Seitenlänge geht (d.h. ein Bild der Größe 1280x800 ist viermal so groß wie 640x400 und nicht nur doppelt) – eigentlich trivial, aber man macht sich das oft nicht klar.

1.2 Booten des Programmes

Wird CHAOS*ultd* GEM gestartet[13], so wird zunächst die (Haupt-)Resource-Datei geladen, dann die Routinen; diese werden initialisiert, wobei sie ihre eigenen Resource-Dateien laden. Schließlich werden noch die Voreinstellungen geladen, die Menüleiste angezeigt und das Objekt-Fenster geöffnet.
Während des Ladens der Module wird eine Dialogbox angzeigt, in der dargestellt wird, welches Modul gerade geladen wird und welche Fraktal-Typen zur Verfügung gestellt werden. Scheitert die Initialisierung einer Routine, so wird diese gesperrt, was in der Dialogbox durch helle Schrift angezeigt wird; solche Routinen können nicht aufgerufen werden, bleiben aber im Speicher stehen (man sollte also die Ursache der Fehlinitialisierung beseitigen oder die Routine weglassen).

1.3 Speicherverwaltung

CHAOS*ultd* besitzt eine eigene Speicherverwaltung, zu der es ein paar Dinge zu erläutern gibt. Seit V6.40 alloziert CHAOS*ultd* nicht mehr gleich nach dem Booten den gesamten für CHAOS*ultd* anschließend zur Verfügung stehenden Speicher, sondern ist in der Lage mehrere Speicherblöcke zu verwalten. Diese Möglichkeit wurde im Hinblick auf die Verwendung von CHAOS*ultd* in Multitasking-Systemen (siehe auch 1.5) geschaffen. Hier wird jetzt nur noch (in etwa) soviel Speicher reserviert wie CHAOS*ultd* gerade tatsächlich benötigt und der Restspeicher steht dem System und anderen Programmen zur Verfügung.
Die neue Speicherverwaltung ist andererseits natürlich ein bißchen komplizierter als die alte und man sollte ein paar Dinge beachten:
In den Einstellungen gibt man neben der Größe des Speichers, den CHAOS*ultd* beim Belegen von Speicher frei lassen soll (Angabe jetzt in Kilobyte) die Größe des Speicherseiten an. Änderungen (beider Angaben) wirken sich direkt auf die Reservierung *neuer* Speicherseiten aus.
Speicher frei lassen sollte man so wie bisher mindestens in der Größenordnung des Bildschirmspeichers.
Die Größe der Speicherseiten hängt von mehreren Faktoren ab. Man sollte sie nicht zu klein wählen, da sonst die Speicherfraktionierung begünstigt wird (der Speicher zerfällt in viele kleine aber nicht zusammenhängende Blöcke, große Blöcke können nicht mehr belegt werden). Möchte man den Restspeicher nutzen (für andere unter Multitasking laufende Programme) sollte man sie aber auch nicht zu groß wählen.

[13] wie man dies macht, ist nicht Gegenstand dieser Anleitung, wer nicht weiß, wie man ein Programm startet, wird vermutlich auch nicht mehr aus CHAOS*ultd* herausfinden, da auch das Beenden des Programmes nicht Gegenstand dieser Anleitung ist. Ich will aber nicht schuld sein, wenn jemand bis zu seinem Lebensende mein Programm laufen lassen muß.

Des weiteren ist zu beachten, daß die Objekte alle in der ersten Speicherseite (deren Größe bei Programmstart festgelegt wird und nicht mehr geändert werden kann) untergebracht werden müssen, wobei ein Objekt ca. 100 Byte belegt. Dies begrenzt unter Umständen die Zahl der möglichen Objekte auf weniger als die sonst theoretisch möglichen 9999.
Als Faustregel kann man wohl sagen, daß Single-Tasking-Nutzer, so sie nicht irgendwelche Accessories parallel zu CHAOS*ultd* nutzen, die viel Speicher brauchen, am besten eine maximale Größe angeben (z.B. 16000 Kilobyte). Es wird dann *eine* Speicherseite angelegt, deren Größe durch den freien Speicher begrenzt ist. CHAOS*ultd* verhält sich dann (fast) so wie mit der alten Speicherverwaltung.
Unter Multitasking gilt es einen Kompromiß zu schließen, wobei man sich auch an der Größe der zu verwaltenden Speicherblöcke, die wiederum durch die Größe der Bilder bestimmt ist, orientieren sollte. Bei Bildgrößen um 32 Kilobyte (ST-Screen-Größen) sind Speicherseitengrößen von 128-256 Kilobyte brauchbar, berechnet man ständig größere Bilder, so sollte man die Seitengröße größer wählen.
Generell kann CHAOS*ultd* auch Speicherblöcke verwalten, die größer als die Seitengröße sind, diese werden direkt vom TOS angefordert. Damit begünstigt man aber wieder die Speicherfraktionierung.
Noch eine Einschränkung zur Objektverwaltung: alle Objekte werden wie gesagt in der ersten Speicherseite verwaltet. Ist dort kein Speicher mehr frei, und kann dort kein freier Speicher geschaffen werden, indem Speicherblöcke in andere Speicherseiten verschoben werden, so sind keine Objekte mehr möglich. (Dies kann auch dann passieren, wenn nach freier Speicher angezeigt wird!).
Im Gegensatz zur früheren Version der Speicherverwaltung werden Garbage-Collections nicht mehr durch Gongs angezeigt (sie treten jetzt auch viel häufiger auf).

Speicherinfo

Der belegte und (intern) freie Speicher wird in der Info-Zeile des Objektfensters angezeigt (s.u.).
Ein detailiertes Speicherinfo kann man mit dem Menüpunkt Speicher-Info im Optionen-Menü aufrufen.
Angezeigt wird,
wieviel Speicher von CHAOS*ultd* belegt wurde und zwar zum einen in Speicherseiten und zum anderen direkt (für Blöcke, die größer als eine Speicherseite sind).
wieviel Speicher in den Speicherseiten noch frei ist.
wieviel Speicher von den Objekten belegt wird und zwar aufgeteilt nach Speicher für die Objektstrukturen selbst, für Parameter (bei Filmen: Filmdaten), für zusätzliche Parameter und für Bilddaten (incl. Farbtabellen). Außerdem wird die Summe dieser Größen angegeben. Angeben werden immer die Daten für alle und alle selektierten Objekte. Bei den Objekten wird zunächst deren Anzahl angeben; in Klammern wird angezeigt, für wieviele Objekte tatsächlich Speicher reserviert ist (CHAOS*ultd* hält meist Platz für neue Objekte bereit, da CHAOS*ultd* den Objektstrukturspeicher in 16er-Blöcken reserviert und beim Freigeben einen 16er-Block für neue Objekte zurückhält). Die Angabe des Speichers für die selektierten Objekte entfällt aus Platzgründen. Bei den anderen Angaben wird zunächst die Summe der Speicherblöcke aller bzw. aller selektierten Objekte, dann die durchnittliche Größe angegeben.
wieviel Speicher ansonsten belegt ist. Hierunter fällt der für Anzeigefenster belegte Speicher ebenso, wie verschiedene Datenpuffer, die nicht unbedingt benötigt werden (Beispielsweise eine Quadrattabelle, die die Fractalroutinen zur Berechnung anlegen, und die so nicht jedesmal neu berechnet werden muß).
der freie Speicher und zwar sowohl der innerhalb der belegten Speicherseiten wie der des Systems (Summe *aller* Speicherblöcke des Systems!).
Die Box kann man mit drei Button verlassen: Ok verläßt die Box ohne weitere Folgen, GarbColl verläßt die Box und löst dabei eine GarbageCollection (GC) aus, das heißt alle (intern leeren) Speicherblöcke werden zusammengefaßt. (Vorwiegend für Testzwecke vorhanden, eigentlich nicht nötig. CHAOSultd führt eine GC wenn nötig von selbst aus) freigeben verläßt die Box, gibt Speicher frei, in dem (entbehrliche) Daten gepuffert sind (solche Daten würden unter sonstiges mitgezählt) und löst eine GC aus (der Aufruf ist ebenfalls nicht nötig, solcher Pufferspeicher wird bei Speichermangel auch automatisch freigeben[14]).

1.4 Dialoge

CHAOS*ultd* verwendet FlyDial analoge MyDials (von Olaf Meisiek (Interface) mit dem NKCC Tastaturhandler von Harald Siegmund), die gegenüber herkömlichen GEM-Dialoge ettliche Vorteile bieten. Auch wenn solche Dialogboxen mittlerweile recht bekannt sein dürften – sie finden schließlich immer öfter Verwendung – sollen diese Vorteile hier nochmals kurz erläutert werden (ohne Anspruch auf Vollständigkeit).
'Fliegen' können die Dialoge durch das Eselsohr rechts oben (nur wenn der Speicher zum Sichern des Hintergrundes gereicht hat, sonst fehlt das Eselsohr). Klickt man es an, so kann man den Dialog auf dem Bildschirm verschieben. Dies lohnt sich vor allem auf Großbildschirmen, oder (gelegentlich) wenn der Dialog etwas verdeckt (wobei dann aber meist

[14] mit einer Einschränkung: zur Schaffung von Platz für Objekte können solche Speicherblöcke nicht freigegeben werden, wenn sie in der ersten Speicherseite liegen

kein Platz ist, den Dialog weit genug wegzuschieben). Mit Shift-Klick auf das Eselsohr wird der Dialog 'durchsichtig', d.h. es wird nur noch ein Rahmen angezeigt, den man verschieben kann. Man kann einstellen, ob Dialoge jedesmal beim Aufrufen zentriert werden sollen oder ob sie an der alten Stelle erscheinen sollen (vgl. Einstellungen, 6.1).
Des weiteren sind MyDials weitgehend tastaturbedienbar. Viele Button kann man anwählen, indem man den in ihnen unterstrichenen Buchstaben zusammen mit der `Alternate`-Taste drückt.
Der Abbruch-Button (oder ein anderer der Exit-Button, auch Ok, wenn das der einzige Exit-Button ist) kann auch mit `Undo` ausgelöst werden.
Button zum Selektieren einzelner Optionen erscheinen mit vorangestelltem Quadrat, das im selektierten Zustand angekreuzt wird, 'Radiobutton' also Button von denen immer nur einer selektiert sein kann erhalten einen vorangestellten Kreis, der beim selektierten Button ausgefüllt wird. Das Ganze ist irgendwie Mac-ähnlich und sieht jedenfalls im Vergleich zu herkömmlichen ST-Dialogen sehr edel aus.
In Text-Feldern hat man die zusätzlichen Möglichkeiten mit `Shift-Cursor links` ans den Anfang, mit `Shift-Cursor rechts` an das Ende des Text-Feldes und mit `Control-Cursor links` bzw. `rechts` Wortweise nach links und rechts zu springen.
Im Prinzip kann man auch noch den Cursor in Text-Feldern mit der Maus setzen, dies funktioniert aber wegen eines Fehlers in den Routinen nicht richtig, wenn das Text-Feld außer dem edierbaren Eintrag noch einen Maskentext enthält.
Dafür hat man in allen Text-Feldern, in denen beliebige Zeichen eingegeben werden können (also nicht nur alphanumerische Zeichen sondern der ganze Ascii-Code) die Möglichkeit durch Drücken der `Insert`-Taste eine Box mit allen Sonderzeichen aufzurufen, aus der man mit der Maus ein Zeichen auswählen kann (braucht man zwar in CHAOS*ultd* nicht, ist aber trotzdem nett).
Weiterhin werden Popup-Menüs unterstützt, von denen CHAOS*ultd* auch ausgiebig Gebrauch macht.

Dialoge in CHAOS*ultd* können auch in Fenster gelegt werden. CHAOS*ultd* verwendet auch in Fenstern ausschließlich modale Dialoge, d.h. solange ein Dialog-Fenster geöffnet ist, kann nicht weitergearbeitet werden, sondern es muß der Dialog bearbeitet und geschlossen werden bevor es weiter geht.
Der Vorteil von Dialogfenstern ist begrenzt, wenn man nicht ein Multitasking-Betriebssystem verwendet (s.u.). Ohne Multitasking-Betriebssystem erlauben Fenster-Dialoge lediglich den Zugriff auf Accessories, mit erlauben sie anderen gleichzeitig laufenden Programmen weiterzulaufen und Ausgaben auf den Bildschirm (in eigene Fenster natürlich) zu machen.
Nachteile haben Dialogfenster auch, die gerade auf einem altersschwachen System wie ich es nutze (8Mhz 68k) zu bedenken sind. Es ist prinzipbedingt nicht möglich, den Hintergrund von Fenstern zu speichern, so daß nach dem Schließen des Fenster-Dialoges der Hintergrund jedesmal neu gezeichnet werden muß und das braucht Zeit. Wieviel hängt natürlich von dem Fenster ab, im allgemeinen nicht ewig, aber doch deutlich länger als das Restaurieren des Hintergrundes von normalen Dialogen, wo der Hintergrund (genügend Speicher vorausgesetzt, s.o.) zwischengespeichert wird, und so von der Dialogroutine selbst restauriert werden kann, was erheblich schneller geht.

1.5 CHAOS*ultd* und Multitasking

CHAOS*ultd* kann unter *präemtpiven* Multitasking-Systemen (fast) ohne Einschränkung betrieben werden. *Präemptives* Multitasking ist im Gegensatz zu kooperativem Multitasking zu sehen, das von CHAOS*ultd* nicht unterstützt wird. Auf dem ST (TT, Falcon) stehen mit Mag!X 2.0 und MultiTos *präemtptive* Multitasking-System zur Verfügung. Mag!X 1.x und MultiGEM bieten nur kooperatives Multitasking. (Ich habe nur Mag!X 2.0. Mit MultiTos habe ich nichts getestet.)
Vielleicht die eine Einschränkung zuerst: bei der (verdeckten) Berechnung von Bildern (und manchen sonstigen Aktivitäten von CHAOS*ultd*, die durch einen Info-Dialog angezeigt werden) kümmert sich CHAOS*ultd* nur relativ selten um GEM-Messages wie Fenster neu zeichnen, Fenster verschieben, Fenster nach oben holen etc.. Genaugenommen immer nur dann, wenn auch die Tastaturabfrage erfolgt (etwa zum Beenden der Berechnung).[15] Dies ist nicht so tragisch, hat aber zur Folge, daß das Programm nur etwas zögerlich (womöglich auch sehr verzögert) auf Fensteraktivitäten reagiert. (Wenn man versucht ein Fenster zu verschieben, dann springt das Fenster womöglich erst ettliche Sekunden nach dem Verschieben an die neue Position.) Hier ist einfach Geduld gefordert. (Die Alternative, ständiges Nachfragen nach GEM-Messages, wäre recht aufwendig zu implementieren und würde die Berechnung verlangsamen.)
Eine zweite kleine Unschönheit: während der Darstellung von Dialogen (in Fenstern) wird das Menü gesperrt, indem alle Menü-Titel (außer dem Accessory-Menü) disabled werden, was aber nicht angezeigt wird. Durch Prozeßwechsel unter Mag!X (MultiTos??) kann es sein, daß diese Menüleiste (grau) dargestellt wird. Nach Beenden des Dialoges wird die Menüleiste wieder enabled und dargestellt. (Gilt auch für Info-Dialoge.)
Wer Multitasking verwendet, sollte auch Fenster-Dialoge wählen. Für die Anzeige von (nicht-Fenster-)Dialogen, (nicht-Fenster-)Info-Dialogen, die Ganzseitenanzeige von Bildern und für die Anzeige der Berechnung wird nämlich

[15] bei kooperativem Multitasking ist diese zugleich die einzige Möglichkeit zum Prozeßwechsel. Da diese eben ein bißchen selten eintritt ist kooperatives Multitasking für CHAOS*ultd* nicht sinnvoll.

der gesammte Bildschirm durch CHAOS*ultd* reserviert. Andere Programme können solange keine Bildschirmausgaben machen (wieweit sie dann weiterlaufen können hängt von den Programmen ab). Deshalb die Fenster-Dialoge. Bei der Berechnung sollte man sich für die verdeckte Berechnung entscheiden.

1.6 Fenster

CHAOS*ultd* verwendet als GEM-konformes Programm natürlich Fenster. Und zwar

- *ein* Objekt-Fenster, in dem die vorhandenen (Bild-)Objekte angezeigt werden

  in der Info-Zeile des Objekt-Fensters wird die Zahl der vorhandenen Objekte, das erste angezeigte Objekt (falls gescrollt wurde), die Zahl der selektierten Objekte, der belegte und der (intern) freie Speicher angezeigt (die letzte Angabe ist natürlich mit der neuen Speicherverwaltung nicht sehr aussagekräftig)

- Anzeigefenster für ein oder mehrere Bild-Objekte (im folgenden Bild-Fenster genannt)
  im Titel des Fensters wird der Name des gerade angezeigten Bildes genannt

- Anzeigefenster für Bildvergleiche/Blöcke/Konstante (im folgenden Anzeige-Fenster genannt)
  Anzeige-Fenster unterscheiden sich von Bild-Fenstern dadurch, daß sie *nicht* direkt ein Bild-Objekt darstellen. Bei Blöcken/Konstanten liegt zwar ein Bild zugrunde, in diesem sind aber zusätzliche Linien eingezeichnet. Anders als Bild-Fenster gibt es in Anzeige-Fenstern immer nur *ein* Bild.

  In der Infozeile von Bild- und Anzeige-Fenstern wird ein Info mit dem Typ und der Größe des Originalbildes sowie der (ja möglicherweise konvertierten) tatsächlichen Anzeige ausgegeben. Wahlweise werden bei Bildern, denen ein Objekt mit geeigneten Koordinaten zugrundeliegt, die Position des Mauszeigers und die zugehörigen Punktkoordinaten in der Infozeile ausgegeben, solange sich der Mauszeiger über dem Fenster befindet (vgl. 4.5).

- Dialogfenster, die schon erklärt wurden

1.7 Objekte

CHAOS*ultd* kennt drei Typen von Objekten:

1. Fraktale

   Unter Fraktalen verstehe ich alle Bilder, die durch irgendwelche Berechnugsroutinen in CHAOS*ultd* erzeugt werden. Insofern ist die Bezeichnung Fraktale natürlich nicht ganz richtig, da solche Bilder nicht notwendig fraktal sein müssen.

2. Bilder

   gemeint sind parameterlose Bilder, die entweder geladen oder aus der Anzeige von Bildern, Vergleichen etc. erzeugt wurden (vgl. 4.7).
   Bilder ohne Parameter gibt es hauptsächlich der Vollständigkeit halber, sie spielen im folgenden kaum eine Rolle

3. Filme

   Filme in CHAOS*ultd* sind einfach geordnete Bildlisten. Sie können etwa zum Verknüpfen mehrerer Bildfolgen oder zum Festlegen der Reihenfolge, in der bestimmte Bilder gezeigt werden sollen, dienen.

Bilder (Fraktale oder einfache Bilder) können jede Größe und Farbtiefe mit den eingangs angegebenen Beschränkungen besitzen.

Jedes im Speicher vorhandene Objekt wird durch ein Icon im Objekt-Fenster angezeigt.
Dabei werden die verschiedenen Typen von Fraktalen durch verschiedene Icon dargestellt. Ebenso gibt es ein eigenes Icon für reine Bild-Objekte und für Filme.
Neben dem Objekt-Typ kann man am Icon ablesen, ob das zugehörige Objekt gespeichert ist oder nicht. Ist das Objekt gespeichert, so wird das Objekt abgehakt dargestellt (was man allerdings nur im unselektierten Zustand sehen kann).
Bei Fraktalen kann man darüberhinaus sehen, ob das Bild schon vollständig berechnet wurde oder ob es noch nicht fertig ist. Unfertige Bilder werden ohne Rahmen, fertige mit Rahmen angezeigt.
Nebeneinander werden im Objekt-Fenster stets nur so viele Objekte angezeigt, wie bei der gerade gewählten Fenster-Breite in das Fenster hinein passen. Sind mehr Objekte vorhanden, als im Fenster dargestellt werden können, so

kann man mit dem vertikalen Scroll-Balken des Fensters einstellen, ab welchem Objekt die Objekte angezeigt werden sollen.
Ebenfalls zum Scrollen dienen die Tasten `Cursor Up` und `Down`. Drückt man die Cursor-Tasten alleine, so wird zeilenweise, bei gleichzeitig gedrückter `Shift`-Taste, wird seitenweise gescrollt.

1.8 Selektieren von Objekten im Objekt-Fenster

Allgemein ist zum Selektieren von Objekten zu bemerken, daß selektierte Objekte selektiert bleiben, auch wenn sie aus dem Fenster herausgescrollt werden und deshalb nicht sichtbar sind.

Maus

Mit der Maus können

- einzelne Objekte angeklickt werden – sie änderen dann ihren Status. D.h. selektierte Objekte werden deselektiert, nicht selektierte ausgewählt.
- *Gummibänder* aufgezogen werden, wie man das vom Desktop her kennt. Gummibänder können (anders als auf dem Desktop) auch von rechts unten nach links oben aufgezogen werden. Während des Aufziehens wird der Mauszeiger nicht als Pfeil sondern als zeigende Hand dargestellt. Angesprochen werden damit alle Objekte, die ganz oder teilweise innerhalb des aufgezogenen Rechteckes liegen.

  Der Zustand dieser Objekte wird je nach dem Status der Sondertasten verändert: ist keine Sondertaste gedrückt, so werden die Objekte selektiert, ist eine `Shift`-Taste gedrückt deselektiert und bei gedrückter `Control`-Taste wird der Objekt-Status geändert.
- Analog zur Vorgängerversion von CHAOS*ultd* kann man auch Objektbereiche ansprechen, indem man zunächst einen `Shift`-Klick auf das erste Objekte ausführt. Klickt man nun ein weiteres Objekt an, so werden alle Objekte zwischen dem zuerst und dem dann angeklickten Objekt angesprochen. Diese werden wieder je nach Sondertasten-Status beim *zweiten* Klick selektiert, deselektiert oder geändert (letzteres ist *nicht* mehr analog zur Version 5.0).

