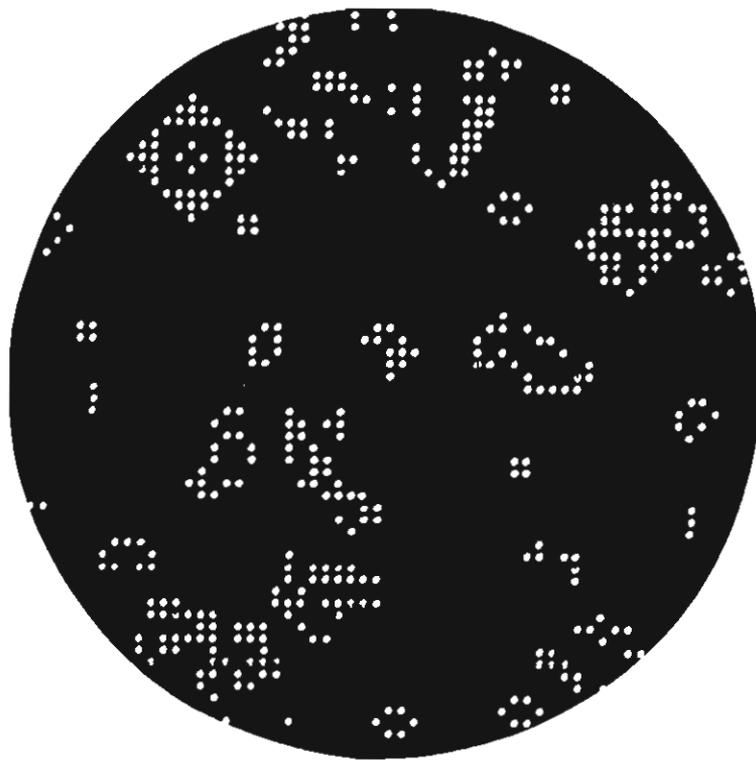

Interface

herausgegeben von der
Abt. Hybridrechenanlage des
EDV-Zentrums der
Technischen Universität Wien

Nummer 14
Dezember 1979



Life-Spiel am Oszillographen

INHALTSVERZEICHNIS

	Seite
AutoPATCH Hardware installiert!	3
Nordamerikanisches EAI - Computer Users' Group Treffen	4
Information an alle Kunden betreffend Datenschutzgesetz im EDV - Zentrum	4
Aktuelle Mitteilungen	5
Kurse	8
TALLY - Drucker	11
RECON	11
680 Componenten Overlay	12
Analoges Time - Sharing nichtlinearer Elemente	14
CAS - Seminar in Dubrovnik über Künstliche Intelligenz	19
Optimale Trassen für ein U - Bahn - Netz	21
Rechnererzeugte Punktbilder auf X/Y - Oszillograph - angewandt für das Life - Spiel	27
Vergleich verschiedener Klassifizierer für die automatische Erkennung von akusti- schen Signalen	31

Redaktion: Elisabeth Wittek

Eigentümer, Herausgeber, Verleger: EDV-Zentrum der Technischen
Universität Wien, Abteilung Hybridrechenanlage, Vervielfältigung:
Österreichische Hochschülerschaft Technik, für den Inhalt verant-
wortlich: Dipl.Ing. Dr.techn. W. Kleinert, alle: Gußhausstraße 27-29,
1040 Wien. Telex: 76875 rzthw a

AUTOPATCH

Hardware installiert !

Nachdem bereits Anfang Oktober ein erfolgreicher Abnahmetest für die AutoPATCH Schaltmatrix im Lieferwerk der Firma EAI in West Long Branch, N.J., USA, durchgeführt wurde erfolgte die Lieferung und Installation der Hardware für die AutoPATCH - Erweiterung der Hybridrechenanlage Mitte November. Am 19.11. wurde der vertragsmäßig vorgesehene Leistungstest erfolgreich durchgeführt und die Abnahme vorgenommen. Die Ergebnisse übertrafen hinsichtlich statischer Genauigkeit (Offset) und dynamischem Verhalten (Bandbreite) durchwegs die vereinbarten Spezifikationen. Wir werden im nächsten INTERFACE noch genauer über den Aufbau und die Funktionsweise dieser über 5120 analoge Schaltstellen verfügenden Switchmatrix berichten.