  Zwischem dem ersten (`Shift`-)Klick und dem Auswählen des zweiten Objektes wird der Mauszeiger innerhalb des Objektfensters als zeigende Hand dargestellt (wie beim Aufziehen eines Gummibandes). Damit soll gewissermaßen der andere Zustand, in dem sich der Mauszeiger jetzt befindet, deutlich gemacht werden. Ein Mausklick in diesem Zustand hat ja eine andere Wirkung als sonst.

  Möchte nach der Auswahl des ersten Objektes doch auf die Selektion eines Objektbereiches verzichten, so kann man neben eines der Objekte klicken, der Sonderzustand der Maus wird dann aufgehoben.
- um die Bilder eines Filmes zu selektieren (deselektieren) kann man einen `Control`-Klick (`Shift-Control`-Klick) auf den Film ausführen

Popup-Menü

Viele der Funktionen von CHAOS*ultd* können auch über Popup-Menüs aufgerufen werden, auch zum Selektieren steht ein solches bereit. Aufgerufen wird es durch Doppelklick auf ein Objekt oder durch einfachen Mausklick neben die Objekte (jeweils mit der linken Maustaste).
Das Popup-Menü stellt vier Bereiche zur Verfügung, zum Selektieren, Deselektieren, Ändern und exklusivem Selektieren. In jedem Bereich kann man wählen, welche Objekte angesprochen werden sollen: Objekt (nur, wenn man auf ein Objekt geklickt hat), Zeile (nur, wenn man in eine Zeile geklickt hat), usw..

Tastatur

Neben den Mauskommandos und dem Popupmenü stehen noch eine Reihe von Tastaturkommandos zum Selektieren von Bilder zur Verfügung:
Mit den Tasten `F1` bis `F10` selektiert man eine Zeile (1 bis 10) im Objektfenster; wählt man dabei eine Zeile, die kein Objekt enthält, so wird die letzte nichtleere Zeile eingestellt – hat man mehr als zehn Zeilen im Objektfenster, dann kann man nur die ersten zehn Zeilen mittels Tastatur ansprechen (kann mir mangels Bildschirmauflösung nicht passieren; gemeint sind immer die angezeigten Zeilen; nicht sichtbare Objekte können via Tastatur nicht selektiert werden).
Mit den Tasten `1` - `9`, `0` und `ß`, `'` sowie `#` kann man dann die einzelnen Objekte dieser Zeile ansprechen (wiederum maximal das 13. Objekte, aber wer hat schon einen so breiten Bildschirm), mit `Tab` die ganze Zeile.
`Clr` wirkt auf alle Objekte (unabhängig der eingestellten Zeile).

Die Taste `Ins` bezieht sich auf alle geänderten, aber nicht gespeicherten Objekte, `Del` auf alle fertigen Bilder, auf alle nicht fertigen. Die Tasten `.` und `-` wirken jedes zweite bzw. vierte Objekt.
Dabei wird das Objekt bzw. werden die Objekte bei normalem Tastendruck selektiert, mit `Shift`-Taste deselektiert, mit `Control` geändert und mit `Alternate` als einziges Objekt selektiert.
Die Tastaturbefehle für das Desktop sind auch in einer Dialogbox zusammengefasst, die man durch Drücken der `Help`-Taste aufrufen kann.

1.9 Bedienungselemente zum Aufruf von Funktionen

Bevor die Funktionen von CHAOS*ultd* im einzelnen erläutert werden, stelle ich zunächst die Möglichkeiten diese Funktionen aufzurufen kurz dar.

Menü

Die GEM-Menüleiste ist ja wohl allgemein bekannt, die Menüpunkte können auch durch die im Menü angezeigten Tasten aufgerufen werden.
Dennoch gibt es ein paar Besonderheiten zu beachten:
Zum einen sei hier ausdrücklich darauf hingewiesen, daß CHAOS*ultd* einige Menüpunkte mehrfach belegt, wobei je nach Sondertastenstatus zwischen den Funktionen unterschieden wird. Dies gilt insbesondere für die Einträge der Fraktal-Typen im neu-Menü und die Funktion neu berechnen.
Zum anderen bei Funktionen, die sich auf Objekte von CHAOS*ultd* – also Bilder, Fraktale etc. – beziehen. Bei der Frage, auf welche Objekte sich die Auswahl einer solchen Funktion jeweils bezieht, muß man zunächst unterscheiden, welches Fenster bei Auswahl des Menüpunktes gerade aktiv ist. Unterschieden wird zwischen Objekt-, Bild- und Anzeige-Fenstern.
Ist kein Fenster von CHAOS*ultd* aktiv (etwa weil ein Fenster eines Accessories aktiv ist) so bezieht sich die Objekt-Auswahl auf das letzte aktive Fenster von CHAOS*ultd*. Ist kein Fenster von CHAOS*ultd* offen, so gilt dasselbe wie bei aktivem Objekt-Fenster.

- Objekt-Fenster aktiv
  - ist kein Objekt selektiert, so bezieht sich die Funktion auf *alle* Objekte (bzw. alle Fraktale, falls die Funktion für einfache Bilder keinen Sinn macht, oder auch alle nicht fertigen Bilder oder was auch immer)
  - sind Objekte selektiert und zwar irgendwelche, so bezieht sich die Funktion auf die selektierten Objekte (soweit sie weitere notwendige Kriterien erfüllen, d.h. unter Umständen auf gar keine, nämlich dann wenn nur Objekte selektiert sind, die die weiteren Kriterien nicht erfüllen; in diesem Fall wird die Funktion ohne weitere Fehlermeldung abgebrochen),

    es seie denn,
  - es wurde bei der Menüauswahl gleichzeitig mit dem Mausklick (links) die rechte Maustaste oder eine der `Shift`-Tasten gedrückt. In diesem Fall bezieht sich die Funktion wiederum auf alle Objekte.
- Bild-Fenster aktiv
  - die „normale“ Menüauswahl bezieht sich auf das gerade angezeigte Bild
  - die Menüauswahl mit gedrückter `Shift`- oder rechter Maustaste bezieht sich auf alle Bilder der Bildliste
- Anzeige-Fenster aktiv

  Die Menüauswahl bezieht sich auf das angezeigte Bild.

Die Auswahlmöglichkeiten, also die Frage, welche Menüpunkte man auswählen kann, und welche gesperrt sind, hängen natürlich von diesen Regeln ab, wegen der Möglichkeit des `Shift`-Klickens kann es sein, daß Menüpunkte für normales Auswählen inaktiv sind aber nicht gesperrt werden.
Bei Funktionen die sich nur auf einzelne Bilder beziehen existiert die Möglichkeit des `Shift`-Klickens nicht, die `Shift`-Taste (rechte Maustaste) wird dann einfach ignoriert.

Popup-Menüs

Mit Hilfe der Popup-Menüs kann man ettliche Funktionen der Menüleiste ebenfalls aufrufen.
Im Objekt-Fenster gibt es mehrere Popup-Menüs: das schon besprochene Menü zum Selektieren von Objekten sowie drei Menüs zum Auswählen von Funktionen.

Die letzten drei erreicht man durch Mausklick *rechts*, wobei unterschieden wird, ob man auf ein Objekt klickt oder nicht. Bei Objekten wird weiter zwischen Filmen und Bild-Objekten unterschieden (deswegen drei Popup-Menüs, die unterschiedliche Funktionen enthalten).
Klickt man auf ein Objekt, so beziehen sich alle folgenden Auswahlen auf genau dieses eine Objekt, ansonsten gelten die gleichen Auswahlregeln wie bei der Auswahl mit dem Drop-Down-Menü bei aktivem Objekt-Fenster. (statt gleichzeitig gedrückter rechter Maustaste, wird das Popup-Menü allerdings mit rechter (oder linker) Maustaste verlassen).
Die Auswahl mit dem Popup-Menü kann auch erfolgen, wenn das Objekt-Fenster zwar offen aber nicht aktiv ist (mit einem Mausklick rechts zum Aufruf des Menüs wird es auch nicht aktiviert).
Weitere Popup-Menüs stehen in Bild-Fenstern und Anzeige-Fenstern zur Verfügung. Beim Popup in Bild-Fenstern wird (wie bei der Menüauswahl) je nach Zustand der `Shift`-Taste unterschieden, ob sich die Funktion auf alle Bilder (gedrückte `Shift`-Taste; natürlich nur falls überhaupt sinnvoll) oder nur auf das aktuelle Bild beziehen soll.

1.10 Online-Hilfe

CHAOS*ultd* verfügt (im Prinzip) über ein umfangreiches kontextsensitives Hilfs-System. Im Prinzip, weil die Hilfstexte noch nicht geschrieben sind, und das Hilfs-System somit wenig hilfreich ist.
Dennoch soll das Hilfs-System hier schon einmal beschrieben werden, auch wenn ich nicht weiß, ob und wann ich die Hilfstexte je schreibe (zumal die Resonanz auf CHAOS*ultd* so umwerfend ist[16], daß das Programm vermutlich eh keiner außer mir benutzt, und wofür wird dann ein Hilfs-System gebraucht).
Das Hilfs-System stellt wie gesagt zahlreiche Hilfstexte zur Verfügung und zwar

- zum Desktop und
- zu allen Menüs

  diese Hilfstexte sind über das Keine Panik!-Menü (ja, das ist jetzt ein Menü) rechts oben zu erreichen. Den Hilfstext zum Desktop erreicht man auch durch Drücken der `Help`-Taste (dieser Hilfstext ist einer der wenigen Texte die schon geschrieben sind).
- zu jeder Dialogbox

  in (fast) allen Dialogboxen existiert ein Hilfe-Button, den man anklicken kann, den man aber auch durch Drücken der `Help`-Taste aktiviert. Im allgemeinen findet sich dieser Button links unten, bei Platzmangel habe ich das Hilfe bisweilen zu H verkürzt.

Nach Aufruf des Hilfs-Systems erscheint der Hilfs-Dialog, in dem der Hilfstext angezeigt wird. Dieser kann aus bis zu 16 Seiten bestehen, zwischen denen man mit den Button weiter und zurück blättern kann. Neben diesen Button wird übrigens angezeigt, welche Seite von wievielen gerade dargestellt wird.
Über den Button befindet sich eine Zeile, in der bis zu drei Querverweise zum Hilfstext angezeigt werden, die man durch Anklicken aufrufen kann. Dann wird der zu diesen Querverweisen gehörende Hilfstext geladen. Zurück zum vorherigen Hilfstext gelangt man mit dem Button voriges. Die maximale Schachtelungstiefe von Querverweisen ist 16.
Ebenfalls wie ein Querverweis gehandhabt wird die Hilfe zur Hilfe, die man mit dem Hilfe-Button im Hilfs-Dialog aufrufen kann (dieser Text existiert ebenfalls bereits).
Unabhängig von der aktuellen Schachtelungstiefe kann der Hilfs-Dialog mit dem Button Abbruch beendet werden.

2 Bilder erzeugen

2.1 neue Bilder – Eingabe der Parameter

Um neue Bilder zu erzeugen wählt man im Menü Neu den Menüpunkt, der zum gewünschten Bildtyp gehört. Das Anklicken des Menüpunktes mit gedrückter `Shift`- oder `Control`-Taste hat eine Sonderbedeutung (vgl. 6.3), diese Sondertasten dürfen also nicht gedrückt sein.
Gibt es mehr als 16 Bildtypen, so wird ein zweites Neu-Menü angezeigt. Dessen Einträge lassen sich genauso wie die des ersten Neu-Menüs via Tastatur ansprechen, wobei die gleichen Tasten Verwendung finden – also die des Ziffernblockes – wobei aber zusätzlich die `Alternate`-Taste gedrückt werden muß.

[16] CHAOS*ultd* ist **POSTCARDWARE!**

Einstellen des Bildformates

Zunächst wird ein Dialog dargestellt, in dem man die Zahl der zu erzeugenden Bilder[17], den Namen der Bilder (oder des Bildes, man kann natürlich auch nur ein Bild erzeugen; bei der Erzeugung mehrerer Bilder werden die Bildnamen von CHAOS*ultd* durchnummeriert, die letzten Zeichen im Namen werden dabei gegebenenfalls überschrieben) angibt, sowie die Größe und Farbtiefe des Bildes festlegt. Letzteres geschieht über Popup-Menüs für die Breite, die Höhe und die Zahl der Farben (bzw. Raster)[18]. Bei Breite und Höhe enthält das Popupmenü natürlich nur eine Auswahl möglicher Größen, dazu gibt es die Möglichkeit, den Eintrag frei im Popup-Menü einzustellen und die Breite (bzw. Höhe) als Zahl einzugeben. Falls nötig wird diese Angabe modulo 16 bei der Breite und modulo 4 bei der Höhe (ab)gerundet (d.h. so, daß das Ergebnis durch 16 bzw. 4 teilbar ist). Hinter dem Popup-Menü für die Zahl der Farben wird noch die Zahl der dafür nötigen Bitmaps (Bildebenen) angegeben.
Es ist möglich, daß bestimmte Routinen nicht alle Bildgrößen und vor allem Farbanzahlen unterstützen. Insbesondere gibt es jetzt schon einige Routinen, die nur zweifarbige Darstellung zulassen. Die Auswahlmöglichkeit ist dann eingeschränkt.
Neben der expliziten Eingabe von Breite, Höhe und Farbzahl kann man auch ein Format als Ganzes auswählen, ebenfalls via Popupmenü. Die Einstellungen für Breite, Höhe und eventuell Farbzahl werden entsprechend den Parametern des gewählten Formates gesetzt. In der Formatdefinition kann die Farbzahl fehlen, dann wird sie auch nicht gesetzt. Der erste Eintrag dieses Popup-Menüs steht für das gerade verwendete Bildschirmformat, es folgen eine Reihe von möglichen anderen Formaten (einfach ausprobieren), ein nicht anwählbarer Eintrag unbekannt, der angezeigt wird, falls man mit der Einstellung für Breite, Höhe und/oder Farbtiefe ein Format gewählt hat, für das kein Eintrag im Popupmenü existiert, und ein Eintrag übernehmen. Mit diesem letzten Eintrag kann man die Parameter von einem geladenen Objekt übernehmen (vgl. 2.2). (Sind nicht alle Bildgrößen und/oder Farbanzahlen möglich, so werden die Parameter nach Wahl eines Formates automatisch angepaßt.)
Unter dem Format-Popup wird angezeigt, wieviel Speicher (intern) noch frei ist, wieviel Speicher ein Bild des gewählten Formates benötigt und wieviel Speicher alle Bilder zu berechnenden Bilder brauchen würden; dabei wird der Speicherplatzbedarf ungepackter Bilder zugrundegelegt, d.h. die Bilder belegen, falls sie gepackt werden, letztenendes weniger Speicherplatz. Die Anzeige wird nach der Anwahl eines der Popup-Menüs aktualisiert, man kann dieses Aktualisieren auch durch Anklicken der Anzeige erreichen. Falls Breite oder Höhe auf frei eingestellt ist, wird der gerade eingegebene Wert für diese Anzeige nur ausgelesen, wenn man nochmals auf frei in diesem Popup-Menü klickt oder die Anzeige durch Anklicken aktualisiert.
Zum Verlassen der Dialogbox gibt es die üblichen Button Ok und Abbruch, links neben diesen gibt es noch einen Button Farben, der in den Farbselector verzweigt.

Farbselector

Der Farbselector dient dem Einstellen der Farben, und das ist leichter gesagt als getan. Das Problem des Farbselectors ist, daß er auch bei monochromer Bildschirmdarstellung die Einstellung von bis zu 256 Farben zulassen muß, und das ist nicht so einfach zu realisieren, wenn man gleichzeitig hohen Komfort bei der Eingabe der Farben erreichen will.
Die Bequemlichkeit des jetzt existierenden Farbselectors ist denn auch bei vielen Farben nicht so ganz überzeugend, wenngleich Interpolationsfunktionen auch die rasche Einstellung vieler Farben erlauben.

Der Farbselector zeigt prinzipiell nur die Einstellung *einer* Farbe an. Diese kann man links unten mit den Button << und >> auswählen, wobei Doppelklick zur ersten bzw. letzten Farbe schaltet. Stehen auf dem Bildschirm vier oder mehr Farben zur Verfügung so wird die gerade eingestellte Farbe in dem Balken unter der Farbnummer angezeigt (wobei alle Veränderungen sofort angezeigt werden). Die Farbeingabe wirkt ansonsten *nicht* direkt auf die eingestellte Farbtabelle.
Den größten Teil des Farbselectors nehmen die beiden Kästen mit den Einstellbalken für die Farbe ein. Links hat man die Möglichkeit, die RGB- (Rot-, Grün- und Blau-) Werte der Farbe einzustellen, und zwar auf einer Skala von 0 bis 1000 (entsprechend den GEM-Konventionen) – rechts kann man alternativ die Farbe im HLS-Modell (hue, lightness, saturation; d.h Farbwert, Helligkeit und Sättigung) einstellen, wobei die Einstellung im einen Modell jeweils automatisch in die Anzeige des anderen übernommen wird.
Mit der Auswahl der einzustellenden Farbnummer und der Einstellung der gewünschten Farbe im RGB- oder HLS-Modell kann man nun die Farbpalette zusammenstellen, was bei vielen Farben aber recht langwierig ist, weshalb es noch einige Hilfsfunktionen gibt. Dazu dienen die Button links von der Farbnummernanzeige.
Mit Interp. kann man die zwischen zwei Farbnummern liegenden Farben durch Interpolation deren Werte berechnen. Klickt man den Button zum ersten Mal an, so wird die aktuell eingestellte Farbnummer als Startnummer gespeichert, der Button wird zur Erinnerung abgehakt dargestellt. Klickt man den Button jetzt noch einmal bei einer anderen

[17] CHAOS*ultd* erlaubt es Bilderfolgen zu erzeugen, bei denen nur die Parameter des ersten und letzten dieser Bilder eingegeben werden. Unter Zahl der Bilder ist die Zahl der zwischen diesen Bildern durch Interpolation zu erzeugenden Bilder gemeint

[18] CHAOS*ultd* kann in monochromen Bitmaps durch Rasterung 5 (2x2 Raster) bzw. 17 (4x4 Raster) Graustufen erzeugen

Farbnummer an (es muß mindestens eine Farbnummer zwischen der Startnummer und dieser anderen Farbnummer liegen, sonst wäre das Ganze wenig sinnvoll), so erscheint ein Popup-Menü, in dem man auswählen kann, ob man im RGB- oder HLS-Modell interpolieren will (interpoliert wird stets linear zwischen den Anfangs- und End-Werten). Klickt man neben das Popup-Menü, so wird die Funktion abgebrochen, ansonsten werden die Farben interpoliert (kann bei vielen Farben vor allem im HLS-Modell etwas dauern).
Übernehmen steht für die Übernahme der aktuell auf dem Bildschirm eingestellten Farben. Damit kann man – wenn man Fensterdialoge verwendet, und die einzustellende Farbtabelle nicht mehr Farben als die Bilderschirmdarstellung enthält – die Farben mit einem externen Accessory einstellen, und dann in CHAOS*ultd* übernehmen.
Holen bedeutet, daß man die Farben aus einem bereits bestehenden Bild-Objekt in CHAOS*ultd* auslesen lassen kann. Wählt man den Menüpunkt an, so kann man auf dem Objektfenster durch Doppelklick ein Objekt auswählen, dessen Farben dann übernommen werden (siehe auch 2.2).
Standart setzt die Farben auf eine (eher mäßige) Standarteinstellung, die auch vor Aufruf des Farbselectors eingestellt wird (s.u.).

Noch ein Wort zur Voreinstellung der Farben. Der Aufruf des Farbselectors von der Dialogbox zum Einstellen des Bildformates ist nicht obligatorisch. Dennoch müssen die Farben natürlich irgendwie gesetzt werden. Ebenso sollte man beim Aufruf des Farbselectors irgendeine wenigstens einigermaßen passende Voreinstellung vorfinden (zumindest müssen die Werte zwischen 0 und 1000 liegen, sonst spielt der Farbselector nämlich verrückt).
Um dies zu erreichen, erzeugt CHAOS*ultd* für die Farbtabelle eine Voreinstellung (Standarteinstellung)[19]. Und zwar entweder unmittelbar vor dem Aufruf des Farbselectors oder bei Verlassen der Dialogbox zum Einstellen des Bildformates, wenn der Farbselector nicht aufgerufen wurde und noch keine Farbtabelle erzeugt wurde. Das Problem dabei ist die unbekannte Zahl von Farben, die sich mit der Einstellung des Bildformates ändern kann. Wird die Zahl der Farben nach Aufruf des Farbselectors erhöht, so geht CHAOS*ultd* davon aus, daß keine Farbtabelle existiert. Wird die Zahl der Farben verringert, so wird die Farbtabelle abgeschnitten, was aber nicht unbedingt zu einem brauchbaren Resultat führen muß – in diesem Fall ist der Anwender gefragt. Man sollte deshalb immer zuerst die Zahl der Farben einstellen (und dann konstant lassen; Vorsicht beim Wählen eines Formates) und sich dann um die Einstellung der Farbtabelle kümmern; dann sollten keine Probleme auftreten.

Fraktal-Parameter

Nach der Eingabe des Bildformates und eventuell der Farbtabelle müssen noch die Fraktal-Parameter, die von den Berechnungsroutinen abhängen und das zu erzeugende Bild im einzelnen festlegen, eingegeben werden. Die Details dieser Eingabe hängen von den Berechungsroutinen ab.