Die Arbeiten an dem von der Hybridrechenanlage selbst entwickelten Parallelen Logik Prozessor (PLP) haben ebenfalls gute Fortschritte gemacht. Dieser digital programmierbare Ersatz für die parallele Logik wird Anfang 1980 zur Verfügung stehen. Der genaue Aufbau des PLP sowie die Beschreibung einer neuentwickelten hybriden Hardware zur Darstellung hybrider Variablen und zur Erzeugung nichtlinearer Funktionen wird ebenfalls im nächsten INTERFACE zu finden sein.

An Software für das AutoPATCH-System ist neben der Installierung des hybriden Compilers ECSSL an der Hybridrechenanlage als unmittelbar nächster Schritt vor allem die Erweiterung des hybriden Prozessors HYBSYS geplant, wodurch das manuelle Stecken durch die automatische Verbindung von sogenannten "analogen Macros" ersetzt werden wird.

W. Kleinert

NORDAMERIKANISCHES EAI - COMPUTER

USERS' GROUP TREFFEN

In der Zeit vom 10. - 12. Oktober hatten Mitarbeiter der Hybridrechenanlage (W. Kleinert, D. Solar) die Gelegenheit, am diesjährigen Herbsttreffen der Nordamerikanischen Users' Group in Schenectady, N.Y., teilzunehmen.

Wir konnten uns nicht nur an Hand der Vorträge und der freundschaftlichen Diskussionen einen hervorragenden Überblick über die Tätigkeit der wichtigsten Hybridrechenzentren in den USA verschaffen, sondern hatten auch die Gelegenheit, in zwei Vorträgen den bei uns entwickelten hybriden Prozessor HYBSYS und über unsere Entwicklungspläne für ein hybrides Time - Sharing - System zu berichten.

INFORMATION AN ALLE KUNDEN

BETREFFEND DATENSCHUTZGESETZ IM EDV-ZENTRUM

Praktische Durchführung an der TU - Wien:

In der Vorstandssitzung des EDV - Zentrums der TU - Wien am 20. November 1979, wurde über das Datenschutzgesetz diskutiert und beschlossen, den Kunden des EDV - Zentrums folgendes zur Kenntnis zu bringen:

Die Auftraggeber (Benützer der EDV-Anlagen) sind verpflichtet, das EDV - Zentrum zu informieren, falls im Sinne des Datenschutzgesetzes sensitive Daten zur Verarbeitung gelangen.

Wird seitens der Benutzer nicht ausdrücklich auf einen solchen Sachverhalt hingewiesen, so wird keine allfällige Kollision mit dem Datenschutzgesetz vorausgesetzt, was mit der bisherigen Gepflogenheit übereinstimmt, daß die Rückgabe der Ergebnisse der EDV - Arbeit in offener Form erfolgt.

aktuelle mitteilungen

ÖFFNUNGSZEITEN

Die Hybridrechenanlage ist von

Montag bis Freitag

von

8 Uhr bis 20 Uhr

geöffnet.

Am 27., 28. und 31. Dezember 1979 wird wegen Systemarbeiten nur ein eingeschränkter Closed Shop-Betrieb möglich sein.

PERSONELLES

Seit Mitte Juli ist Frau I. Husinsky für ein Jahr in Amerika. An ihrer Stelle ist Frau E. Wittek an der Hybridrechenanlage angestellt, die während der Abwesenheit von Frau Husinsky auch die Redaktion des INTERFACE übernommen hat.

NEUE DRUCKSORTEN

Seit Juni ist ein User-Manual erhältlich, das sämtliche wichtigen Informationen für die Benützung des Multiprogramming Betriebssystems JCS/TS 7 auf einem EAI PACER 600 Hybridssystem enthält.

NEUE HARDWARE

Seit kurzem ist ein neues Drucker-Terminal TALLY T1612 an der Hybridrechenanlage installiert (siehe Seite 11).

An neuen Interfacekomponenten stehen ab sofort 64 zusätzliche Controllines zur Verfügung, wobei die Controllines 0 - 59 den (nunmehr logischen) Trunks 0 - 59 und die Controllines 60 - 63 den logischen Trunks 0 - 3 zugeordnet sind.

Ferner wurden die digitalen Potentiometer erweitert. Ab sofort stehen 60 4-Quadranten-DCAs (0 - 59) zur Verfügung, es bleiben somit 60 Servopots (60 - 119).

Achtung!: Programme, die bisher die Potentiometer 40 - 59 verwendet haben, müssen geändert werden.

NEUE SOFTWARE

Zur Wahl des Schriftbildes am TALLY-Drucker und zum Setzen und Zurücksetzen der 64 zusätzlichen Controllines wurden neue Subroutines entwickelt.

Für die Benutzer des Analogrechners wurde ein digitales Unterstützungsprogramm zum Auslesen und Setzen der einzelnen Komponenten entwickelt (siehe Seite 12).

String-Manipulations-Routinen

Die in der THLIB1 des Digitalrechenzentrums zur Verfügung stehenden Character- und String-Manipulations-Routinen können nun auch an der Hybridrechenanlage benützt werden.

Die Programme wurden bezüglich Aufruf und Funktion unverändert übernommen, sie sind daher mit den Routinen der THLIB1 völlig kompatibel. Veränderungen wurden jedoch an den internen Basisroutinen vorgenommen, um sie der 2 Byte per Wort/ 16 Bit per Byte - Konfiguration des PACER so anzupassen, daß kürzestmögliche Laufzeiten erreicht werden.

Eine neu geschriebene Subroutine COLSEQ erlaubt die beliebige Umdefinition der Collating Sequence auf der Basis des 64-Zeichen-ASCII-Zeichensatzes.

Normalerweise liegt der Zeichenbewertung die Standard-ASCII-Collating Sequence zugrunde, und der Benutzer kann nach Verwendung einer anderen Sequence durch Setzen eines Indikators auch jederzeit zu dieser zurückkehren.

Das Paket besteht aus folgenden Routinen:

ISEKCH sucht Characterposition in einem String
ISEKST sucht Stringposition in einem String
ISKNTC sucht Position des ersten Characters in einem String, der von einem gegebenen Character verschieden ist
KOMPCH vergleicht Characters
KOMPST vergleicht Strings
MAKINT bildet ASCII-Zeichen auf Integer ab
MOVSTR überträgt String in einen anderen String
MOVOST überträgt String in einen anderen String, Überlappen der Strings ist erlaubt
MOVCHR überträgt Character eines Strings in einen anderen String
MOVINT überträgt Character in einen String
MAKII holt N-Bit Integer aus String

Beschreibungen des Subroutine-Pakets sind erhältlich.

kurse

Folgende Kurse werden im Wintersemester 1979/80 an der Hybridrechenanlage abgehalten werden:

RH1 GERÄTETECHNIK EAI-PACER 600 AUTOPATCH-SYSTEM

Dauer: 3 Tage

Termin: 1980/01/14 bis 1980/01/16

je 4 Stunden

Vortragender: Dr. W. Kleinert

RH3 HINWEISE FÜR FORTRAN-PROGRAMMIERER AN DER HYBRIDRECHENANLAGE

Dauer: 1 Tag

Termin: 1980/03/11

1 Stunde

Vortragender: Dipl. Ing. F. Blöser

RH6 EAI ASSEMBLER

Dauer: 5 Tage

Termin: 1979/12/10 bis 1979/12/14

je 4 Stunden

Vortragender: Dipl. Ing. F. Blöser

RH7 SOFTWAREUNTERSTÜTZUNG FÜR DIE BENÜTZUNG DES PACER 600 ALS PLOTTER SYSTEM

Dauer: 2 Tage

Termin: 1980/03/11 bis 1980/03/12

je 3 Stunden

Vortragender: Dipl. Ing. F. Blöser

RH8 TERMINALPROGRAMMIERUNG

Dauer: 1 Tag

Termin: 1980/03/13

1 Stunde

Vortragender: E. Wittek

RH9 EINFÜHRUNG IN DIE HYBRIDE PROGRAMMIERUNG

Dieser Kurs wird nach Bedarf für wissenschaftliche Benutzer, die keinerlei Vorkenntnisse auf dem Gebiet der hybriden Programmierung besitzen und an einer möglichst raschen Problemlösung interessiert sind, abgehalten.