2.2 Eingabehilfen

Übernahme von Daten aus bestehenden Objekten

Bei der Einstellung des Bildformates und der Farbtabelle, aber auch bei der Eingabe von Fraktal-Parametern, ist es möglich, die Daten aus einem schon bestehenden Objekt zu übernehmen, bei Fraktal-Parametern kann man auch Blöcke oder einzelne Punktkoordinaten in Bildern mit der Maus markieren (soweit die Berechnungsroutinen dies unterstüzen).
Ruft man in der entsprechenden Dialogbox diese Funktion(en) auf, so wird zunächst die Dialogbox geschlossen und der Desktop angezeigt, man befindet sich in einer Art Auswahl-Zustand von CHAOS*ultd*. Viele der Funktionen von CHAOS*ultd* lassen sich auch in diesem Zustand aufrufen, *nicht* allerdings das Verlassen des Programmes und alle Funktionen zum Erzeugen von Bildern. Es ist prinzipiell unmöglich, diesen Zustand aus sich selbst heraus nochmal aufzurufen (wenn man etwa die Bildparameter und dann den Farbselektor aufruft, kann man die Funktion holen *nicht* aufrufen).
Wie kann man nun aber Objekte oder Blöcke bzw. Koordinaten auswählen? Was man auswählen kann, hängt von der aufrufenden Funktion ab, ebenso wie die Frage, welche Daten dann übernommen werden.
Objekte werden ausgewählt, indem man auf ein Objekt im Objekt-Fenster doppelklickt. Dabei ist es möglich, daß sich die Objekt-Auswahl auf Objekte eines speziellen Objekttypes beschränkt, die Auswahl eines Objektes mit einem anderen Objekttyp bewirkt dann den Abbruch der Auswahl.
(Bisher) nur beim Edieren von Filmen (vgl. 3.1) ist auch die Angabe eines Bereiches von Objekten möglich, analog zum Selektieren eines Objektbereiches. So wie dort, klickt man zunächst ein Objekt mit gedrückter `Shift`-Taste an – dieses Objekt ist dann der Anfang des Bereiches; ebenso wie beim Selektieren wird der Mauszeiger als zeigende Hand dargestellt (CHAOS*ultd* weiß ja auch noch gar nicht, ob man einen Bereich selektieren oder auswählen will). Doppelklickt man jetzt auf ein Objekt, so wird dieses als Ende des Bereiches erfaßt und der Bereich ist ausgewählt.

[19] die Voreinstellung ist derzeit bei zwei Farben scharz und weiß (genaugenommen umgekehrt); bei 4 bis 16 Farben werden die Farben einer 16 farbigen Farbtabelle entnommen und bei mehr als 16 Farben zwischen den Farben dieser 16 farbigen Tabelle interpoliert

Dabei wird die Reihenfolge der beiden Objekte beachtet, d.h. der Bereich beginnt bei dem zuerst angeklickten Objekt und endet beim zweiten, auch wenn das zweite in der Objektliste weiter vorne steht.

Blöcke werden ausgewählt, indem man in ein Bild-Fenster klickt. Damit startet man die Blockdefinition. Der Block kann jetzt mit der Maus und/oder Tastatur festgelegt werden. Zur Mauseingabe gibt es drei (zwei) Modi, zwischen denen mit der linken Maustaste weitergeschaltet wird. Im ersten Modus legt man die Größe des Blockes fest, wobei man die Blockhöhe nur festlegen kann, wenn man beim Starten des Blockfestlegens die `Shift`-Taste gedrückt hatte. Andernfalls wird die Blockhöhe so festgelegt, daß das Verhältnis von Blockhöhe zu Breite dem der Bildgröße entspricht.
Im zweiten Modus kann man den Block verschieben, im dritten den Block rotieren, wenn dies vom Fraktaltyp unterstützt wird (bisher nur Fractale).
Der jeweilige Mausmodus wird – zusammen mit den Blockkoordinaten und dem Winkel – in der Infozeile des Fensters angezeigt, wobei `gr` für Größe ändern, `pos` für Position ändern und `rot` für Winkel ändern steht. Ist der Winkel ungleich 0, so sind die Blockkoordinaten so zu verstehen, daß sie ein Rechteck angeben, das noch um seinen Mittelpunkt mit dem angegebenen Winkel gedreht wird. Die linke obere Ecke hat nach dieser Drehung natürlich andere Koordinaten als das angegebene minimale x und y.
Beendet wird die Blockdefinition mit der rechten Maustaste, wobei ein Einfachklick 'Ok' und ein Doppelklick 'Abbruch' bedeutet.
Mit der Tastatur kann der Block unabhängig vom Mausmodus verschoben, skaliert und gedreht werden.
Mit den Ziffernblock-Tasten `4`, `6`, `8` und `2` sowie mit den Cursortasten kann der Block nach links, rechts, oben oder unten verschoben werden. Mit `(` und `)` sowie `+` und `-` kann man die Blockbreite bzw. -höhe verändern, letzteres natürlich nur, wenn das Seitenverhältnis beliebig ist. Mit den Tasten `0` und `.` kann man den Block rotieren (falls möglich). (Es sind jeweil die Tasten auf dem Ziffernblock gemeint).
Die jeweilige Änderung ist klein, wenn man die Taste allein drückt, mittel bei gedrückter `Shift`-Taste und (relativ) groß bei gedrückter `Alternate`-Taste.
`Undo` bricht ab, `Return` und `Enter` übernehmen den Block.
Wählt man einen Block in einem bereits gedrehten Bild aus, so ist zu beachten, daß der Block zunächst immer erst ungedreht (Winkel 0 Grad) aufgezogen wird, d.h. der Block erscheint relativ zum Bild gedreht (weil er ja zurückgedreht werden muß). Beim Verschieben wird die Mausbewegung bzw. die Tastatureingabe so transformiert, daß eine Bewegung nach rechts auch den Block auf dem Bildschirm nach rechts bewegt. Bei gestauchten Bildern, bei denen das Verhältnis von Bildbreite zu -höhe stark von entsprechenden Verhältnis der maximalen/minimalen Koordinaten abweicht wird die Verschiebung in x bzw. y-Richtung entsprechend skaliert.
Blockauswahl ist nicht in allen Bildern möglich, da nicht immer eine Zuordung der Bildpunkte zu Koordinaten möglich ist (beispielsweise nicht bei 3d Fractalen; ebensowenig bei geladenen parameterlosen Bildern). Blöcke können allerdings nicht nur in Bild-Fenstern, sondern auch in Anzeige-Fenstern bei Block- oder Konstanten-Anzeigen ausgeschnitten werden, sie beziehen sich dann auf das zugrundeliegende Bild (vgl. 4.3 und 4.4). Prinzipiell nicht möglich ist das Ausschneiden von Blöcken aus Bildvergleichen.

Koordinaten eines Bildpunktes wählt man durch Doppelklick in ein Bild aus, wobei die gleichen Einschränkungen wie bei Blöcken gelten.
Abbrechen kann man die Auswahl von Daten durch einen Doppelklick ins Objektfenster, neben ein Objekt.
Während dieses Auswahlzustandes kann man mit der `Help`-Taste einen Hilfsdialog anzeigen, der diese Optionen kurz wiederholt, wobei nicht mögliche Optionen grau dargestellt werden. Außerdem wird angezeigt, wofür man die Daten auswählt, und ob ein beliebiges Objekt oder nur ein Objekt eines speziellen Objekttypes (der dann genannt wird) ausgewählt werden soll.

Laden aus Parameterdateien

Bei einigen Fraktal-Typen (IFS, L-Systeme) können Teile der Parameter aus Dateien geladen werden, wobei die Parameterdateien jeweils mehrere Parametersätze enthalten.
Zu diesem Zweck gibt es eine Parameter-Auswahlbox, deren Bedienung im folgenden erläutert werden soll.
Nach dem ersten Aufruf einer solchen Lade-Funktion erscheint zunächst der Dateiselektor, in dem man eine Parameterdatei angeben muß. Bei weiteren Aufrufen *kann* die Datei vom letzten Mal noch im Speicher vorhanden sein, so daß dieser Schritt unter Umständen entfällt.
Die Datei wird eingelesen und eine Tabelle mit den Parametersätzen erzeugt. In der Auswahlbox werden nun in einem ziemlich großen Feld (4x16 oder so) die Parametersätze angezeigt. Sind mehr Parametersätze vorhanden, als auf einmal angezeigt werden können, so werden mehrere Seiten verwaltet, zwischen denen man mit zwei Button hin und herschalten kann.
Ansonsten hat man die Möglichkeit, einen Parametersatz auszuwählen, indem man ihn anklickt (führt unmittelbar zum Verlassen der Auswahlbox), eine neue Parameterdatei zu laden (`Laden`) oder die Auswahlbox ohne Wahl zu

verlassen (Abbruch). Der Name der geladenen Parameterdatei wird zwischen den Scrollbutton und dem Lade-Button angezeigt.

2.3 Berechnung von Bildern

Berechnungsmodi

Es gibt zwei grundsätzlich verschiedene Berechnungsmodi in CHAOS*ultd* GEM. Im einen wird auf dem (gesamten) Bildschirm das Ergebnis der laufenden Berechnung angezeigt (soweit es draufpaßt). Im anderen wird die Berechnung ohne gleichzeitige Anzeige auf dem Bildschirm durchgeführt. Dann weist lediglich ein Info-Dialog rechts oben auf dem Bildschirm (gegebenenfalls im Fenster) auf die Berechnung hin. Dadurch spart man sich die Punktzeichnerei auf dem Bildschirm, die wegen der GEM-konformen Ausgabe halt auch ihre Zeit braucht. Wichtig für Multitasking-User: wird der Info-Dialog nicht in ein Fenster gelegt, so wird durch ihn der ganze Bildschirm für GEM-Ausgaben gesperrt! Beim Mitzeichnen auf den Bildschirm wird das Bild in die Bildmitte zentriert, solange es kleiner als der Bildschirm ist; ist es größer, so wird die linke obere Ecke angezeigt.
Bei diesem Mitzeichnen kann (bisher nur bei monochromen Bildern oder Bildschirmen) das Ausgeben der Punkte gepuffert werden, was man aber auch abstellen kann (dann geht es noch langsamer). An dieser Stelle findet sich ein Fehler, nicht in CHAOS*ultd* aber in NVDI (V2.01). Macht man das Ganze in mittlerer Auflösung (aber mit zweifarbigem Fraktal), dann wird von den gepufferten Punkten immer nur einer gezeichnet (betrifft nur die Bildschirmdarstellung; intern wird das Bild korrekt erzeugt). Dieser Fehler liegt nicht bei CHAOS*ultd* (er verschwindet, wenn man NVDI rausschmeißt), am einfachsten umgeht man ihn, indem man die Pufferung der Ausgabe abschaltet.

Berechnungsunterbrechung

Die aktuelle Berechnung eines Bildes kann (soweit von den Berechnungsroutinen unterstützt) auf zweierlei Weise unterbrochen werden.
Durch Drücken der `Esc`-Taste kann man die Berechnung des aktuellen Bildes abbrechen. Hält man gleichzeitig die `Shift`-Taste gedrückt, so werden eventuell noch folgende Bilder nur noch erzeugt, ihre Berechnung aber nicht mehr begonnen. Die `Alternate`-Taste bewirkt, daß werden weitere Bilder auch nicht mehr erzeugt werden (wobei vorher eventuell noch nachgefragt wird, ob man dies wirklich will). *Achtung!* Der Sondertastenstatus wird nicht beim Tastendruck selbst registriert, sondern die Sondertaste muß bis zum tatsächlichen Abbruch gedrückt werden.
Durch Drücken der `Space`-Taste kann man einen Info-Dialog aufrufen, in dem angezeigt wird, welches Bild gerade berechnet wird, wie weit dieses Bild schon berechnet wurde etc.. Durch Verlassen des Info-Dialoges mit Abbruch kann man die Berechnung abbrechen, wobei die Sondertasten wie bei `Esc` wirken.
Im Zusammenhang mit dem Berechnungsinfo ist noch eine kleine Unzulänglichkeit von CHAOS*ultd* zu konstatieren: wird das Berechnungsinfo beim Berechnen mit Anzeige aufgerufen und ist CHAOS*ultd* (genauer MyDial) nicht in der Lage, den Hintergrund zu puffern, so wird dieser nicht restauriert, sondern der Bildschirm schlicht gelöscht (Auf das zu berechnende Bild hat das natürlich keinen Einfluß), anschließend werden nur die neu berechneten Punkte ausgegeben.

Weiterberechnen von Bildern

Durch Anklicken von weiter im Extra-Menü kann man unfertige Bilder weiterberechnen lassen.
Die Berechnung kann genauso wie beim Berechnen neuer Bilder abgebrochen werden.
Ist ein Algorithmus nicht in der Lage ein Bild nach Berechnungsabbruch weiterzuberechnen, so kann er bei Berechnungsabbruch das Bild für fertig erklären; einige Routinen erlauben dies auch durch den Benutzer zu erzwingen (z.B. Hüpfer). weiter läßt sich in diesem Fall natürlich nicht mehr verwenden.
Wird das Bild mit Anzeige berechnet, so wird das bisher berechnete Bild ebenfalls angezeigt. Reicht der Speicher für diese Anzeige nicht, so unterbleibt diese Anzeige und man sieht nur die neu erzeugten Punkte. Dies geschieht *ohne* Fehlermeldung.
Bei übergroßen, zeilenweise berechneten Bildern wird beim Aufruf von weiter der beim Berechnen angezeigte Bildausschnitt angepaßt. Dieser wird solange nach unten verschoben, bis das restliche Bild (in der Höhe) ganz auf den Bildschirm paßt. Bis dahin erfolgt die Anzeige der Berechnung so, daß die erste neu berechnete Zeile jeweils die oberste ist.

2.4 Neu berechnen

Dieser Menüpunkt dient dazu, Bilder mit teilweise geänderten Parametern neu zu berechnen – etwa wenn man Bilder testweise in einem kleinen Bildschirmformat berechnet hat, und jetzt große Bilder haben will, aber auch wenn man einen der Parameter aller Bilder (z.B. die Berechnungstiefe von Apfelmännchen) ändern will.

Zunächst kann man eventuelle Änderungen des Bildschirmformates eingeben. Dazu dient die gleiche Dialogbox, in der man auch das Bildschirmformat neuer Bilder einstellt, allerdings mit einigen Änderungen. Zum einen kann man bei den Parametern des Bildschirmformates angeben, welche der Parameter verändert werden sollen, indem man die Button links von den Parametern selektiert. Zum anderen kann man die Box statt mit Ok mit zwei anderen Button verlassen, nämlich Neu und Überschr., wobei der erste Button bedeutet, daß veränderte Bilder neu erzeugt werden[20], der zweite Button bedeutet, daß veränderte Bilder die alten Bilder überschreiben.

Werden die Bilder neu erzeugt, so wird ihr Bildname aus dem alten Bildnamen wie folgt erzeugt:
ist der Name weniger als 7 (6 oder weniger) Zeichen lang, so wird _N an den alten Namen angehängt
ist der Name genau 7 Zeichen lang, so wird N angehängt.
ist der Name 8 Zeichen lang, so läßt sich nichts mehr anhängen; in diesem Fall wird das letzte Zeichen ermittelt, das noch keine Ziffer ist; endet der Name nicht auf eine Ziffer, so wird eben der letzte Buchstabe verwendet; enthält der Name nur Ziffern, so wird die erste dieser Ziffern herangezogen. Dieses letzte Zeichen wird nun eins weitergezählt (aus A wird B etc.). Z wird zu _, _ und 9 wird zu A.

Nach der Eingabe der zu ändernden Format-Parameter wird für alle Bildtypen der betroffenen Bilder, die die Funktion unterstützen[21], eine Dialogbox angezeigt, in der die zu ändernden Parameter und ihre neuen Werte eingeben werden können. Verläßt man eine solche Dialogbox mit Abbruch, so werden die Bilder dieses Types nicht neu berechnet, die Funktion wird aber nicht insgesamt abgebrochen. (Die Funktion kann auch teilweise unterstützt werden, in dem Sinne, daß eine Änderung der Parameter nicht möglich ist, und es somit auch keine Dialogbox zu diesem Zweck gibt, daß aber Änderungen des Bildformates möglich sind.)
Nach Eingabe der zu ändernden Parameter werden die Bilder mit den geänderten Parametern (soweit nötig) neu berechnet.

2.5 extended Neu, Block, ...

(eine blöde Überschrift, aber was blöderes ist mir nicht eingefallen)

Neben der Möglichkeit Bilder mit geänderten Parametern neu berechnen zulassen, indem man die Funktion Neu berechnen aufruft, gibt es für *einzelne* Bilder noch eine Alternative.
Man ruft dazu Neu berechnen auf, wobei nur ein Fraktal angesprochen sein darf, und drückt dabei die Control-Taste. Man landet dann nicht in dem Menüs für neu berechnen, sondern in denen für das Erzeugen neuer Bilder des entsprechenden Bildtypes, wobei aber die Daten des Fraktals voreingestellt werden. Man kann so alle Parameter – und nicht nur die im Neu-Dialog vorgesehenen – ändern und anschließend das Bild erzeugen. Dabei wird das Bild stets neu erzeugt (und zwar mit dem in den allgemeinen Parametern angegebenen Namen; voreingestellt ist dafür der Name des Bildes, das selektiert war), überschreiben ist nicht möglich. Der Effekt ist damit der gleiche, als hätte man das Berechnen von Bildern dieses Types aufgerufen, und dann in den allgemeinen Parametern wie in den speziellen Fraktalparametern mit der Funktion Daten holen die Daten des Bildes übernommen. Allerdings ist der Aufruf über Neu berechnen doch um einiges einfacher.
In analoger Weise kann man auch Blöcke ausschneiden oder Konstanten für Bilder vorgeben.
Dazu markiert man einfach in einem Bild- oder Anzeige-Fenster bei gedrückter Control-Taste einen Block oder eine Konstante (Doppelklick) analog zum Holen von Daten. Anschließend geht es weiter wie bei 'extended Neu'.
Noch eine Anmerkung: einige der Berechnungsroutinen unterscheiden beim Holen von Daten zwischen der Übernahme aller Daten und der nur eines Teiles (z.B. Fractale). Diese Unterscheidung gibt es auch hier, wobei zum Unterschieden (wie beim Holen der Daten) die Shift-Taste dient. Ohne Shift werden alle Daten übernommen. Der Status der Shift-Taste wird entweder unmittelbar vor oder alternativ nach der Eingabe der *allgemeinen* Parameter herangezogen. Die Möglichkeit, die Shift-Taste vor der Eingabe der allgemeinen Parameter zu drücken entfällt bei extended Neu im allgemeinen, da dann ja Neu für alle Objekte und nicht nur für das eine selektierte Objekt aufgerufen würde (eine bedauerliche Doppeldeutigkeit der Bedienung).
Weitere Anmerkung: die Möglichkeiten des extended Neu werden von allen Routinen, die auch das Übernehmen von Daten ermöglichen, unterstützt, derzeit also von allen (auch wenn die Routine neu berechnen selbst gar nicht unterstützt).

3 Bilder verwalten

Die weiteren Einträge im Extra-Menü dienen im weitesten Sinne der Verwaltung der Bilder (zum Menüpunkt neues Objekt siehe 4.7).

[20] natürlich nur, wenn ihre Parameter wirklich verändert wurden
[21] möglicherweise auch – wie bei Fractalen, s.u. – für mehrere Bildtypen gemeinsam

3.1 Film edieren

Dieser Menüpunkt dient dem Erzeugen neuer Filme und dem Ändern bestehender Filme gleichermaßen.
Wie schon gesagt, sind Filme in CHAOS*ultd* einfach geordnete Listen von Bildern.
Ein neuer Film wird erzeugt, indem man den Menüpunkt ohne selektierten Film aufruft. In einer Dialogbox kann (muß!) man dann den Filmnamen eingeben. Ist bei Aufruf der Funktion genau ein Film selektiert, so kann man diesen Film edieren. (Ist mehr als ein Film selektiert, kann der Filmeditor nicht aufgerufen werden.)
In beiden Fällen gelangt man in den Filmeditor, eine Dialogbox zum Edieren von Filmen.
Den größten Teil des Filmeditors bildet die Anzeige eines Bildausschnittes des aktuellen Bildes. Dabei wird die für die Bild-Anzeige eingestellte Konversion verwendet (vgl. Konvertieren der Anzeige). Die Anzeige läßt sich abschalten (dann wird einfach eine weiße Fläche angezeigt, so daß man nicht auf das Auspacken des Bildes warten muß), sie ist nur dann aktiv, wenn der Button 'Bild' rechts unten selektiert ist. Bei großen Bildern wird natürlich nicht das ganze Bild angezeigt, sondern nur die rechte obere Ecke.
Unter der Bild-Anzeige befindet sich eine Info-Zeile und zwei Zeilen mit Funktions-Button, ähnlich wie in CHAOS*ultd* V5.0x.
In der Infozeile stehen (von links nach rechts) der Filmname, die Anzahl der Bilder im Film, die Nummer und der Name des Bildes, das gerade angezeigt wird (gemeint ist die Nummer im Film, nicht die auf dem Desktop).
In der oberen Kommandozeile gibt es die folgenden Kommandos:
Die Button $<-$ und $->$ dienen zum Blättern im Film. Ein Einfachklick schaltet ein Bild weiter ($->$) oder zurück ($<-$), Doppelklick schaltet an den Filmanfang oder an das Ende.
Gewissermaßen vor dem Filmanfang mit Bild 1 wird noch ein (stets leeres) Bild Null verwaltet, beim Einfügen von Bildern werden diese nämlich nach dem aktuellen Bild eingefügt, um also ein neues Bild 1 einzufügen muß man auf Bild Null schalten.
Womit wir beim nächsten Punkt wären, dem Einfügen von Bildern. Dazu dient der Button Einfügen. Klickt man ihn an, so wird der Desktop angezeigt, und man kann das oder die Bilder angeben, die eingefügt werden sollen. Einzelne Bilder doppelklickt man einfach an. Darüberhinaus kann man aber auch durch `Shift`-Klick / Doppelklick mehrere aufeinanderfolgende Bilder in den Film übernehmen, wobei die Reihenfolge der beiden Mausklicks berücksichtigt wird (vgl. Eingabehilfen, 2.2). Doppelklick neben ein Bild bricht die Funktion ab, ohne ein Bild in den Film aufzunehmen.
Die gegenteilige Wirkung, nämlich das Löschen von Bildern übernimmt der Button Löschen. Mit ihm wird das aktuelle Bild aus dem Film gelöscht.
Mit dem nächsten Button Optionen kann man die Anzeigeoptionen des Films einstellen. Sie entsprechen den Optionen beim Anzeigen von Bildern, jeder Film verwaltet allerdings eigene Optionen (die beim Erzeugen des Filmes mit den gerade eingestellten Anzeigeoptionen für Bilder vorbelegt werden). Dies ist die einzige Möglichkeit, die Anzeigeoptionen im Film zu ändern. Änderung der Anzeigeoptionen beim Zeigen von Filmen (s.u.) führen nur zur Änderung für dieses Anzeigen.
In der unteren Kommandozeile befinden sich die Button für die Blockfunktionen. Mit Einfachklick auf Start und Ende kann man Blockanfang und -ende festlegen, wobei Start natürlich vor Ende liegen muß. Sind die Blockmarken gesetzt und die jeweiligen Bilder als aktuelles Bild eingestellt, so erscheinen diese Button invers; mit Doppelklick auf Start oder Ende kann man zu der entsprechenden Marke springen.
Es folgen die Blockkommandos Copy, Move und Delete, die sich wohl selbst erklären. Das Kommando Reverse dreht die Reihenfolge der Bilder eines Blockes um.
Die maximale Zahl von Bildern, die ein Film enthalten kann, ist auf 32767 begrenzt. Die maximale Zahl von Bildern, die der Filmeditor verwalten kann, gibt man in den Einstellungen an (256-32767).
Verlassen wird der Filmeditor wie üblich mit Ok oder Abbruch.

3.2 Bilder packen

CHAOS*ultd* ist in der Lage, Bilddaten im Speicher zu komprimieren (packen), so daß sie weniger Speicherplatz belegen. Andererseits müssen gepackte Bilder vor dem Anzeigen entpackt werden, was natürlich Zeit braucht. Deshalb kann man einerseits in den Einstellungen wählen, ob Bilder gepackt werden sollen oder nicht und andererseits gepackte Bilder auspacken oder nicht gepackte Bilder packen lassen. Zum letzteren dient der Menüpunkt Bilder packen, man kann hier Bilder packen aber auch entpacken.
Nach Anwahl des Menüpunktes Bilder packen erscheint eine Dialogbox, in der angezeigt wird, wieviel Speicher die entpackten Bilder, die gepackten Bilder und die gepackten Bilder entpackt belegen. Außerdem wird die Größe des freien Speichers angezeigt.
Verlassen kann man die Box außer mit Abbruch mit Packen oder Entpacken, so daß man nicht gepackte Bilder packen oder gepackte Bilder entpacken kann. Während des Packens/Entpackens eines Bildes wird ein Info-Dialog angezeigt, zwischen zwei Objekten ist der Abbruch der Funktion mittels der `Esc`-Taste möglich.

3.3 Bilder konvertieren

Die Funktion konvertieren erlaubt es, Bilder zu skalieren (vergrößern oder verkleinern) und mehrfarbige Bilder in zweifarbige (monochrome) Bilder umzurechnen, die Farben werden dann durch Grauraster dargestellt.
Die Funktion wirkt auf beliebige Bildobjekte, das Ergebnis wird in neuen Bild-Objekten abgelegt, deren Namen änlich wie bei neu berechnen erzeugt wird, nur daß statt des 'N' ein 'K' zum Markieren verwendet wird.
Beim Skalieren hat man die Möglichkeit, die Bilder um einen Faktor 2 zu vergrößern (Seitenlänge!) oder um Faktoren 1/2, 1/4 oder 1/8 zu verkleinern. Außerdem kann man noch angeben, ob das eventuell existierende Raster eines Bildes beim Skalieren beachtet werden soll (sonst wird auf Pixelebene skaliert). Die Größe des Zielbildes wird dabei auf ein Vielfaches von 16 (Breite) bzw. 4 (Höhe) abgerundet.
Beim Konvertieren gibt es mehr Möglichkeiten:
Konvertiert werden prinzipiell nur Bilder mit mehr als eine Bitmap (vier oder mehr echte Farben).
Die einfachste Methode ist das Auswählen von Bitmaps. Dabei gibt man an, welche Bitmaps ins Zielbild kopiert werden sollen, wobei natürlich nur solche Bitmaps berücksichtigt werden, die im Originalbild auch vorhanden sind. Damit wird gleichzeitig die Zahl der Bitmaps im Zielbild festgelegt; soll gar keine der Original-Bitmaps verwendet werden, so wird *eine*, leere Bitmap angelegt.
Als Spezialfall kann man statt alle Bitmaps anzuwählen auch den ersten mit x bezeichneten Button selektieren, dann werden alle Bitmaps kopiert, das Bild also nicht konvertiert.
Die anderen (echten) Konvertierungsroutinen konvertieren grundsätzlich farbige Bilder in monochrome.
Dabei muß zunächst zwischen grundsätzlich unterschiedlichen Algorithmen unterschieden werden, der Fehler-Diffusion (Floyd-Steinberg), die in CHAOS*ultd* auch mit Graurastern arbeitet, und dem (geordneten) Dithern. Am besten probiert man die verschiedenen Algorithmen einfach aus, und schaut sich die Ergebnisse an, jedenfalls werde ich mich jetzt nicht zu weitschweifigen Erläuterungen aufschwingen.
Bei der Fehlerdiffusion kann man wählen, ob der Fehler auf 6, 12 oder 0 Punkte verteilt werden soll (im letzten Fall findet keine Fehlerdiffusion mehr statt, so daß man direkt in Grauraster übersetzen lassen kann), und ob Grauraster verwendet werden sollen (dann wird das Bild dadurch nochmals vergrößert!). Möglich sind 1x1 Raster (also Pixel, zwei Farben), 2x2 Raster (5 Farben) und 4x4 Raster (17 Farben).
Weiter kann man angeben, ob die Farben absolut oder relativ umgesetzt werden sollen, im ersten Fall ist schwarz schwarz, grau grau und weiß weiß, im letzten Fall wird die dunkelste Farbe des Quellbildes zu schwarz (auch wenn sie grau ist), die hellste zu weiß. Dadurch werden zwar die Farben verfälscht, dafür ist der Kontrast im Ergebnis besser.
Der Dithering-Algorithmus (orderd dither) stellt (bisher) keine weiteren Optionen zur Verfügung.
Für beide Algorithmen muß noch die Umsetzung der Farbe angegeben werden. Im RGB-Modus gehen der Rot-, Grün- und Blau-Anteil gewichtet ein, im HLS-Modus wird die Lightness L, also das Maximum des Rot-, Grün und Blau-Anteiles, verwendet (Achtung, L kann leicht für verschiedene Farben gleich sein). Wählt man Ignorieren der Farbpalette, so wird den Farben einfach ein Graukeil zugeordent, von Schwarz für Farbe 0 bis Weiß für die letzte Farbe.
Verläßt man den Dialog mit Ok, so werden die Bilder zuerst skaliert, dann konvertiert, wobei Konvertieren bei Bildern, die schon monochrom sind, ausgelassen wird. Gleichzeitig wird die aktuelle Einstellung der Konversion in die Voreinstellungen übernommen, d.h. beim nächsten Aufruf des Konvertierens von Objekten findet man die Einstellung vom letzten mal wieder vor.
Verkleinert man Bilder, so wird die Größe des Zielbildes so abgerundet, daß die Breite (Höhe) wieder durch 16 (4) teilbar ist. Wird das Zielbild schmaler als 64 Pixel oder weniger hoch als 40 Pixel, so kann nicht skaliert werden und es wird keine neues Objekt erzeugt.
Achtung! Insbesondere das Konvertieren kann recht lange dauern (Minuten). Die Möglichkeit des Abbrechens mit Esc ist nur *nach dem Ende* des Konvertieren eines Bildes möglich, falls man mehrere Bilder konvertieren läßt (analog etwa zum Packen oder Laden).

3.4 Löschen

Möchte man Objekte löschen, so selektiert man diese und klickt anschließend den Menüpunkt Objekte löschen im Datei-Menü an. (Man kann sich das selektieren auch sparen, dann werden aber alle Objekte gelöscht).
Nach einer eventuellen Warnung (vgl. Einstellungen, 6.2) werden die Objekte gelöscht.
Es ist nicht möglich, Bilder, die sich in einer Bildliste für ein Bild-Fenster befinden, die in Filmen enthalten sind, oder die gerade im Filmeditor aufgelistet werden, zu löschen; versucht man dieses, so wird man in einer Dialogbox darauf hingewiesen. Bezieht sich Löschen nur auf selektierte Bilder, so kann man fortfahren, indem man weiter wählt. Dann werden die Bilder der Bildliste, der Filme oder des Filmeditors vom Löschen ausgenommen.

4 Anzeigefunktionen

4.1 Bilder anzeigen

Wenn man Bilder einmal erzeugt hat, dann will man sie natürlich auch anschauen. Dazu gibt es in CHAOS*ultd* (mal wieder) zwei prinzipielle Möglichkeiten: das Anzeigen in einem (Bild-)Fenster und das Anzeigen auf dem ganzen Bildschirm. Letzteres entspricht (in etwa, d.h. modulo interner Details, die sich aber auf Geschwindigkeit etc. auswirken) der Anzeige in der Version 5.0, insbesondere mit der Möglichkeit Bilder automatisch weiterschalten zu lassen. Allerdings sind die diversen Optionen beim Anzeigen, fast ausschließlich von der Anzeige im Fenster bzw. vom Desktop aus erreichbar (wie das Anzeigen der Parameter, Konstante anzeigen, Koordinaten zeigen ...). In der Ganzseiten-Anzeige kann man nur noch die Anzeigeoptionen einstellen.
Bilder, die die größer als der Bildschirm sind, können natürlich nicht ganz angezeigt werden. Bei ihnen kann mit den Cursor-Tasten (bzw. den Scrollbalken der Fenster) gescrollt werden.
Ein weiteres Problem ist die Anzeige von Bildern mit mehr Farben (Bildebenen) als auf dem Bildschirm. Dazu kann man einstellen, wie die Bilder konvertiert werden sollen. Prinzipiell hat man hier die gleichen Möglichkeiten (und Einschränkungen) wie beim Konvertieren von Objekten (vgl. 3.3)[22]. Man sollte aber zusätzlich den Zeitbedarf für das Konvertieren bedenken (auf einem normalen 8Mhz ST ist wohl nur die Auswahl von Bitmaps sinnvoll, lediglich die Skalierungsroutinen sind mittlerweile optimiert). Eingestellt wird die Anzeigekonversion mit dem Menüpunkt Anzeige konvert. im Optionen-Menü (die Einstellung gilt gleichzeitig auch für den Filmeditor). Beim Konvertieren der Anzeige wird – wie beim Konvertieren von Objekten – die Größe des Zielbildes modulo 16 bzw. 4 abgerundet. Hier darf das Bild aber auch kleiner als 64x40 Pixel werden. Verschwindet die Größe des Bildes (1/8 von 64 ist 8, modulo 16 abgerundet bleibt nichts übrig), so kann nicht skaliert werden und es wird das Originalbild unskaliert verwendet.
Für die Anzeige der Farben wird die Farbtabelle des Bildes verwendet, sofern nicht konvertiert und auch keine Bitmaps ausgewählt wurden (d.h. bei Standard-Bitmapauswahl von Bildern mit nicht mehr Farben als auf dem Bildschirm dargestellt werden können). Andernfalls wird für die Anzeige eine eigene Farbtabelle erzeugt, und zwar eine Standard-Schwarz-Weiß-Tabelle bei (echter) Konversion. Bei der Auswahl von Bitmaps wird die Tabelle entsprechend den ausgewählten Bitmaps aus der Original-Tabelle zusammengestellt.
Bei der Anzeige im Fenster führt die Anzeige der Farben des Bildes natürlich zur Änderung der Farben auf dem ganzen Desktop. Da dies bei Aufruf von Dialogboxen unter Umständen sehr unpraktisch ist, kann man in den Einstellungen drei Modi für das Einstellen der Farben von Bild- und Anzeige-Fenstern wählen: im einfachsten Fall bleiben die Farben in jedem Fall entsprechend dem angezeigten Bild eingestellt. Alternativ gibt es die Möglichkeit, beim Aufruf von Dialogboxen die Farben auf die des Desktops umzustellen. Nach Verlassen der Dialogbox werden dann wieder die Farben des Bildes angezeigt. Als dritte Möglichkeit kann man noch einstellen, daß die Farben des Bildes nur solange angezeigt werden, wie sich der Mauszeiger im Fenster des Bildes befindet (dies schließt die Farbanpassung für Dialoge natürlich ein).

Die Anzeige im Fenster wird durch den Menüpunkt Fenster im Zeigen-Menü aufgefufen, die Anzeige auf dem ganzen Bildschirm durch den Menüpunkt Bildschirm.
In jedem Fall wird zunächst intern eine Bildliste aller betroffenen Bilder erstellt, die im weiteren die im Fenster dargestellten Bilder festlegt. Diese Bildliste ist unabhängig von späteren Veränderungen der Objekte, sie bleibt unverändert, auch wenn man zusätzliche Bilder selektiert oder deselektiert (was ohnehin nur beim Anzeigen in einem Fenster möglich ist, die Ganzseitendarstellung ermöglicht natürlich keinen Zugriff auf das Objekt-Fenster).

Anzeigeoptionen

CHAOS*ultd* verwaltet für jede Bildliste (also für jedes Bild-Fenster und jeden Film) eigene Anzeigeoptionen; bei deren Einstellen kann man wählen, ob die Einstellung nur für diese lokalen Optionen gelten sollen, oder ob sie auch global übernommen werden sollen, was aber nur heißt, daß sie dann auch für die *im folgenden* geöffneten Bild-Fenster gelten.
Einstellen lassen sich die Anzeigeoptionen zusammen mit weiteren Parametern unter dem Menüpunkt Optionen.
Zunächst hat man den Anzeigemodus, den man auf Warten auf Taste und Kontinuierlich einstellen kann.
Für kontinuierliche Anzeige kann man darunter die Wartezeiten (nach einem Bild und am Ende der angezeigten Bilder) einstellen, wobei man die Zeiten sowohl numerisch (in ms)[23] wie auch mit Hilfe eines Sliders wählen kann (dabei dient das Anklicken des grauen Balkens nicht zum Verschieben des Sliders, sondern der Slider wird entsprechend des als Text angezeigten Wertes gesetzt).
Die Wartezeit muß mindestens auf 20 ms gesetzt werden, da es sonst zu Bildschirmflackern kommt; gibt man kleinere Zeiten an, so wird die Wartezeit automatisch auf diesen Wert gesetzt.
Desweiteren kann man noch einstellen, ob die Bilder nur einmal oder wiederholt angezeigt werden sollen (nur dann findet die Wartezeit am Anzeigeende Verwendung), und ob im zweiten Fall dann wieder mit dem ersten Bild ange-

[22] allerdings werden bei der Auswahl von Bitmaps nur so viele Bitmaps kopiert, wie für die Bildschirmdarstellung zur Verfügung stehen
[23] aus technischen Gründen nur in 5 ms Schritten, ist der Wert nicht durch 5 teilbar wird er abgerundet;

fangen werden soll ('− > − >') oder ob die Bilder abwechselnd vor- und rückwärts durchgegangen werden sollen ('− > < −').

automatisches Weiterschalten in Fenstern: Hier gibt es einige Einschränkungen. CHAOSultd schaltet maximal alle 20ms *ein* Fenster weiter. Dabei werden alle 20ms alle Fenster daraufhin untersucht, ob sie weitergeschaltet werden sollen, und das erste dabei gefundene Fenster weitergeschaltet. Bei kleinen Wartezeiten, großen Bildern (deren weiterschalten halt nu mal länger dauert) und langsamen Rechnern (wie meinem) führt dies dazu, daß bei mehreren automatisch weiterzuschaltenden Bildern nur das (oder die) ersten zum Zuge kommen, während die anderen gewissermaßen verhungern. Die Einschränkung, jeweils nur ein Fenster weiterzuschalten, dient dazu, daß man das Programm trotz Weiterschaltens von Bildern einigermaßen vernüftig bedienen kann.

Konvertieren der Anzeige

Neben der Einstellung der Anzeigekonversion für alle Anzeigen ist es auch möglich, die Anzeige einzelner *angezeigter* Bilder zu konvertieren. Diese Funktion kann nur über die Popup-Menüs in Bild- und Anzeige-Fenstern (Menüpunkt konvertieren aufgerufen werden. In der üblichen Dialogbox für die Konvertierungsparameter kann man die Konversion einstellen und die Anzeige wird dann konvertiert. Die konvertierte Anzeige bleibt solange erhalten bis man entweder erneut konvertieren für diese Anzeige aufruft (die neuerliche Konversion bezieht sich wieder auf das Ausgangsbild, *nicht* auf das Ergebnis der letzten) oder bis man die Anzeige weiterschaltet oder sonstwie erneuert (z.B. nach Weiterberechnen des Bildes). Es ist möglich, für das konvertierte Bild die Ganzseitenanzeige aufzurufen.

4.2 Bildvergleich

Mit der Funktion Bildvergleich können Bilder (genauer die Information in den Bitmaps der Bilder) durch logische Verknüpfung verglichen werden.
Wird die Funktion aufgerufen, so muß man zunächst in einer Dialogbox angeben, welche Verknüpfung (and, or oder xor) ausgeführt werden soll. Sind die Bilder verschieden groß, so bestimmen die maximalen Abmessungen die Größe des resultierenden Bildes. In diesem Fall kann man noch wählen, ob die kleineren Bilder relativ zu diesem Größten zentriert oder im linken oberen Eck eingesetzt werden sollen. Die Farbtiefe des Ergebnisses bestimmt sich ebenfalls aus der des Bildes mit der größten Farbtiefe, als Farbtabelle wird die des ersten Bildes mit dieser maximalen Farbtiefe verwendet. Nach dem Verlassen des Dialoges werden die Bilder verknüpft und das Ergebnis – gegebenfalls konvertiert – in einem Anzeige-Fenster dargestellt.
Man beachte, daß es je nach Bilderzahl und Bildgröße unter Umständen etwas länger dauern kann, bis die Verknüpfung der Bilder durchgeführt wurde und das Anzeige-Fenster geöffnet wird.
Bildvergleiche zweier Bilder kann man mit der Verknüpfung xor durchführen; dann bleiben im resultierenden Bild genau die Teile der Bilder übrig, die *nicht* gleich sind (mit der Verknüpfung and kann man umgekehrt die Bildteile erhalten die gleich sind).

4.3 Block zeigen

Mit der Funktion Block zeigen kann man sich anzeigen lassen, welchem Bildausschnitt ein oder mehrere andere Bilder in einem (Mutter-)Bild entsprechen.
Wie beim Bildvergleich wird ein Anzeige-Fenster geöffnet, in dem das Ergebnis angezeigt wird. Ergebnis ist in diesem Fall das Bild, das den größten Ausschnitt besitzt, in diesem sind die Ausschnitte, die die anderen Bilder darstellen als Rechtecke markiert. (Ob es sich dann wirklich um Blöcke handelt ist natürlich eine andere Frage). Bei gedrehten Bildern sind die Rechtecke natürlich gedreht, unter Umständen auch verzerrt.
Die Funktion wird nicht von allen Bildtypen unterstützt. Es muß zum einen ein Koordinatensystem existieren, damit man überhaupt von einem Ausschnitt sprechen kann, und dieses Koordinatensystem muß in einfacher Weise auf die Bitmap abgebildet sein, damit Blöcke markiert werden können. (Die erste Voraussetzung wird etwa von L-Systemen verletzt, 3d-Fractale erfüllen zwar die erste, nicht aber die zweite Voraussetzung.)

4.4 Konstante anzeigen

Mit dieser Funktion kann man sich in einem Bild die Lage einer durch die Parameter bestimmten Konstante (zweikomponentig, x- und y- Komponente bzw. Real- und Imaginärteil) anzeigen lassen.
Abgesehen davon, daß die Funktion nur auf ein Bild wirkt, ist sie analog zum Block anzeigen.
Man beachte, daß das Fadenkreuz, mit dem die Konstante markiert wird, bei gedrehten Bildern nicht parallel zur x- bzw. y-Achse des Bildes ist.

4.5 Koordinaten anzeigen

In allen Bildern, zu denen ein Koordinatensystem gehört, kann man – statt des einfachen Bildinfos – die Koordinaten am Ort des Mauszeigers anzeigen lassen.
Zum Umschalten zwischen einfachem Bildinfo und Koordinatenanzeige dient der Menüpunkt Koordinaten anzeigen.
Die Koordinaten werden nur angezeigt, solange sich der Mauszeiger über dem *aktiven* Bild- oder Anzeigefenster befindet. Man beachte auch, daß nicht bei allen Bildern Koordinaten angezeigt werden können.

4.6 Punktinfo

Durch Doppelklick in ein Bildfenster kann man sich ein Punktinfo zu den angegebenen Koordinaten anzeigen lassen (sofern von den Routinen unterstüzt, bisher nur bei Fractalen).

4.7 neues Objekt

Bei allen Bild- und Anzeige-Fenstern kann man mit der Funktion neues Objekt im Extra-Menü ein neues Bild-Objekt erzeugen, das das angezeigte Bild enthält.
Dabei gibt man in einer Dialogbox den Namen des neuen Objektes an und wählt, ob man das Original oder die Anzeige in das neue Objekt übernehmen will. Der Unterschied ist, daß im zweiten Fall die tatsächliche Anzeige, die unter Umständen nicht alle Bitmaps umfaßt oder konvertiert wurde, im ersten Fall das dieser Konversion (oder Bitmap-Auswahl) zugrundeliegende Bild verwendet wird. Für Bild-Fenster wird Anzeige für Anzeige-Fenster Original voreingestellt. Verläßt man den Dialog mit Ok so wird anschließend das Objekt erzeugt.
Es ist nicht möglich ein neues Objekt zu erzeugen, wenn es kleiner als 64x40 Pixel groß ist (kann durch skalieren für die Anzeige vorkommen).

4.8 Bild-Parameter anzeigen

Unter Bildparameter im Zeigen-Menü kann man sich die allgemeinen Bildparameter eines Bildes anzeigen lassen. Dabei muß man zwischen den Parametern eines Bild-Objektes und einer Bild-Anzeige unterscheiden. Wird der Menüpunkt für ein Bild-Objekt aufgerufen (entsprechendes Popup-Menü, aktives Objekt-Fenster ...), so werden die Parameter des Bild-Objektes angezeigt, ist ein Bild- (oder Anzeige-)Fenster im Vordergrund, die Parameter der Anzeige.
Angezeigt wird

- Bildname und Dateipfad

  Name und Pfad können ediert oder durch Anklicken des Quadrates rechts über dem Pfad mittels der Dateiselectorbox geändert werden

  Bei Anzeigen wird der Name des zugehörigen Objektes angegeben, gibt es kein solches, so entfällt die Angabe. Der Pfad wird in jedem Fall ignoriert. Änderungen werden nicht übernommen.

- Bildtyp, etwa Bild (für einfaches Bild) oder der Fraktaltyp

- Bildgröße

  hier wird die Größe in Bild-Punkten angegeben. bei unfertigen, zeilenweise berechneten Bildern wird neben der Höhe des fertiggerechneten Bildes in Klammern die aktuell vorliegende Größe des berechneten Teilbildes angegeben

  Außerdem wird die Zahl der Farben angegeben

- Größe der Bitmap (Breite mal Höhe mal Zahl der Bitmaps)

  kann bei Verwendung von Rastern von der Bildgröße abweichen, dann wird bei der Bildgröße ein Rasterpunkt als *ein* Punkt, bei der Größe der Bitmap als 2x2 bzw. 4x4 Punkte gezählt

- nur bei Objekten: Speicherbedarf des gespeicherten (meist also gepackten) Bildes und Speicherbedarf des vollständig berechneten entpackten Bildes

- Anstelle des Speicherbedarfs wird bei Anzeigen die Konversion der Anzeige ausgegeben.

Änderungen des Bildnamens und Pfades werden mit Ok übernommen, mit Abbruch verworfen.

4.9 Farben

Mit diesem Menüpunkt kann man den Farbselektor für ein Objekt oder ein Anzeigefenster aufrufen.
Wird der Selektor für ein Anzeigefenster aufgerufen, so werden die Farben des Objektes nur geändert, wenn direkt die Objekt-Farbtabelle zur Anzeige der Farben verwendet wird.
Wird der Selektor für ein Objekt aufgerufen, das gerade angezeigt wird, und wird für die Anzeige dieses Objektes direkt die Objekt-Farbtabelle verwendet, so ändert sich auch die Farbe der Anzeige, die aber nicht unbedingt *automatisch* aktualisiert wird.
Worauf sich die Farbselektion jeweils bezieht wird in der Infozeile im Farbselektor angegeben.

4.10 Fraktal-Parameter anzeigen

Mit dem Menüpunkt Parameter im Zeigen-Menü kann man sich die Parameter eines Fraktals anzeigen lassen. Details sind Sache der Berechnungsroutinen.

4.11 Info anzeigen

Unter Info kann man eine Dialogbox aufrufen, in der man einige mehr oder weniger (eigentlich weniger) interessante Informationen über die berechneten Bilder anschauen kann, im wesentlichen die Berechnungszeit, die Zahl der berechneten Iterationen und die Zahl der gezeichneten Bildelemente[24].
Ist genau ein Fraktal selektiert, so kann die Berechnungsroutine für das Bild eine detaillierte Info-Box ausgeben, die dann natürlich von der Berechnugsroutine abhängt.
Stellt die Berechnungsroutine eine solche Funktion nicht zur Verfügung, unterstützt aber die Ausgabe von Bildinfos[25], oder sind mehrere Bilder betroffen, so wird eine Info-Box angezeigt, in der die Anzahl der Bildpunkte, die Zahl der berechneten Iterationen[26] sowie die benötigte Rechenzeit – insgesamt und pro Bildelement bzw. Iteration – ausgegeben wird.

5 Input/Output

5.1 Speichern

Mit den Menüpunkten Speichern und Speichern ... kann man Objekte abspeichern.
Für Fraktale und Filme gibt es ein eigenes Dateiformat, Bilder können im GEM-Image-Format (inclusive XIMG-Header) und (aus Gründen der Tradition) im Stad-Format gespeichert werden; beim Stad-Format wird nur eine Bitplane (wie beim Anzeigen auf einem monochromen Bildschirm) und gegebenfalls nur die linke obere Ecke gespeichert. Fraktale lassen sich ebenfalls in diesen Bildformaten speichern, dabei werden aber die Bild-Parameter nicht mit abgespeichert, so daß als Bild gespeicherte Fraktale *nicht* als gespeichert betrachtet und gekennzeichnet werden.
Speichern ... unterscheidet sich von Speichern dadurch, daß man zunächst in der Fileselectorbox ein Dateipfad und optional ein Extension angeben kann. Der Pfad wird dann zum Defaultpfad, der für alle Objekte vorgeschlagen wird; mit dem Extension kann der Bildtyp gewählt werden (*PAC* Stad, *IMG* GEM-Image). Ist ein Extension angegeben, so werden auch Fraktale defaultmäßig in diesem Format gespeichert.
Im Modus Speichern ist der Defaultpfad der in den Einstellungen angegebene, er wird aber nur verwendet, wenn der Pfad eines Bildes noch nicht gesetzt ist (das ist dann der Fall, wenn das Bild neu erzeugt wurde und nicht geladen).
Für jedes zu speichernde Objekt wird nun zunächst eine Dialogbox aufgebaut, in der die Nummer des Objektes, der Objekttyp (Fraktal oder Bild) und der Objektstatus (geändert oder unverändert, d.h. gespeichert) angezeigt wird (die Anzeige der Box kann in den Einstellungen abgestellt werden, s.u.).
Unter diesen zur Information dienenden Angaben befindet sich der Bildname und der Dateipfad, außerdem wird das Dateiformat angezeigt. Dieses kann man durch Anklicken eines der Button ändern, es wird dann automatisch das entsprechende Extension in den Namen eingetragen. Mit dem Button rechts über dem Pfad können Namen und Pfad via Dateiselectorbox geändert werden, natürlich ist dies auch direkt mit der Tastatur möglich.
Der Dateityp, unter dem ein Bild oder auch ein Fraktal abgespeichert wird, wird durch das hier eingestellte Dateiextension festgelegt. *CHS* ist nur bei Fraktalen möglich und bedeutet das Abspeichern als Fraktal; mit *PAC* oder *IMG* wird das Fraktal oder das Bild im entsprechenden Dateityp gespeichert. Andere Extension sind nicht möglich; versucht man mit einem anderen Extension zu speichern, so korrigiert CHAOS*ultd* das Extension, gibt eine Meldung aus und verläßt die Dialogbox nicht.

[24] Bildelemente sind meist Punkte, es kann sich aber auch um Linien handeln
[25] indem sie dem Verwaltungsteil Informationen über Berechnungszeit sowie Iterations- und Bildelement-Anzahl zur Verfügung stellt
[26] was im einzelnen unter einer Iteration zu verstehen ist liegt bei der Berechnungsroutine; korrekter wäre vieleicht auch die Bezeichnung Berechnungsschritte – nicht alle Routinen sind iterativ

Zum Verlassen der Box stehen vier Button zur Verfügung: bei Speichern wird das Bild gespeichert und die Dialogbox für das nächste Bild angezeigt, weiter geht zum nächsten Bild über ohne das aktuelle Bild zu speichern, Abbruch bricht den ganzen Speicherprozess ab.
Bleibt noch Info aus. Mit diesem Button kann man veranlassen, das für die noch folgenden, zu speichernden Bilder keine ausführliche Infobox mehr ausgegeben wird. CHAOS*ultd* zeigt dann nur noch den Namen der zu speichernden Bilder dar, es werden automatisch die Voreinstellungen zum Speichern verwendet. (Dies läßt sich auch von Anfang an erreichen, indem man in den Einstellungen die Nachfrage bei save abstellt. Es ist aber zumindest anfangs zu empfehlen, wenigstens beim ersten Bild zu überprüfen, ob die Speicherfunktion auch das tut, was man wollte.). In diesem Fall kann das Abspeichern durch Drücken der Esc-Taste abgebrochen werden.
Kommt es beim Speichern zu einem Namenskonflikt (eine Datei soll erzeugt werden, die schon besteht) so wird dies (abhängig von den Einstellungen des Programms) in einer der Dialogbox beim Speichern sehr ähnlichen Box angezeigt. Man kann hier den Namen nochmals ändern, und die Box wiederum auf viererlei Weise verlassen. Überschreiben speichert das Objekt mit dem eingestellten (eventuell geändertem) Namen ohne sich um eine eventuell bestehende Datei zu kümmern, Weiter und Speichern arbeiten wie oben beschrieben; Msg. aus schaltet analog zu Info aus die Warnung von Überschreiben ab.
Gespeicherte Objekte werden auf dem Desktop abgehakt dargestellt, Fraktale, die nur als Bilder gespeichert wurden gelten wie gesagt nicht als gespeichert.
Filme werden erst nach allen anderen Objekten gespeichert. Sie können prinzipiell erst gespeichert werden, wenn alle in ihnen enthaltenen Objekte gespeichert sind; andernfalls erhält man eine Fehlermeldung.

5.2 Laden

Mit dem Eintrag Laden im Menü Datei kann man Objekte wieder einladen.
Es erscheint die Dateiselectorbox, mit der einzelne oder auch mehrere Bilder ausgewählt werden können.
Gibt man explizit einen Dateinamen an, so wird diese eine Datei geladen.
Gibt man aber keinen Dateinamen an, so werden alle in der Dateiselectorbox gerade dargestellten Dateien, der unterstützten Dateiformate (also alle Dateien mit den Extension *CHS*, *PAC* und *IMG*) geladen. Das Einladen läßt sich durch Drücken der Esc-Taste abbrechen.
Beim Laden von Filmen (Extension *FFX*) werden die in diesen enthaltenen Bildern – sofern sie nicht bereits geladen sind – automatisch nachgeladen.
Seit V6.40 kann CHAOS*ultd* auch Bilder von V5.0x (außer Feigenbaumdiagrammen) mit Parametern (und nicht nur als Bild) laden.

5.3 Drucken

CHAOS*ultd* verfügt über eine Schnittstelle zu externen Hardcopy-Routinen. Hardcopy-Routinen haben die Endung *HCP* (für HardCoPy) und müssen sich wie *XCH*-Routinen im Ordner *CHS_ULTD.GEM* befinden. Von den dort befindlichen Routinen wird nur die (physikalisch) erste Routine geladen. Bisher existiert nur eine Routine für 24 Nadel-Matrixdrucker, die im letzten Teil dieser Anleitung beschrieben wird.
Zum Aufruf der Routine existieren die Menüpunkte die Menüpunkte Bilder drucken und Druckparameter.
Zur Unterstüzung der Hardcopy-Routine verfügt CHAOS*ultd* über eine eigene, schnelle Ausgaberoutine auf die Centronics-Schnittstelle. Da sich solche Routinen eigentlich überhaupt nicht gehören, kann man diese abschalten. In den Einstellungen kann man die CHAOS-Routine zur Ausgabe an den Drucker ein- (Button selektiert) oder ausschalten.
Die eigene Routine hat den Vorteil, daß sie schneller ist. Allerdings werden Druckerspooler umgangen (sind also wirkungslos) und das Ausdrucken über serielle Schnittstelle ist nicht möglich (falls man nicht über die CHAOS*ultd*-Routine druckt, sondern über TOS, so wird je nach Einstellung der Druckerparameter (Kontrollfeld!) parallel bzw. seriell gedruckt, letzteres ist nicht getestet). Wer also seriell drucken will oder einen Spooler für sinnvoll hält (ist er bei Graphikdruck eigentlich nicht), der schalte die CHAOS*ultd*-Routine ab.
Außerdem habe ich meine Druckroutine natürlich nur mit meinem Rechner und meinem Drucker getestet. Ob sie in allen Fällen korrekt arbeitet weiß ich nicht.

5.4 Clipboard

CHAOS*ultd* unterstützt das Clipboard, obwohl ich nicht so recht weiß, wofür man das eigentlich braucht (naja, ich will nicht ungerecht sein, in einer Multitasking-Umgebung mag es schon seine Berechtigung haben).
Die Unterstützung ist allerdings nicht vollständig, mit dem Clipboard kann nur gearbeitet werden, wenn der Clipboard-Pfad gesetzt ist (z.B. von einem Clipboard-Accessory oder von Gemini oder von was auch immer; CHAOS-*ultd* selbst setzt den Pfad jedenfalls nicht). Ist der Clipboard-Pfad nicht gesetzt, so brechen die Clipboard-Funktion ohne Fehlermeldung ab.

Man kann einerseits Bilder (nein *ein* Bild, das ist ja das Dumme am Clipboard) aus dem Clipboard laden, sofern eine *SCRAP.IMG* Datei im Clipboard liegt. Ist keine Datei vorhanden so erscheint eine Fehlermeldung, daß kein Bild im Clipboard liegt.
Zum anderen kann man ein Bild (wenn genau eins selektiert ist) im Clipboard als *SCRAP.IMG* Datei (und nur als solche) ablegen. Alle eventuell vorhandenen *SCRAP*-Dateien werden gelöscht.

6 Einstellungen

Die Einstellungen sind aus Platzgründen auf zwei Dialogboxen verteilt.

6.1 Einstellungen I

Hier finden sich die Einstellungen für Dialogfenster, Berechnung und Ränder beim Anzeigen. Im einzelnen kann man wählen:

- Dialoge in Fenster legen
- Dialoge zentrieren
- Info-Dialoge in Fenster legen (gilt nur für die Dialoge beim Berechnen; die Info-Dialoge beim Laden/Speichern werden nicht in Fenster gelegt)
- die Positionierung der Info-Dialoge, an der letzten Position, zentriert oder rechts oben
- Darstellung des Randes beim Anzeigen auf dem Bildschirm: weiß, grau oder schwarz
- Darstellung des Randes beim Anzeigen in Fenstern
- Berechnung von Bildern anzeigen oder verdeckt berechnen
- Puffern der Ausgabe auf den Bildschirm
- Gong am Berechnungsende; der Button hinter dem Text, dient zum Testen des Gonges

  Anmerkung: der Gong wird durch Ausgabe des Zeichens 7 (nicht die Ziffer **7**, sondern der ASCII Code 7) via Gemdos erzeugt. Das ist der selbe Gong, der (ensprechende Einstellung im Kontrolfeld vorausgesetzt) auch ertönt, wenn man neben eine Dialogbox klickt. Dazu ist es nötig, daß in den Systemvariablen das entsprechende Flag gesetzt ist, das man eben auch löschen kann, beispielsweise mit dem Kontrolfeld. Deswegen setzt CHAOS*ultd* diese Flag beim Programmstart und setzt es beim Verlassen wieder auf den alten Wert zurück. Ändert man nun während CHAOS*ultd* läuft mit einem Accessory dieses Flag, so wird erstens (beim Löschen des Flags) der Gong beim Berechnungsende nicht funktionieren und zweitens diese Änderung beim Verlassen von CHAOS*ultd* überschrieben. Das Ein- oder Ausschalten des Gongs schaltet auch die drei Gongs ein bzw. aus, mit denen auf eine Garbage Collection hingewiesen wird.
- die Druckroutinen zur Ausgabe auf die Centronics-Schnittstelle

6.2 Einstellungen II

Im zweiten Teil der Einstellungen kann man wählen:

- welcher Dateipfad standartmäßig zum Laden bzw. Speichern verwendet wird
- ob vor dem Speichern die Info-Dialogbox angezeigt werden soll
- welche Sicherheitsabfragen CHAOS*ultd* tätigen soll

  vor dem Überschreiben von Dateien

  vor dem (unwiederuflichen) Abbrechen der Berechnung

  vor dem Löschen von Objekten und

  vor dem Verlassen des Programmes
- wieviel Speicher CHAOS*ultd* freilassen soll
- ob neue/geladene Bilder gepackt werden sollen
- für wieviele Bilder im Filmeditor Speicher reserviert werden soll (min. 256, max. 32767)

6.3 Parameter-Voreinstellungen

Mittels `Shift`-Klick auf einen der Einträge zum Erzeugen neuer Bilder eines bestimmten Fraktal-Types kann man die Voreinstellungen der Parameter für diese Fraktal-Typ aufrufen und die Parameter ändern – soweit diese Funktion unterstützt wird, was aber meist (bisher stets) der Fall ist. Ebenso lassen sich die allgemeinen Bildparameter für jeden Bildtyp einstellen.

Seltener vorhanden ist eine Funktion die bei `Control`-Klick auf einen dieser Einträge aufgerufen wird, und für die es keine fest vorgesehene Verwendung gibt, sie wird einfach für irgendwelche zusätzlichen Aktionen, die womöglich gebraucht werden, bereitgehalten.

6.4 Parameter speichern

Durch Auswahl des Menüpunktes Parameter speichern kann man die eingestellten Parameter abspeichern. Die Parameter werden ausschließlich in der Datei *CHS_ULTD.SET* im Ordner *CHS_ULTD.GEM* gespeichert.
Zu den Parameter die gespeichert werden gehören die Einstellung von CHAOS*ultd* (einschließlich der Anzeigeoptionen und der Einstellungen der Konvertierungsroutinen), die der verschiedenen Routinen (soweit geladen) und die Positionen der Dialoge und Fenster. Dialog- und Fensterpositionen werden vor dem Speichern in auflösungsunabhängige Normalkoordinaten umgerechnet, so daß diese Einstellung auch bei Auflösungswechsel erhalten bleibt (jedenfalls einigermaßen). Früher schon gespeicherte Parameter nicht geladener Routinen werden kopiert, sie bleiben also erhalten. (Möchte man solche Parameter aus der Einstellungsdatei löschen, so kann man diese während CHAOS*ultd* läuft löschen (mit einem Accessory oder einem guten Fileselektor, falls man kein Multitasking hat) und dann die Parameter speichern)

Teil II
die Fractal-Routinen

1 der Algorithmus

Fractale im Sinne dieser Routinen beruhen auf einer Verallgemeinerung von Mandelbrot- und Julia-Mengen – erstere sind auch als Apfelmännchen bekannt.
Apfelmännchen beruhen bekanntlich auf Folgen komplexer Zahlen, die durch die Rekursionsformel

$$z_{n+1} = z_n^2 - z_0$$

definiert sind[27].
Das Bild entsteht dann so, daß man einen Teil der komplexen Ebene auf den Bildschirm abbildet und für jeden Punkt z_0 untersucht, ob und wenn wann die Glieder der zugehörigen Folge dem Betrage nach einen bestimmeten Wert (die Abbruchbedingung) überschreitet. Die Farbe des Punktes bestimmt sich dann aus der der Anzahl der nötigen Iterationen bis zum Erreichen der Abbruchbedingung.
Julia-Mengen werden fast genauso berechnet, allerdings lautet die Rekursionsformel hier

$$z_{n+1} = z_n^2 + c$$

wobei c eine für alle Bildpunkte gleiche, komplexe Konstante ist.
In den Fractal-Routinen von CHAOS*ultd* sind nun diese beiden Formeln zur folgenden zusammengefaßt:

$$z_{n+1} = z_n^2 + \alpha z_0 + c$$

wobei α eine (reelle) Konstante ist.
Apfelmännchen erhält man aus dieser Formel leicht durch $\alpha = \pm 1$ und $c = 0$, Juliamengen durch $\alpha = 0$ und $c \neq 0$.
Diese Berechnungsformel liegt den Bildern, die mit den Menüpunkten Fractal(e) und 3d Fractal(e) erzeugt werden, zugrunde. Bei pseudo-dreidimensionalen Bilder wird dabei die Berechnungstiefe eines Punktes nicht nur durch seine Farbe dargestellt, sondern zusätzlich als Höheninformation verwendet.
Für freie Fract. kann die Berechnungsformel explizit angegeben werden so daß man selbst experimentieren kann.

2 Parameter

2.1 benötigte Parameter

Wie gesagt, wird ein Ausschnitt der komplexen Zahlenebene auf den Bildschirm abgebildet. Dazu muß man natürlich diesen Ausschnitt, durch den minimalen und maximalen Real- und Imaginärteil festlegen. CHAOS*ultd* erlaubt es diesen Ausschnitt noch zu drehen, wofür zusätzlich ein Winkel phi angegeben wird (in Grad). Das letztendlich gezeichnete Bild entsteht, indem der durch minimale/maximale Real- und Imagnärteile festgelegte Block mit diesem Winkel um seinen Mittelpunkt gedreht wird. Die linke obere Ecke ist somit für phi ungleich Null *nicht* mehr durch den minimalen Real-, maximalen Imaginärteil gegeben!
Des weiteren muß α, die Konstante c (Real- und Imaginärteil), die Abbruchbedingung[28], also die Zahl die der Betrag der komplexen Zahl aus der Rekursionsformel übersteigen soll, und die maximale „Tiefe“, d.h. die maximale Anzahl von Rekursionsschritten, nach der die Berechnung spätestens abgebrochen wird, festgelegt werden.
Die max. Tiefe kann bis 9999 groß werden, je größer sie ist, desto länger dauert allerdings auch die Berechnung des Bildes. (Bei einer Tiefe von 9999 kann die Berechnung eines einzigen Bildes durchaus mehrere Tage in Anspruch nehmen. Im allgemeinen verwendet man kaum Tiefen über 250).
Die Abbruchbedingung kann aufgrund des für schnelle Berechnungen (s.u.) verwendeten Festkommazahlenformates nicht größer als 11 werden, aufgrund ihrer Definition ist sie stets positiv, also größer 0. Für die anderen Parameter wird lediglich geprüft, ob der minimale Real-/ Imaginärteil auch wirklich kleiner als der maximale ist, ebenfalls aufgrund des Zahlenformates dürfen diese Parameter dem Betrage nach nicht größer als ca. 127 sein. Da die interessanten Strukturen aber ohnedies im Bereich von ± 3 real und ± 2 imaginär liegen (zumindest in etwa) sollten sich damit trotz der an dieser Stelle fehlenden Sicherheitsabfragen keine Probleme ergeben[29].

[27] statt des '-' findet man teilweise auch ein '+'
[28] viele Apfelmännchenprogramme verwenden 2 als Abbruchbedingung, allerdings finde ich die Bilder, die mit der Abbruchbedingung 10 entstehen, schöner
[29] im Berechnungsmodus exakt (s.u.) lassen sich theoretisch auch größere Bereiche realisieren, aber wie gesagt, finden kann man da an sich nichts

Weiterhin muß man natürlich festlegen, wie die Anzahl der Rekursionsschritte in Farben umgesetzt werden soll. Auch hier erlaubt CHAOS*ultd* wesentlich mehr Einstellungsmöglichkeiten als andere Programme.
Gewöhnlich ordnet man die Farben der Schrittanzahl so zu, daß sich die Farben immer wieder zyklisch wiederholen. In CHAOS*ultd* sind dagegen maximal fünf unabhängige Bereiche möglich. Für jeden Bereich, der durch die maximale Schrittanzahl, für die er noch gelten soll, festgelegt wird, kann man entweder eine feste Farbe oder aber das zyklische Wiederholen einer bestimmten Anzahl von Farben festlegen. Im letzteren Fall muß man angeben, welche Farben (kleinste und größte Nummer), welche Farbe zuerst, und wieviele verschiedene Schrittanzahlen für eine Farbe man haben will. Zu guter letzt kann man noch wählen, in welcher Farbe Punkte erscheinen sollen, für die die maximale Tiefe erreicht wurde.
Außer diesen Parametern, die den Bildinhalt beschreiben, kann man zwischen zwei Berechungsroutinen wählen: schnell und exakt, wobei schnell mit einem speziellen Festkommazahlenformat arbeitet, während exakt die Rechenroutinen von Turbo/Pure C verwendet[30].
exakt ist natürlich wesentlich langsamer, weshalb ich von der Benutzung im allgemeinen abrate. Allerdings muß man auch ganz klar sagen, daß es im Modus schnell zu mehr oder weniger großen Rundungsfehlern kommt, die umso größer sind, je stärker man den Ausschnitt vergrößert hat. Besonders stark macht sich dies auch bei kleinen Drehwinkeln bemerkbar (der Unterschied zwischen 1 Grad gedreht und 0 Grad gedreht verschwindet ab einer gewissen Vergrößerung). Die Routinen von CHAOS*ultd*, die etwa die Koordinaten am Mauszeiger berechnen, beim Ausschneiden von Blöcken etc. rechnen mit Fließkommazahlen also (ziemlich) exakt. Dadurch kann es natürlich zu Differenzen kommen, die aber letztlich harmlos sind, solange man sich für Fraktale vor allem ihrer Ästhetik wegen interessiert. Andernfalls sollte man im Zweifel zur exakten Berechnung greifen (und sich eine FPU leisten).

2.2 Eingabe der Parameter

Wie man sieht ist für jedes Bild also eine ziemlich Anzahl an Parametern einzustellen. Dazu kommt noch, daß die meisten Parameter doppelt vorhanden sind, um Bildfolgen, bei denen nur das erste und letzte Bild angegeben werden, zu ermögliche – die Parameter der übrigen Bilder werden dann durch Interpolation berechnet.
Für gewöhnliche Bilder, wie man sie mit anderen Fraktalprogrammen auch erzeugen kann, gibt es aber einige Funktionen, die das Einstellen der Parameter erleichtern.
Um Fractale zu erzeugen klickt man in der Menüleiste den Menüpunkt Fractal(s) im Menü Neu an. Es erscheint dann die (ziemlich umfangreiche, also nicht erschrecken) Dialogbox zum Eingeben der Parameter, in der (sofern die Einstellungsdatei *CHS_ULTD.SET* korrekt geladen wurden) bereits Parameter voreingestellt sein sollten (diese Voreinstellung ist jedesmal dieselbe, sie kann geändert werden, wenn man den Menüpunkt Fractal(s) mit gedrückter `Shift`-Taste aufruft; bei einem erneuten Aufruf findet man also nicht wieder die Parameter vor, die man zuletzt eingestellt hatte).

Einzelbilder

Zunächst sollen der Einfachkeit halber nur die Parameter erläutert werden, die man für die Erzeugung eines *einzelnen* Bildes benötigt, wenn man also in den allgemeinen Parametern unter Bilderzahl 1 eingetragen hat.
Oben links wird angezeigt, welchen Bildnamen, Bilderzahl und Bildgröße man eingestellt hat.
Darunter kommen zweispaltig die einzelnen numerischen Bildparameter, wie in 2.1 erläutert. Dabei interessiert für Einzelbilder nur die linke Spalte, die rechte bleibt unbenutzt; auch die Button rechts von den Parametern brauchen vorerst nicht zu interessieren.
Nun folgt noch die Einstellung des Berechnungsmodus, schnell oder exakt.
Die Einstellung der Farben wird aus Platzgründen lediglich angezeigt (im kleinsten Systemzeichensatz), eingestellt wird sie in einer eigenen Dialogbox, in die man mit dem Button Farbcod. gelangt.

In dieser Box gibt man die Angaben für bis zu fünf Farbbereiche ein, die aber nicht alle genutzt werden müssen. Zunächst kommt die Angabe, bis zu welcher Tiefe der Bereich gelten soll. Der erste Bereich beginnt mit der Tiefe 0, der zweite direkt nach dem ersten usw. Der letzte verwendete Bereich muß durch die Angabe von 9999 als Endtiefe gekennzeichnet werden, folgende Bereiche werden, auch falls vorhanden nicht mehr beachtet. Bei den Bereichen wird nicht darauf geachtet ob sie sinnvoll mit der max. Berechnungstiefe zusammenpassen, d.h. man könnte auch bei einer max. Berechnungstiefe von meinetwegen 50 einen Bereich von 75-100 definieren, obwohl ein solcher Bereich nie genutzt würde.
Als nächstes muß man festlegen, ob in dem Bereich die Farben zyklisch durchgewechselt werden sollen oder nicht. Dazu dient der Button zykl.; ist er aktiviert, so werden die Farben durchgewechselt, ansonsten nicht. Im letzteren Fall muß nur noch festgelegt werden, welche Farbe dem Bereich zugeordnet werden soll. Dies geschieht durch den Eintrag Farbe. Dabei gibt es eine Besonderheit, gibt man als Farbe -1 an, so wird die auf die zuletzt verwendete Farbe

[30] das heißt es wird mit 80 Bit Fließkommazahlen und einer entsprechenden Genauigkeit von 19 signifikanten Ziffern gearbeitet. Außerdem wird ein eventuell vorhandener Fließkommacoprozessor automatisch genutzt.

folgende Farbe hergenommen, handelt es sich um den ersten Farbbereich, so wird die Farbe 1 verwendet (auf diese Weise kann man nach zyklischen Bereichen im Farbverlauf fortfahren ohne sich erst die letzte Farbe des Bereiches überlegen zu müssen).
Bei zyklischen Bereichen legt der Eintrag Farbe die erste verwendete Farbe fest. In den Einträgen ... bis ... gibt man an, welche Farben hergenommen werden sollen. Ist der erste Eintrag kleiner als der zweite werden die Farben aufsteigend, ist der erste größer absteigend erzeugt. Für beide Einträge hat man -1 und 999 als Joker: -1 wird als 0 bei zweifarbigen Bildern und als 1 sonst ausgewertet, 999 steht für die letzte mögliche Farbe. Mit Step erklärt man, wieviele Tiefenstufen in der gleichen Farbe dargestellt werden sollen (nicht jede wievielte Farbe zwischen ... bis ... hergenommen werden soll).
Als letzter Farbparameter muß schließlich noch eingestellt werden, in welcher Farbe Punkte dargestellt werden sollen, die „tiefer“ als die max. Tiefe liegen, auch hier ist -1 als Joker verwendbar.

Ähnlich wie man mit Farbcod. in die Farbeinstellung gelangt, kann man mit Allg. Par. die Dialogbox für die Einstellung der allgemeinen Parameter aufrufen, etwa um Bildformat, Name oder Bilderzahl nochmal zu ändern.

Was jetzt noch fehlt, sind die Button -> alles, -> links und -> rechts. Mit diesen ruft man Popup-Menüs auf, die verschiedene Funktionen zur Eingabe der Parameter bieten.
Das alles-Popup bietet Funktionen, die sich auf alle Parameter (d.h. die Parametersätze für das erste und die für das letzte Bild, auch wenn ich für letztere auf den nächsten Abschnitt verweisen muß) beziehen. Hier kann man erstens alle Einträge im gesammten Dialog löschen, zweitens die Voreinstellung aufrufen (die schon bei Aufruf des Dialoges eingestellt war) und drittens, die Parameter links nach rechts kopieren.
Die links- und rechts-Popups sind gleich, sie unterscheidet nur die Tatsache, daß links auf die linke und rechts auf die rechte Parameter-Hälfte wirkt; außerdem werden die für links und rechts gemeinsamen Parameter übernommen. Letzteres kann man durch Drücken der `Shift`-Taste beim Auswählen des Objektes verhindern.
In diesem Popup stehen die folgenden Optionen zur Verfügung: man kann die Parameter löschen, man kann aus Objekten Daten holen und man kann sich das Verhältnis von Höhe zu Breite der Parameter und des Bildformates ansehen. Es existiert noch ein weiterer Eintrag im Popup-Menü, nämlich Random, der aber bei Fractalen nicht aktiv ist[31]. Beim Holen von Daten kann man entweder alle oder – mit `Shift` – nur die nach rechts und links getrennten Daten übernehmen (natürlich nur, wenn das Objekt ein Fractal ist). Möglich ist auch die Auswahl eines (beliebigen) Objektes, eines Blockes oder einer Konstante (die dann bei c eingetragen wird). Ist das Objekt kein Fractal, so wird nur der Bildausschnitt (wie bei Block) übernommen.
Soweit zu den Parametern für die Erzeugung eines Bildes. Ich kann gut verstehen, daß Ihnen das ganz schön kompliziert vorkommt. Es ist wohl auch nicht ganz einfach, aber zum einen muß man ja nicht jedes Detail ausnutzen und zum anderen muß man einfach ein bißchen herumprobieren, dann findet man sich mit der Zeit schon zurecht. Wer allerdings geglaubt hat, schlimmeres könnte ihm nicht wiederfahren, der hat sich geirrt, es kommt noch schlimmer:

Fractalbildfolgen

Wie bereits erwähnt gehört es zu den Fähigkeiten von FRACTAL, nicht nur Einzelbilder, sondern auch Bildfolgen berechnen zu können. Dabei müssen die Bildparameter für das erste und das letzte Bild sowie die Bildzahl angegeben werden, FRACTAL interpoliert dann die Werte für die dazwischen liegenden Bilder. Die Parameter werden in der selben Dialogbox eingegeben, wie die Parameter von Einzelbildern[32]. Dabei dient die linke Spalte für die Parameter des ersten, und die rechte Spalte für die Parameter des letzten Bildes. Alle Einstellungen die nur einfach vorhanden sind gelten für alle Bilder der Bildfolge, dies sind Bildgröße, Berechnungsmodus sowie die Farben. Bei den anderen Parametern kann schließlich noch die Art der Interpolation gewählt werden. Dazu dienen die Button rechts von den Parameterangaben. Gewählt werden kann lineare (0), progressive (+) und degressive (-) Interpolation. Bei linearer Interpolation sind die (absoluten) Unterschiede zwischen den Parametern zweier aufeinanderfolgender Bilder stets gleich groß, bei progressiver Interpolation nehmen sie zu und bei degressiver ab. Für den Bildausschnitt existiert noch I, hier wird so interpoliert, daß der relative Unterschied (in der Bildfläche) konstant ist (irgendwie so).

2.3 3d Fractale

Bei 3d Fractalen sind nunmehr (V6.1 der Fractal-Routinen) auch farbige Bilder möglich. Dabei werden die Punkte nicht nur unterschiedlich hoch sondern auch noch – analog zu 2d Fractalen – farbig gesetzt.
Die Farbcodierung wird in der Parametereingabe *nicht* angezeigt, weil kein Platz vorhanden ist. Sie wird wie bei 2d-Fractalen eingeben.
Bei 3d Fractale wird die Berechnungstiefe durch die Tiefe oder Höhe der gezeichneten Punkte dargestellt.

[31] die Popups existieren in gleicher Form (aber mit teilweise anderen Einschränkungen) auch bei den anderen Berechnungsroutinen
[32] Wer also die Parametereingabe für ein Bild verstanden hat, der wird auch mit Bildfolgen keine großen Probleme haben, insofern stimmt die oben gemachte Bemerkung, es käme noch schlimmer, vielleicht doch nicht so ganz.

Die Parameter entsprechen weitgehend denen von normalen Fractalen. Allerdings ist die maximale Tiefe bei 3d Fractalen entsprechend der Bildhöhe und Breite begrenzt, damit das Bild immer in die Bitmap hineinpasst.
Es kann auch nicht zwischen schneller und genauer Berechnungsroutine unterschieden werden (verwendet wird stets die schnelle Berechnugsroutine), und die Farbeinstellung fällt ganz weg.
Eingestellt werden muß statt dessen, ob das Fractal hoch oder tief gezeichnet werden soll, und mit welchem Raster das Fractal dargestellt werden soll. Hier hilft Ausprobieren der Möglichkeiten und Anschauen der Ergebnisse glaube ich besser als langatmige Erklärungen.

2.4 freie Fractale

Für freie Fractale, d.h. Bilder mit frei definierter Rekursionsformel, kommen zu den Parametern für 2d-Fractale noch einige hinzu, die in einer eigenen Dialogbox vor den anderen Parametern eingegeben werden müssen.

Berechnungsformel

Zunächst einmal muß man natürlich die Berechnungsformel angeben. Diese ist ein beliebiger Funktionsstring, der (komplexe) Konstante, Variable, Operatoren und Funktionen enthalten kann.
Konstante müssen die Form *Realteil + i Imaginärteil* besitzen und vorsichtshalber geklammert werden (normalerweise braucht man eh keine).
An Variablen kann man zn, d.h. den momentanen Wert der Rekursionsfolge, $z0$ den Startwert, d.h. die komplexen Koordinaten der Bildschirmpunktes, die reelle Zahl a und die komplexe Zahl c (wie in den weiteren Parametern eingegeben) verwenden.
An Operationen kann die üblichen binären Operationen +, -, *, / und ^ (Potenz) verwenden; außerdem Klammern (und) und Vorzeichen -.
An (komplexen) Funktionen werden unterstützt:

Re	Realteil einer komplexen Zahl
Im	Imaginärteil einer komplexen Zahl
konj	komplex konjugieren
abs	Absolut*quadrat* einer komplexen Zahl (abs$(x+iy) = x^2+y^2$) (der Name ist eigentlich falsch!)
quad	Quadrat
kubik	dritte Potenz
sin	Sinus
cos	Cosinus
exp	Exponentialfunktion
log	Logarithmus

Anmerkung: möchte man die n-te Potenz einer Zahl zn verwenden (n ganz), so sollte man nicht zn^ n benutzen, sondern die Funktionen quad, kubik, und Produkte aus zn und ihnen – z.B. zn^ 4=quad(quad(zn)) – was wesentlich schneller berechnet wird.
Beispiele für eine Berechnungsformel:
Apfelmännchen: quad(zn)-z0 Juliamengen: quad(zn)+c
Die Berechnungsformel wird zur Auswertung übrigens compiliert, d.h. in Maschinencode übersetzt, wenn auch nicht allzu guten. Tritt dabei ein Fehler auf, so wird dieser in einer Alertbox angezeigt, die Dialogbox kann nicht mit Ok verlassen werden.

weitere Parameter für freie Fractale

Neben der Berechnungsformel kann man auch die Abbruchbedingung und die Art des Punkte-Setzens beeinflussen.
Bei der Abbruchbedingung hat man die Möglichkeit eine oder zwei relevante Größen zu bestimmen, wobei der Betrag, der Real- sowie der Imaginärteil der Folgenglieder zur Wahl stehen (die Abbruchbedingung lautet immer: *Größe* > *Abbruch*, wobei Abbruch der in den Parametern wie bei normalen Fractalen angegebene Wert ist).
Wählt man zwei Größen und damit eigentlich zwei Bedingungen, so kann man noch die logische Verknüpfung (and, or oder xor) zwischen diesen wählen.
Ähnlich legt man die Bedingung zum Punkte-Setzen fest. Entweder man läßt die Punkte entsprechend den Berechnungsschritten setzen – nur dann kann man bei den übrigen Parametern auch die Farbeinstellung wie bei normalen Fractalen vornehmen – oder man definiert eine Bedingung die erfüllt sein muß, damit ein Punkt gesetzt wird. Diese Bedingung entspricht der Abbruchbedingung und wird genauso festgelegt (man kann hier natürlich andere Kombinatinen wählen), allerdings werden hier Bedingungen *Größe* < *Abbruch* verwendet[33].

[33] natürlich sind nicht alle Kombinationen von Abbruch- und Punktsetz-Bedingung auch sinnvoll, das sollte man sich vorher überlegen

Nach Verlassen der Dialogbox für frei definierte Fractale mit Ok muß man in der ganz normalen Parameterdialogbox (wie für normale Fractale) noch die übrigen Parameter angeben.

Und was soll das alles?

Wer sich dies fragt, weil er kaum noch verstehen, was denn nu bei welcher Einstellung wie berechnet und gezeichnet wird, soll hier noch kurz erklärt werden, woher frei definierte Fractale kommen.
Sie kommen von Pickover-Mengen, die im *Spektrum der Wissenschaft* Sonderheft *Computerkurzweil IV* unter dem Stichwort Biomorphe beschrieben wurden.
Dabei verwendet man Berechnungsformeln z.B. vom Typ $zn\hat{}n + c$ (n=3,4,5...) und als Abbruch- wie als Punktsetz-Bedingung Re or Im (vgl. auch die Beispiel-Bilder).

3 unterstützte Optionen

Da die Fractal-Routinen gewissermaßen das Maß aller Dinge bei der Entwicklung von CHAOS*ultd* waren[34], jedenfalls haben sie die möglichen Optionen vorgegeben, werden im Prinzip alle möglichen Optionen von den Fractal-Routinen auch in irgendeiner Weise unterstüzt.

3.1 Parameter anzeigen

Alle Parameter werden angezeigt. Bei freidefinierten Fractalen muß man zur Anzeige der Berechnungsformel und Abbruch- sowie Punktsetz-Bedingung in eine zweite Dialogbox (über den Button − >frPar) verzweigen.

3.2 Bild-Info anzeigen

Für die Fractal-Routinen steht ein umfangreiches Bild-Info für Einzelbilder zur Verfügung, die CHAOS*ultd*-Routine zur Anzeige von Infos über mehrere Bilder wird unterstützt.
In Einzelbild-Infos wird angezeigt:
die Anzahl der Bildpunkte (nicht Pixel!) – insgesammt und als Höhe mal Breite;
die Berechnungszeit in Stunden, Minuten und Sekunden, sowie die Zeit pro Zeile in Sekunden und pro Punkt in Millisekunden;
die mittlere Tiefe (Zahl der Iterationen geteilt durch Anzahl der Bildpunkte) und zum Vergleich die maximale Tiefe;
die Anzahl der Iterationen und die Zeit pro Iteration (in Mikrosekunden);
die Anzahl der nötigen Multiplikationen, die mittlere Zeit pro Multiplikation[35] und die sich daraus ergebende Zahl von Multiplikationen pro Sekunde.
Die Angaben über Multiplikationen sind bei frei definierter Berechnungsformel nicht möglich.
Für Infos über mehrere Bilder entsprechen die Angaben über Iterationen, Bildpunkte und Berechnungszeit denen der Einzelinfos.

3.3 Punktinfo

Bei normalen Fractalen kann ein Punktinfo angezeigt werden, wobei neben Koordinaten auch die Tiefe angezeigt wird; ist sie größer als die maximale Tiefe des Bildes, so wird dies durch das unendlich-Symbol (∞) markiert.

3.4 Block/Konstante zeigen

Block-Auschnitte enthalten können nur zweidimensionale Bilder. 3d Bilder können lediglich aus Ausschnitt in einem zweidimensionalen Bild angezeigt werden. Ebenso können keine Blöcke aus 3d Bildern entnommen werden.
Als Konstante werden die Parameter real con (x) und imag con (y) angezeigt (in zweidim. Bildern).

3.5 neu berechnen

Beim neu berechnen von Fractalen können Abbruchbedingung, maximale Tiefe, Genauigkeit, Darstellung bzw. Farbtabelle, die Farbkodierung von 2d-Bildern, das Raster und der Modus von 3d-Bildern sowie für frei definierte Fractale die Berechnungsformel, die Abbruchbedingung und die Punktsetz-Bedingung neu vereinbart werden.

[34] was nicht heißt, daß das so bleiben muß
[35] d.h. die Berechnungszeit geteilt durch die Zahl der Multiplikationen, tatsächlich dauert eine Multiplikation an sich weniger lang, wobei auch noch zwischen Quadratur einer Zahl und Multiplikation zweier Zahlen unterschieden werden müßte

In der Dialogbox muß dazu einerseits der gewünschte neue Wert einer Größe angegeben werden, der dann für alle zu neu berechnenden Bilder Verwendung findet, andererseits muß das Neu-setzen dieses Wertes durch selektieren des zugehörigen Button aktiviert werden.
Die Einstellungen für frei definierte Fractale finden sich in einer eigenen Dialogbox, in die man mit dem Button fr. Param. gelangt; auch die Farbtabelle wird natürlich in der üblichen Weise eingegeben.
Bilder werden grundsätzlich nur neu berechnet, wenn sich ihre Parameter geändert haben.

Teil III
die Hüpfer-Routinen

1 der Algorithmus

Hüpfer basiert auf einem Algorithmus von Barry Martin, der im Computer-Kurzweil der Zeitschrift *Spektrum der Wissenschaft* vorgestellt wurde.
Dabei betrachtet man eine Folge von Zahlenpaaren x_n und y_n mit der folgenden Iterationsvorschrift:

$$\begin{aligned} x_{n+1} &= y_n - \mathrm{sign}(x_n) \cdot \sqrt{\mathrm{abs}\,(b \cdot x_n - c)} \\ y_{n+1} &= a - x_n \end{aligned}$$

wobei a, b und c frei wählbare Parameter sind.
Die Punkte (x_n, y_n) werden in einem Koordinatensystem aufgetragen, woraus sich mehr oder weniger interessante Figuren ergeben.
Anders als bei Fractalen (siehe Teil II) erhält man hier eine Vielfalt von Formen nicht so sehr aus der Wahl des Koordinatenausschnittes sondern aus der Wahl der Parameter a, b und c. Das Ergebnis hängt außerdem – ebenfalls im Unterschied zu Fractalen – recht stark von der Anzahl der Iterationen ab: einerseits können ganze Bildteile unter Umständen erst nach sehr vielen Iterationen zum Vorschein kommen, andererseits können auch Strukturen bei zu vielen Iterationen durch die vielen gezeichneten Punkte wieder überdeckt werden.
Man beachte, daß aufgrund des Algorithmus die Rechenzeit *nicht* von der Bildgröße abhängt[36]. Berechnet man kleine Bilder, so führt das lediglich zu einer Speicherplatz- nicht aber zu einer Rechenzeitersparnis.

2 Parameter

Die Parameter sind wie üblich doppelt anzugeben – für das erste und für das letzte Bild einer Bilderfolge; berechnet man nur ein Bild, so braucht man natürlich nur die linke Spalte auszufüllen. Auch die Button für den Interpolationsmodus sind ja bekannt.
Zunächst einmal muß man die drei Parameter a, b und c der Iterationsvorschrift angeben; es folgen die Startwerte für x und y sowie die Festlegung des Bildausschnittes. Aus traditionellen Gründen geschieht dies im allgemeinen nicht durch minimalen und maximalen Wert für x und y sondern durch einen Vergrößerungsfaktor (scale) und einen Offset, der die Position der Bildmitte angibt (jeweils für x und y getrennt). Man kann in der Dialogbox auch durch Anklicken des Button min/max auf die Eingabe der Minimal- und Maximal-Werte umschalten, dabei ist aber zu beachten, daß für die Interpolation bei Bildfolgen stets mit den Vergrößerungsfaktoren und Offset's gerechnet wird (d.h. bei Verlassen der Box werden die Werte in diesen Modus umgerechnet).
Schließlich ist noch die Iterationsanzahl anzugeben und die Schrittweite, d.h. die Zahl der Iterationen nach der jeweil ein Punkt gesetzt werden soll (normalerweise setzt man nach jedem Schritt einen Punkt, d.h. Schrittweite 1).
Die Eingabe der Parameter wird wie bei Fractalen durch eine Reihe von Möglichkeiten unterstützt. Zusätzlich zu den Möglichkeiten die man bei Fractalen hat, kann man mit Random die zufällige Vorbelegung der Parameter a, b und c erreichen. Den Bereich für diese Zufallszahlen kann man durch Aufruf des Hüpfer-Menüeintrages zusammen mit der CONTROL-Taste einstellen, wobei man links den minimalen und rechts den maximalen Wert angibt. Die Möglichkeit das Verhältnis von Bildbreite zu Höhe anzuzeigen fehlt.
Ebenfalls Zufallszahlen für die Bildparameter a, b und c erreicht man durch den zusätzlichen (zu den von den Fractal-Routinen bekannten) Interpolationstyp r wie random. Wählt man r als Interpolationstyp, so wird nicht interpoliert sondern ein Zufallswert zwischen den zwei angegebenen Werten ermittelt (in diesem Fall ist auch bei der Berechnung eines einzigen Bildes die rechte Parameterhälfte (wenigstens teilweise) relevant).
Die Vorbelegung der Parameter kann man wie üblich durch SHIFT-Klick auf den Hüpfer-Menüeintrag erreichen.

3 unterstützte Routinen

Die meisten Möglichkeiten von CHAOS*ultd* werden unterstützt.
Berechnungsabbruch ist alle 256 Berechnungsschritte möglich – wie üblich durch Drücken der ESC-Taste. Man kann dabei durch gleichzeitiges Drücken von CONTROL erreichen, daß das Bild – soweit bisher berechnet – für fertig erklärt

[36] die Rechenzeit hängt hier lediglich von der Zahl der Iterationen ab, weshalb sie auch vorher abgeschätzt werden kann

wird, die Angabe für die Iterationsanzahl wird dann entsprechend herabgesetzt (ohne Auswirkung auf weitere Bilder etwa in einer Bilderfolge).
Beim Parameter anzeigen kann man umgekehrt die Iterationsanzahl wieder erhöhen, die entsprechende Angabe in der Dialogbox ist edierbar. Man kann also die gewünschte neu Iterationsanzahl eingeben, die aber größer sein muß, als die Anzahl der bisher berechneten Iterationen (andernfalls wird die Angabe schlicht ignoriert); ein bereits fertiges Bild kann auf diese Weise wieder unfertig werden.
Weiter kann man Bilderinfos anzeigen lassen, wobei es aber kein ausführlicheres Info für Einzelbilder gibt. Blockroutinen werden ebenso unterstützt. Das Neu-Berechnen von Hüpferbildern ist möglich, wobei alle Parameter (einschließlich Bildausschnitt) verändert werden können.

Teil IV
die L-System-Routinen

Diese Routinen wurden (für FractInt) von **Adrian Mariano** geschrieben und von **Nicholas Wilt** durch (80xxx) Assembler-Routinen beschleunigt.
Zunächst eine Erklärung, was L-Systeme überhaupt sind (die man so nicht verstehen muß). Ich zitiere die Anleitung zu FractInt (Ausschnitt aus dem Anhang A, SUMMARY OF FRACTAL TYPES):

> lsystem
> Using a turtle-graphics control language and starting with an initial axiom string, carries out string substitutions the specified number of times (the order), and plots the resulting.

Sagt doch alles, oder? (Wenn man's mal verstanden hat schon)

1 Definition der Parameter

Bei L-Systemen werden – gewissen Regeln folgend – Linienstücke gezeichnet. Die Regeln kann man sehr flexibel definieren, die möglichen Ergebnisse reichen von Koch-Kurven und ähnlichem bis zu Penrose-Tilings (das sind nicht-periodische Parkettierungen der Ebene aus nur zwei verschiedenen Grundelementen).
Die Regeln werden durch Zeichenketten beschrieben, wobei man einen Anfangsstring, das Axiom, und einen Satz von Ersetzungsregeln hat. Die Zeichen des Anfangsstrings werden gemäß den Ersetzungsregeln ersetzt, so daß man eine neue Zeichenkette erhält. Dieser Vorgang wird mehrfach wiederholt. Am Schluß erhält man eine End-Zeichenkette, deren Zeichen als Zeichenbefehle interpretiert werden und anhand derer das Bild gezeichnet wird[37].

Um das Ganze etwas klarer zu machen vielleicht ein einfaches Beispiel:
Als Anfangsstring könnte man
'F'
verwenden, als (einzige) Ersetzungsregel (Zeichen für die keine Regel definiert ist, bleiben erhalten):
'F=F+F--F+F'
Damit erhält man nach der ersten Ersetzung:
'F+F--F+F',
nach der zweiten
'F+F--F+F+F+F--F+F--F+F--F+F+F+F--F+F'
und nach der dritten:
'F+F--F+F+F+F--F+F--F+F--F+F+F+F--F+F+F+F--F+F+F+F--F+F--F+F--F+F+F+F--F+
F--F+F--F+F+F+F--F+F--F+F--F+F+F+F--F+F+F+F--F+F+F+F--F+F--F+F--F+F+F+F--F+F'
Jetzt zeichne man für jedes 'F' ein (jeweils gleichlanges) Linienstück und drehe die Zeichenrichtung bei jedem '+' um 60° nach links, bei jedem '–' um 60° nach rechts.
Was man dann nach einiger Zeit erhält ist die Kochkurve. (Der Computer kann das natürlich ein bißchen schneller.)

Die vollständige Definition von L-Systemen ist natürlich umfangreicher.
Zunächst muß ein Drehwinkel festgelegt werden. Dies geschieht durch eine (ganze) Zahl n, womit der Winkel auf 360/n Grad eingestellt wird. (Im Beispiel war n also 6.) n muß zwischen 2 und 50 liegen.
Dann muß man das Axiom und die Ersetzungsregeln angeben.
Beim Zeichnen werden dann die folgenen Kommandos erkannt:
Standart-Kommandos:

- F zeichne vorwärts
- G gehe vorwärts ohne zu zeichnen
- \+ erhöhe den Winkel (um 360/n Grad), d.h. drehe nach links
- \- erniedrige den Winkel (um 360/n Grad), d.h. drehe nach rechts
- | versuche eine 180°-Wende. Ist n ungerade, wird um den größtmöglichen Winkel kleiner 180° gedreht.

Alternative Zeichen-Kommandos:
Diese Kommandos kommen ohne die Winkeleinstellung auf 360/n Grad aus, sind damit aber auch langsamer. Man sollte sie also nur verwenden, wenn die 360/n-Einstellung für den Winkel nicht flexibel genug ist (die Winkelangabe nn im folgenden muß nicht ganzzahlig sein).

[37] in der Realisierung werden diese Zeichenketten nicht wirklich gebildet sondern das Bild durch einen rekursiven Algorithmus gezeichnet

D zeichne vorwärts

M gehe vorwärts ohne zu zeichnen

nn erhöhe den Winkel um nn Grad, d.h. drehe nach links

/nn erniedrige den Winkel um nn Grad), d.h. drehe nach rechts

Farbeinstellung:

Cnn wähle Farbe nn

<nn erhöhe Farbnummer um nn

>nn erniedrige Farbnummer um nn

weitere Kommandos (für Fortgeschrittene):

! drehe die Richtungen um, d.h. vertausche die Bedeutung von '+','-' und '/'

@nnn multipliziere die Länge der Zeile mit nnn
nnn kann eine einfach (Fließkomma-)Zahl sein, oder ein vorangestelltes I für invers und/oder Q für Quadratwurzel besitzen (@IQ2 teilt die Länge durch Wurzel 2)

[Push: speichere momentanen Winkel und Position auf einen Stack

] Pop: kehre zur Positon (und Winkel) des letzten push zurück

Alle anderen Zeichen dürfen ebenfalls in Kommando-Strings vorkommen. Sie werden beim Zeichnen ignoriert, können aber verwendet werden um kompliziertere Ersetzungen zu ermöglichen.
Außer der Definition des L-Systems hat man nur noch die Rekursionstiefe, also die Zahl der Ersetzungen als Parameter. Diese sollte man nicht zu hoch wählen, wobei diese natürlich noch von der Definition des L-Systems abhängt. In den allermeisten Fällen sind aber schon Tiefen von 4 oder 5 ausreichend. Größere Tiefen dauern nicht nur länger sondern können unter Umständen gar nicht gezeichnet werden.

2 Eingabe der Parameter

Zur Eingabe der Parameter hat man eine – im Gegensatz zu anderen Parameterdialogen – recht einfache Dialogbox. Direkt eingeben kann man nur die Rekursionstiefe, dies ist auch der einzige Parameter, der für Bildfolgen interpoliert werden kann.
Darunter wird der Name des eingestellten L-Systems und die Definition ausgegeben. Die Definition beginnt mit dem Winkel (angle) und dem Axiom. Es folgen maximal acht Transformationsregeln. Tatsächlich kann die Definition mehr Regeln umfassen, die dann aber nicht angezeigt werden. Es kann auch sein, daß das Axiom oder Regeln nicht vollständig angezeigt sondern abgeschnitten werden, einfach weil der Platz nicht reicht.
Um die Definition zu ändern hat man die vier Button in der vorletzten Zeile:
Mit Löschen löscht man den Eintrag, mit Holen kann man die Definition aus einem Objekt übernehmen, mit Laden die Definition aus einer Parameter-Datei laden und mit Edieren eine Editor-Dialogbox aufrufen.
Die Definition des L-Systems kann in dieser Editor-Dialogbox eingegeben werden. Der Editor unterliegt bezüglich der Größe der Definiton (Zahl der Regeln und Länge der Zeilen) den gleichen Einschränkungen wie die Anzeige der Definition. Er kann nur aufgerufen werden, wenn die momentan eingestellte Definition nicht zu groß ist. Andernfalls ist der Ediere-Button gesperrt.
Im Editor findet man je eine Zeile für den Namen der Definition, den Winkel, das Axiom und bis zu 8 Regeln.
Darunter befinden sich Button zum Speichern, Laden und Löschen der Definition sowie ein Abbruch und ein Ok Button.
Geladen (im Parameter- wie im Edit-Dialog) werden die Definitionen aus L-System-Parameterdateien (*.L). Die mitgelieferten Dateien *FRACTINT.L*, *TILING.L* und *PENROSE.L* sind der FractInt-Distribution entnommen. Autoren: Adrian Mariano, Joe Saverino, Herb Savage, Morgan Savage, Jim Hanan, Ken Philip, Philippe Hurbain (ohne Anspruch auf Vollständigkeit, siehe Kommentare in den Dateien). Das Format der Definitionsdatei dürfte klar sein, wenn man sich die Beispieldateien anschaut.
Zum Auswahldialog siehe den entsprechenden Abschnitt der Anleitung.
Wird vom Editor-Dialog aus geladen und ist die Definition zu groß, so wird eine Fehlermeldung ausgegeben und der Editor-Dialog gelöscht. Ansonsten sind geladene Definitionen nicht an die Einschränkungen der Anzeige und des Editor-Dialoges gebunden, sie können aber natürlich auch nicht beliebig groß werden. Hier gilt, daß Zeilen nicht länger als 160 Zeichen sein dürfen (das wird in CHAOS*ultd* glaube ich nicht vorrausgesetzt, ist aber für die Kompatibilität zu

FractInt sinnvoll, Ersetzungsregeln können auch mehrzeilig sein, indem man das gleiche Zeichen mehrfach definiert), und nicht mehr als 25 Zeilen möglich sind (andernfalls erhält man bei Start der Berechnung eine Fehlermeldung).
Zum Speichern der Definition im Editor ist zu sagen, daß natürlich eine Datei geschrieben wird, die wieder geladen werden kann. Existiert die Datei, die man im Fileselector angibt, bereits, so wird die Definition angehängt, ohne daß geprüft wird, ob eine gleichnamige Definition schon in dieser Datei steht (will man also eine korrigierte Definition wieder in die gleiche Datei schreiben, so muß man die alte Version hinterher von Hand (Texteditor!) löschen).
Ebenfalls beschränkt die maximale Größe der Definition in der Voreinstellung, wo – unabhängig von allen bisherigen Einschränkungen – gilt, daß die Defintion nicht länger als 512 Byte sein darf. Ist die Definition (beim Setzen der Voreinstellung) länger, so wird sie – ohne Warnung – einfach nicht in die Voreinstellung übernommen und die alte Voreinstellung bleibt erhalten.

3 Berechnung von L-Systemen

Bei der Berechnung von L-Systemen wird zunächst die Größe des Resultats ermittelt, d.h. es passiert erst einmal nichts, was man auf dem Bildschirm mitverfolgen könnte. Danach wird dann das L-System gezeichnet.
Die Berechnung L-Systeme kann zwar abgebrochen werden, Fortsetzen der Berechnung ist aber unmöglich (wegen des rekursiven Algorithmus), so daß das Bild für fertig erklärt wird. Beim Versuch abzubrechen wird man in einer Alert-Box gewarnt.
Bei der Berechnung wird *externer* Speicher (vom System) benötigt (die Menge hält sich in Grenzen; 8kByte reichen glaube ich meistens).
Es kann bei der Berechnung zu Fehlern kommen, erstens wegen Speichermangels oder drohendem Stacküberlauf, zweitens wegen Unmöglichkeit das Bild zu skalieren. In beiden Fällen wird eine Alert-Box ausgeben, die Berechnung abgebrochen. Das Bild bleibt unfertig (nochmaliges Berechnen führt aber (außer eventuell bei Speichermangel) zum gleichen Fehler); solche Bilder sollte man löschen.

Anmerkungen

Ach ja, noch eine Anmerkung zu L-Systemen: diese Bilder besitzen *kein* Koordinaten-System.
Letzte Anmerkung zu L-Systemen: einige der Farb-L-Systeme aus den Parameterdateien (...Color) produzieren bei 2 Farben nur leere Bitmaps (weiß auf weiß sieht man so schlecht), d.h. man muß Raster oder mehr Farben verwenden.

Teil V
die Feigenbaum-Routinen

Feigenbaum-Diagramme können nur in 2 Farben erzeugt werden.

Algorithmus

Der Algorithmus von Feigenbaum-Diagrammen ist noch einfacher als der für Hüpfer.
Dabei wird eine Folge von Zahlen nach der Rekursionsformel

$$x_{n+1} = r \cdot x_n \cdot (1 - x_n)$$

gebildet (mit $x_0 \neq 0$, z.B. $x_0 = 0.5$).
Aufgetragen werden die Werte, die x_n annimmt, gegen den Parameter r, wobei man allerdings die ersten $N1$ Folgenglieder wegläßt.
Als Ergebnis erhält man das Feigenbaumdiagramm. Warum das so aussieht wie es aussieht kann man z.B. im *Spektrum der Wissenschaft* Sonderheft *Computerkurzweil III* nachlesen.

Parameter

Nötige Parameter zum Plotten von Feigenbaumdiagrammen sind minimaler und maximaler Wert der Parameters r (üblicherweise zwischen 0 und 4) sowie der Folgenglieder x (natürlich nur für die Ausgabe, die Werte liegen zwischen 0 und 1), der Startwert x_0 (ebenfalls zwischen 0 und 1), der Zeichenmodus (hoch oder quer), die Zahl der Iterationen vor dem Plotten der Punkte (z.B. 200) und die Zahl der Iterationen beim Plotten der Punkte (z.B. ebenfalls 200), wobei hier noch eine Schrittweite angegeben werden kann, d.h. die Zahl an Iterationen zwischen zwei Plots (normalerweise 1, d.h. man läßt jeden Folgenwert zeichnen).
Es gibt die üblichen Interpolationstypen falls man mehrere Bilder berechnet. Ebenso stehen die üblichen Hilfsfunktionen zur Parametereingabe zur Verfügung.

unterstützte Routinen

Die Feigenbaumroutinen unterstützen nicht alle Möglichkeiten von CHAOS*ultd*. Unterstützt werden die Anzeige der Parameter, die Anzeige von Bilderinfos (aber ohne eigenen Dialog für Einzelbilder) und Blockroutinen.
Nicht unterstützt wird die Funktion neu berechnen. Feigenbaum-Diagramme sind einfach nicht abwechslungsreich genug um diese Routinen auch noch zu implementieren, die Routinen sollten ursprünglich ja vor allem als Beispiel für externe Routinen und deren Realisierung dienen.

Teil VI
die XCHAOS-Routinen

Die XCHAOS-Routinen erlauben es, neue Berechnungsroutinen zu realisieren, ohne daß man sich jedesmal mit den Dialogboxen zur Parametereingabe herumschlagen kann. Man kann einfach definieren, welche Parameter gebraucht werden, und die XCHAOS-Routinen passen mit diesen Daten dann einen für alle XCHAOS-Routinen gemeinsamen Eingabedialog an und übernehmen die Eingabe der Parameter. Konkret geschrieben werden muß für den neuen Fraktal-Typ dann nur noch die eigentliche Berechnungsroutine, was natürlich sehr viel weniger Arbeit darstellt.
Natürlich unterliegt man dabei ettlichen Einschränkungen, wieviele Parameter möglich sind usw., dennoch stellen die XCHAOS-Routinen für nicht zu komplizierte Fraktale eine elegante Lösung dar.
Die Unterroutinen zum Modul XCHAOS werden allerdings nicht mehr extern nachgeladen, sondern sind integraler Bestandteil des Moduls, man kann hier also nicht mehr auswählen, welche der Routinen man laden will, entweder man lädt alle oder – durch Entfernen des Moduls XCHAOS – garkeine.
Bei der Parametereingabe hat man die üblichen Hilfen, zusätzlich existiert ein Popup-Menü Voreinst., das verschiedene Voreinstellungen für die linke Parameterhälfte vorsieht. (Diese Möglichkeit wurde geschaffen, weil ich stets mehrere Bildtypen zusammengefaßt habe, und die Grundbilder teilweise verschiedene Parameter benötigen, die so miteingestellt werden können. Wenn man also die Routinen ausprobieren will, schaltet man am besten mit diesem Popup-Menü zwischen den Bildtypen um, nicht direkt mit den dafür vorgesehenen Button.)
Noch eine Anmerkung zu den Button:
aus technischen Gründen werden alle Button stets wie Radio-Button dargestellt (also mit Kreis), das heißt aber nicht automatisch, daß sie auch welche sind.
Von den XCHAOS-Routinen werden alle wesentlichen Funktionen von CHAOS*ultd* unterstützt; (noch) nicht unterstüzt wird Ausdrucken von Parametern.

1 Barnsley-Fractale

Die Formeln für Barnsley-Fractale wurden aus dem PC-Programm Fractint entnommen, sie wurden von M. Barnsley entdeckt (oder erfunden, wie immer man will). CHAOS*ultd* kennt vier verschiedene Barnsley-Fractale, eins mehr als Fractint, weil die Beschreibung und die Realisierung im Programm eines der Fractale in Fractint irgendwie nicht übereinstimmt (so ganz verstehe ich gerade auch nicht, was da eigenlich los ist, was ich berechne und was das mit der Berschreibung von Fractint zu tun hat; einerlei, die Routinen sind jetzt wie sie sind. CHAOS*ultd* rechnet jedenfalls so wie unten angegeben).
Von jeder dieser vier Varianten gibt es zwei Versionen: Mandelbrot- und Juliamengen. Generell entspricht die Berechnung der Bilder derjenigen von üblichen Mandelbrot- und Juliamengen. Damit ist auch der Unterschied zwischen den beiden Versionen klar: bei Julia-Versionen gibt es eine globale, d.h. für alle Punkte des Bildes gleiche Konstante c, die man in den Parametern angibt, bei den Mandelbrot-Versionen wird diese Konstante für jeden Punkt mit dessen Koordinaten initialisiert. Dafür werden in der Mandelbrot-Version als Startwert für die Iteration nicht die Koordinaten sondern die Parameter verwendet. Iteriert wird in den einzelnen Bildtypen (1-3 und I) wie folgt (z_n und c sind *komplexe* Zahlen, mit $\Re c$ ist der Realteil, mit $\Im c$ der Imaginärteil von c gemeint):

- Typ 1 (entspricht Beschreibung, falls mit 'modulus' das Absolutquadrat gemeint ist):

$$\begin{aligned} z_{n+1} &= (z_n - 1) \cdot c \quad \text{falls } \Re z_n \geq 0 \\ z_{n+1} &= (z_n + 1) \cdot \bar{c} = (z_n + 1) \cdot |c|^2 / c \quad \text{sonst} \end{aligned}$$

- Typ I (entspricht Algorithmus in Fractint):

$$\begin{aligned} z_{n+1} &= (z_n - 1) \cdot c \quad \text{falls } \Re z_n \geq 0 \\ z_{n+1} &= (z_n + 1) \cdot c \quad \text{sonst} \end{aligned}$$

- Typ 2:

$$\begin{aligned} z_{n+1} &= (z_n - 1) \cdot c \quad \text{falls } \Re z_n \cdot \Im c + \Re c \cdot \Im z_n \geq 0 \\ z_{n+1} &= (z_n + 1) \cdot c \quad \text{sonst} \end{aligned}$$

- Typ 3:

$$\begin{aligned} z_{n+1} &= (\Re z_n^2 - \Im z_n^2 - 1) + \imath \cdot (2 \Re z_n \cdot \Im z_n) \quad \text{falls } \Re z_n > 0 \\ z_{n+1} &= (\Re z_n^2 - \Im z_n^2 - 1) + \Re c \cdot \Re z_n + \imath \cdot (2 \Re z_n \cdot \Im z_n + \Im c \cdot \Re z_n) \quad \text{sonst} \end{aligned}$$

In den Parametern gibt man wie bei Fractalen den Bildausschnitt an, also den minimalen und maximalen Wert des Real- und Imaginärteils, die hier aber mit minimalem und maximalem x- bzw. y-Wert bezeichnet werden. Es folgen Real- und Imaginärteil der Konstante c (bzw. des Startwertes z_0), die Abbruchbedingung und die maximale Tiefe (letztere wiederum anlog zu den Parametern bei Fractalen).
Schließlich wählt man noch, welchen Bildtyp (1, I, 2 oder 3) und welche Version (`Juliamenge` oder `Mandelbrotmenge`) man wünscht.
Außerdem gibt es schnelle und exakte Berechungsroutinen, ebenfalls wie bei Fractalen; allerdings ist die Bilderzeugung von Barnsley-Fractalen generell langsamer als bei Fractalen.

2 seltsame Attraktoren

Unter (seltsame) Attraktoren sind mehrere Kurven (d.h. Lösungen von Differentialgleichungen) bzw. Punktmengen (analog zum Hüpferalgorithmus) zusammengefasst. Auch diese Bildtypen wurden dem Programm FractInt – bzw. seiner Beschreibung – entnommen.
Attraktoren heißen die Dinger, weil sich die Kurven bzw. Punkte irgendwann nur noch in bestimmten, eigenartig geformten (deshalb ‚seltsam'), Raum- bzw. Flächenbereichen finden. Auf die Details soll hier (so unbefriedigend das sein mag, aber zu allgemeinen Theorie-Erklärungen habe ich jetzt einfach keine Lust) nicht weiter eingegangen werden. Wer nicht weiß, das eine Differentialgleichung (DGL) ist, der muß sich eben ohne Verständnis an der Kurve freuen oder es sein lassen.
Im einzelnen hat man:

- Lorenz-Attraktor
  dem Lorenz-Attraktor liegt das DGL-System
  $$\begin{aligned} \dot{x} &= -ax + ay \\ \dot{y} &= bx - y - zx \\ \dot{z} &= -cz + xy \end{aligned}$$
  zugrunde, wobei a, b und c anzugebende Parameter sind
- Rössler-Attraktor
  ebenfalls ein DGL-System:
  $$\begin{aligned} \dot{x} &= -y + z \\ \dot{y} &= x + ay \\ \dot{z} &= b + xz - cz \end{aligned}$$
- Pickover-Atttrakor
  keine DGL, sondern eine diskrete Punktfolge mit
  $$\begin{aligned} x &= \sin(ay) - z\cos(bx) \\ y &= z\sin(cx) - \cos(dy) \\ z &= \sin(x) \end{aligned}$$
  (scheint nicht so recht zu funktionieren, aber ich weiß auch nicht, wie die Bilder überhaupt ausschauen sollen)
- Henon-Attraktoren
  auch eine Punktmenge, diesmal nur noch zweidimensional
  $$\begin{aligned} x &= 1 + y - ax^2 \\ y &= bx \end{aligned}$$
- der Gingerbreadman
  $$\begin{aligned} x &= 1 - y + |x| \\ y &= x \end{aligned}$$
  (die Parameter a und b werden als Startwert für x und y verwendet)

Alle diese Routinen wurden zusammengefaßt, wodurch Parameter teilweise überzählig sind.
Bei den 3d-Routinen (Lorenz, Rössler und Pickover) muß man sich für ein Darstellungsebene entscheiden, d.h. es gibt *keine* 3d-Darstellung, sondern es werden nur zwei der drei Koordinaten dargestellt. Bei den 2d-Routinen ist die Darstellungsebene natürlich immer die xy-Ebene. Bei der Bezeichnung minimale/maximale x/y-Werte (oben in den Parametern) bezieht sich das x/y auf den Bildschirm, nicht auf die Koordinaten, d.h. wählt man die zx-Ebene, so ist minimales x der kleinste z-Wert, der ganz links am Bildschirm angezeigt wird (usw.).
Die Differentialgleichungs-Systeme werden auf die allerprimitivste Weise, also durch Diskretisierung (kein Runge-Kutta oder sonst was besseres) gelöst, wobei der Zeitschritt eingestellt werden kann (dt). Je kleiner dieser ist, des genauer die Lösung, desto länger aber auch die Berechung. Neben der Größe des Zeitschrittes gibt man noch die Zahl der Zeitschritte (bzw. Iterationen) an.
Die Startwerte von x, y und z für die Berechnung sind (außer beim Gingerbreadman) fest vorgegeben: $x = y = z = 1.0$.

3 Diffusion

Die Diffusions[38]-Routinen wurden dem PC-Programm FractInt entnommen. Sie wurden (für FractInt) von Adrian Mariano geschrieben und von Juan J. Buhler um zusätzliche Optionen ergänzt.
Der Algorithmus wurde unter dem Titel 'Fraktales Wachstum' in Computer-Kurzweil III (Spektrum der Wissenschaft Sonderheft 8) vorgestellt. Dort kann man auch nachlesen, wie man das Ganze interpretieren kann.
Bei diesem Bildtyp wird zunächst ein Start'bild' erzeugt, das aus einem Punkt in der Mitte des Bildes, aus einer Linie am unteren Ende oder einem das Bild einschließenden Rechteck bestehen kann.
In dieses Bild werden jetzt zufällig Punkte gesetzt und diese dann solange jeweils um eine Position nach links, rechts, oben oder unten bewegt (random walk), bis sie auf einen bereits vorhandenen Punkt stoßen. Das Bild ist fertig erzeugt, wenn die entstandene Struktur den Bildschirm füllt (was *sehr* lange dauern kann, wenn das Bild etwa 320x200 Punkte oder größer ist).
Mehr geschieht eigentlich nicht. Allerdings gibt es noch ein paar technische Details. Würde der Punkt wirklich total zufällig in das Bild gesetzt, dann würde es ewig dauern, bis das Bild fertig ist. Deshalb gibt es einen Mechanismus, der den Punkt in der Nähe der schon vorhandenen Struktur losläßt, wobei in den Parametern spezifizieren kann, was man unter 'Nähe' verstehen will (wieviele Pixel der etwa der Kreis um den Punkt groß sein soll; der Kreis wächst natürlich mit der Struktur mit).
An Parametern hat man keine Koordinaten (wozu auch), die Wahl des Anfangszustands, die Möglichkeit zwischen 8 und 4 Nachbarn zu wählen (dies ist eine Erweiterung der FractInt-Routine meinerseits, bei 8 Nachbarn zählen die Diagonalen mit, bei 4 nicht), die Möglichkeit den Punkt auf seinem Irrweg anzuzeigen oder nicht (letzteres geht natürlich viel schneller) und den Abstand beim Loslassen der Punkte.

4 Popcorn

Popcorn basiert auf einem von Clifford A. Pickover vorgeschlagenen Algorithmus, den ich dem Sonderheft Computer-Kurzweil IV der Zeitschrift Spektrum der Wissenschaft entnommen habe.
Zugrunde liegt die Iterationsformel:

$$\begin{aligned} x_{n+1} &= x_n - h\sin(y_n + \tan(3y_n)) \\ y_{n+1} &= y_n - h\sin(x_n + \tan(3x_n)) \end{aligned}$$

Anders als bei Hüpfer wird hier aber nicht mit einem Wertepaar x_0, y_0 gestartet, sondern mit ettlichen solchen Wertepaaren. Dabei dienen Punkte des gezeichneten Koordinatensystems als Anfangswerte, wobei man wählen kann, wie dicht diese Punkte im Bild liegen sollen.
Als Parameter hat man den Koordinatenausschnitt, die Größe von h, die klein gewählt werden sollte und den Abstand zwischen Anfangspunkten in Raster-Punkten.
Für die Berechnung kann man zwischen exakt und schnell wählen, wobei zu beachten ist, daß die bei schnell verwendeten Fixkommazahlen nur mit 16 Nachkommabits also ca. 5 Stellen nach dem Komma rechnen. Diese Ungenauigkeit planzt sich natürlich durch den iterativen Algorithmus weiter fort, so daß sich die Ergebnisse bei schneller und exakter Berechnung doch deutlich unterscheiden. Andererseits ist die exakte Berechnung auf Rechnern ohne FPU wegen der Trigonometrie ziemlich langsam. Und letztenendes kommt es im Rahmen eines Programmes wie CHAOS*ultd* ja weniger auf mathematische Exaktheit an, als auf die Ästhetik der entstandenen Bilder – mir jedenfalls.

5 Kreis^2

Dieser Bildtyp – es handelt sich hier übrigens (wie auch beim Popcorn) *nicht* um ein Fractal – beruht auf einer sehr einfachen Formel:

$$z = \int (x^2 + y^2)$$

wobei x und y die Punktkoordinaten sind. Gezeichnet wird dann einfach mit der Farbe $z\%max$ wobei max die maximal mögliche Farbe und und % der Modulo-Operator ist (der also den Teilungsrest von z/max liefert).
That's all.
Die Routine wurde von John E. Connett vorgeschlagen und in Computer-Kurzweil I (Spektrum der Wissenschaft Sonderheft) vorgestellt.
Einzugeben sind natürlich maximale und minimale Koordinaten und die maximale Farbe, die natürlich nicht größer als die Zahl der Farben im Zielbild sein darf. Auch hier kann man wieder zwischen exakter und schneller Berechnung wählen.

[38] genauer: diffusionsbegrenzte Aggregation

Teil VII
Hardcopy-Routinen

Momentan gibt es zwei Druckertreiber, einen für 24 Nadel-Matrixdrucker und einen (einfacheren) Gdos-Treiber. Von den Druckertreibern (**.HCP* im Ordner *CHS_ULTD.GEM* wird stets nur der (physikalisch) erste geladen. Man sollte deshalb den anderen (oder beide, wenn man keinen Treiber braucht) entweder durch umbenennen deaktivieren oder gleich löschen (natürlich nur in der eigenen Installation).

1 24 Nadel-Matrixdrucker-Treiber

Druckerparameter

Dies ist natürlich der uninteressantere Teil der Hardcopy-Routine. Er wird mit dem Menüpunkt Druckparameter aufgerufen (obwohl es natürlich eigentlich Druck*er*parameter sind).

Einzugeben sind die folgenden Druckercodes:

- die Breite einer Graphikzeile in Punktspalten (bei 180 DPI)
  (bei A4 Druckern ist dies 1440; CHAOS*ultd* orientiert sich an dieser Angabe erstens um nicht zu viele Graphikdaten zu drucken und zweitens um Spalten zu setzen)
- Graphik 180: 180dpi Graphikdruck. Die Anzahl der Spalten wird vom Programm selber angefügt, dies soll durch das xx am Ende der Zeile symbolisiert werden.
- CR: damit wird nach Ausgabe einer Graphikzeile ein Zeilenvorschub (und Wagenrücklauf) erzielt.
- Abst. Graphik: Zeilenvorschub auf xx 1/180 Zoll einstellen. Der Parameter xx am Ende wird wieder vom Programm angefügt.
- Abst. Text: Zeilenvorschub für Text
- Druckkopf positionieren: dieses Kommando setzt den Druckkopf an die Position xx (kennen wir schon, das xx wird vom Programm am Ende des Kommandos angefügt) und zwar in 1/60 Zoll. (was anderes kann der P6+ irgendwie nicht, wie das mit anderen Druckern ist weiß ich nicht)
- Linefeed, Formfeed, Init Drucker
  diese drei letzten Kommandos können im Drucken-Dialog unmittelbar an den Drucker geschickt werden, Formfeed wird auch zum Auslösen eines Seitenvorschubes nach x Bildern (su.) verwendet, Init Drucker wird am Anfang der Hardcopy ausgegeben, um den Drucker zurückzusetzen.

Voreingestellt sind NEC P6+ Codes (sofern die Voreinstellungen nicht verloren gingen), die bei den meisten Druckern passen dürften.

Bilder drucken

Dies ist natürlich der interessante Teil der Routine (der aber nur funktioniert, wenn man sich erfolgreich um den uninteressanten gekümmert hat).
Die Hardcopy-Routine kann nur aufgerufen werden (Menüpunkt Bilder drucken), wenn der Drucker bereit ist. Andernfalls erscheint eine Fehlermeldung.
Die Hardcopy-Routine ist recht flexibel gehalten und besitzt deshalb etliche Parameter.
Zunächst kann man die Zahl der Spalten, die nebeneinander gedruckt werden sollen einstellen. Das Blatt wird dann in entsprechend viele Spalten aufgeteilt – unabhängig von der Breite der zu druckenden Bilder und unabhängig von ihrer Zahl (d.h. hat man ein sehr breites und ein sehr schmales Bild und stellt zwei Spalten ein, so bekommen beide Bilder je die Hälfte des Platzes; stellt man 8 Spalten ein, druckt aber nur ein Bild, so steht diesem einen Bild nur die erste der 8 Spalten, also ein achtel des Platzes zur Verfügung!). Möglich sind 1-8 Spalten, wobei bei 8 Spalten diese doch sehr schmal werden.
Dann kann man die Bildgröße einstellen, indem man angibt ob jedes Pixel des Bildes einfach, doppelt, dreifach oder vierfach gedruckt werden soll (in der Breite *und* in der Höhe, d.h. bei doppelt werden für ein Bild-Pixel 4 Drucker-Pixel gedruckt).
Als nächstes lassen sich Bilder normal oder quer, d.h. um 90 Grad gedreht ausdrucken.
Dann stellt man ein, ob leere Zeilen ausgegeben werden sollen oder nicht - wenn man sie nicht braucht, kann man durch nicht drucken Papier sparen.

Die Möglichkeit Bilder konvertieren zu lassen gibt es auch zum Ausdruck. Man kann die Dialogbox zum Einstellen der Konversionsparameter vom Hardcopy-Dialog aus aufrufen (mit dem Button einstellen, die jeweilige Einstellung wird kurz in zwei Info-Zeilen angegeben. Grundsätzlich werden Bilder nur monochrom gedruckt. Läßt man Farbbilder drucken, so werden wie für die Anzeige auf einem *monochromen* Monitor konvertiert (nur bei der Auswahl von Bitmaps wesentlich, sonst wird eh nur eine Bitmap erzeugt). Das Skalieren von Bildern ist natürlich auch möglich, man sollte aber beachten, daß – anders als bei n-fachen Ausgabe von Pixeln – erst das ganze Bild skaliert wird (was natürlich Speicher benötigt).
Weiter ist noch anzugeben, welcher Abstand zwischen zwei Bildzeilen gelassen werden soll (die Höhe der Bildzeile ergibt sich aus dem höchsten der in die Spalten zu druckenden Bilder). Möglich ist klein, mittel und groß sowie Parameter. Letzteres funktioniert *nur* bei *einspaltiger* Ausgabe. Dann werden unter jedes Bild dessen Parameter gedruckt, soweit von den Routinen unterstüzt (bisher alle außer XCHAOS-Routinen (Barnsley, Attraktoren, Diffusion, ...). Bei mehrspaltiger Ausgabe entspricht Parameter 'groß. Zuletzt kann man noch einstellen, nach wievielen Bildzeilen ein Seitenvorschub (Formfeed) aufgerufen werden soll (funktioniert nur innerhalb EINES Hardcopy-Aufrufs, die Routine merkt sich nicht wieviele Bildzeilen sie das letzte mal ausgegeben hat). Stellt man 0 ein, so wird kein Seitenvorschub ausgelöst.
Mit den Button LF, FF und Init kann man die entsprechenden Druckercodes direkt an den Drucker schicken (um einen Zeilenvorschub (LineFeed, LF) oder einen Seitenvorschub (FormFeed FF) auszulösen, beziehungsweise den Drucker zurückzusetzen (Init).
Zum Verlassen der Box gibt es neben dem üblichen Abbruch (Verlassen ohne Drucken) und Ok (Verlassen mit Drucken) noch Einst., hier werden die Einstellungen als Voreinstellungen übernommen, ohne daß anschließend gedruckt wird (wenn ich mich nicht irre, werden die Einstellungen auch bei Ok in die Voreinstellungen übernommen (was sich ändern könnte)).

Die Bilder werden invers gedruckt, wenn sie auf dem SM124 (also dem monochromen Monitor) so angezeigt werden. Dies ist dann der Fall, wenn die Farbe 0 den Blau-Anteil 0 aufweist, wobei beim konvertieren natürlich unter Umständen eine neue Farbtabelle erzeugt wird.
Paßt ein Bild nicht in den Platz, der für es zur Verfügung steht, ist das Bild also breiter (oder bei quer höher) als die Spalte, so wird nur der rechte bzw. untere(? probiert's aus, ich weiß es gerade nicht) Teil des Bildes gedruckt.
Zu den Spalten ist noch zu sagen, daß sie vor allem für den Druck gleich großer Bilder gedacht sind; läßt man Bilder verschiedener Größe in Spalten drucken, so ist das Ergebnis eher unbefriedigend.

2 Gdos-Treiber

Der Gdos-Treiber liegt nur in einer Version V0.1 vor, und das soll heißen, daß es sich hier um eine höchst vorläufige, nicht ausgefeilte Version handelt (zumal das Ausgeben von Bitmaps via Gdos nu wirklich keine Freude ist).

Parameter

An Parametern ist zunächst einmal der gewünschte Gdos-Treiber einzustellen, wozu die ersten neun Druckertreiber angezeigt werden (die Treiber werden aus der *ASSIGN.SYS*-Datei gelesen, wird diese Datei nicht gefunden, so kann der Treiber nicht eingestellt werden. Der eingestellte Treiber wird jeweils gleich geladen, um die möglichen Zeichensätze und ihre Größen ermitteln zu können.
Zum Treiber stellt man dann noch den zu verwendenden Zeichensatz (hier werden die ersten acht angezeigt) und des Größe ein.
Unter den Einstellungen zum Treiber ist ein Pfad für Image-Dateien anzugeben. Der Treiber druckt über die Gdos-Funktion zum Ausdrucken von GEM-Image-Dateien, die vorher gespeichert werden. Wählt man die Option Dateien anschließend löschen, so werden die Dateien nach dem Drucken gelöscht. Bleibt noch die Option CR ausgeben. Sie dient dazu, ein Problem zu umgehen, das mit meinem Drucker und meinen Gdos-Treibern auftrat. Nach der letzten Zeile wird vom Gdos-Treiber kein Wagenrücklauf (CR) mehr ausgeführt, die erste Zeile einer weiteren Ausgabe wird deshalb zu weit rechts ausgegeben. Selektiert man CR ausgeben, so wird nach der Ausgabe noch ein CR an den Drucker geschickt (via TOS, d.h. dies geht nur bei Anschluß des Druckers an die paralelle bzw. serielle Schnittstelle (vgl. 24Nadeldrucker-Hardcopy). Gleichzeitig wird bei aktivieren dieser Funktion vor Aufruf der Hardcopy getestet, ob der Drucker empfangsbereit ist.

Hardcopy ausgeben

Hier wird die eigentliche Hardcopyroutine aufgerufen.
An Parametern hat man die Möglichkeit die Größe des Bildes, die Position (nur horizontal) sowie den (vertikalen) Abstand zwischen zwei Bildern. Beim letzteren kann man mit Parameter erreichen, daß nach jedem Bild dessen Parameter ausgegeben werden (soweit von den Routinen unterstützt). Dabei (und nur hier) wird der eingestellte

Zeichensatz verwendet. Die Möglichkeiten des Konvertierens und der Seitenvorschub nach Bildern entspricht den Möglichkeiten des 24 Nadelmatrixdrucker-Treibers.

Anmerkungen

Erstens: im Prinzip sollte es auch möglich sein, Metafile-Ausgabe zu machen. Allerdings habe ich noch kein Programm gefunden, das Metafiles mit mehr als einem GEM-Image laden und anzeigen kann. Die Metafile-Ausgabe ist deshalb und anderer Schwierigkeiten wegen nicht möglich.
Zweitens: meine NEC-P6 Druckertreiber haben eine merkwürdige Vorstellung über die Breite und Höhe (in Punkten) des Druckers. Dies führt dazu, daß zentrierte Bilder zu weit rechts erscheinen, und der Seitenvorschub nicht unbedingt richtig erfolgt.
This is not a bug in CHAOS*ultd*!
Drittens: die Ausgabe der GEM-Image-Dateien erfolgt immer monochrom!