Dauer: 2 Tage

Vortragender: D. Solar

Nähere Auskünfte bei Hr. D. Solar (1040 Wien, Gußhausstraße 27-29, 4. Stock, Zimmer 1401, Tél. 65-37-85/747 DW).

RH10 DATENVERBINDUNG MIT CYBER 74

Dauer: 1 Tag

Termin: 1979/12/04

1 Stunde

Vortragender: Dipl. Ing. F. Blöser

RH11 ASSEMBLER-PROGRAMMIERUNG FÜR FORTGESCHRITTENE
MIT ÜBUNGEN

Dauer: 5 Tage

Termin: 1979/12/17 bis 1979/12/21

je 3 Stunden

Vortragender: Dipl. Ing. A. Blauensteiner und
Dipl. Ing. F. Blöser

Kursort: 1040 Wien, Gußhausstraße 27-29, 4. Stock, Zimmer 1401
Auskünfte und Anmeldungen: Hr. M. Schandl, 65-37-85/803 DW.

Vom Institut für Technische Mathematik werden folgende Kurse abgehalten:

AH1
AH2 PROGRAMMIEREN VON ANALOGRECHENANLAGEN

Grundlagen und Anwendungsmöglichkeiten des Analogrechners mit praktischen Übungen an den Rechnern EAI-10000 und EAI-680

Termin: 1980/02/18 - 1980/02/22

AH3
AH4 HYBRIDE SPRACHEN UND VERFAHREN

Grundprinzipien und Einsatzgebiete des Hybridrechnens, aufbauend auf Vorkenntnissen über Analogrechnen, wie sie in den Kursen AH1 und AH2 vermittelt werden. Simulationssprache CSMP, hybride Programmiersprache HOI, FORTRAN-Hybrid-Ergänzungen.

Insbesondere wird das auf das AutoPATCH-System ausgerichtete Programmpaket HYBSYS, das die Benützung des Analogrechners wesentlich vereinfacht (automatische Skalierung!), vorgestellt und besprochen.

Termin: 1980/02/25 - 1980/02/29

Auskünfte und Anmeldungen zu den Kursen bei Dipl. Ing. Dr. F. Breitenacker oder bei Dipl. Ing. Frank Rattay, Institut für Technische Mathematik (TU Wien, Gußhausstraße 27-29, Tel. 65-37-85/829 oder 899 DW).

TALLY-DRUCKER

An der Hybridrechenanlage ist seit kurzem ein Matrixdrucker TALLY T1612 installiert. Dieser Matrixdrucker ersetzt das defekte Teletype-Terminal.

Der TALLY-Drucker hat insgesamt 6 verschiedene Schreibdichten, sodaß auf dem an der Hybridrechenanlage üblichen 80-Spalten-Papier (A4) programmgesteuert von 43 bis 141 Spalten/Zeile gewählt werden können. Durch sein angeschlossenes Keyboard kann der TALLY-Drucker auch als Eingabeeinheit angesprochen werden.

Seine hohe Geschwindigkeit (160 Zeichen/Sekunde) erlaubt zusammen mit einem internen 1 kB Buffer einen hohen Zeichendurchsatz.

Der TALLY-Drucker kann normal über FORTRAN Ein/Ausgabe (2/1) sowie über spezielle Routinen zur Einstellung des Schriftbildes angesprochen werden. Unterlagen darüber sind an der Hybridrechenanlage, Zimmer 1404 erhältlich.

RECON

Wir möchten unsere Kunden darauf hinweisen, daß am EDV-Zentrum der Technischen Universität Wien, Abteilung Hybridrechenanlage, seit dem Jahr 1978 Zugriff zum Literatursuchsystem RECON der European Space Association (ESA) mit ihrem Zentralrechner in Frascati, Italien besteht. Dieser Kundendienst, der ausschließlich der Versorgung der Institute der Technischen Universität Wien dient, wird als Außenstelle der Österreichischen Gesellschaft für Sonnenenergie und Weltraumforschung betrieben und vom Bundesministerium für Wissenschaft und Forschung finanziert. Seit Bestehen dieser Einrichtung wurde bereits eine beträchtliche Anzahl von On-Line Literaturrecherchen abgewickelt.

Falls sie Interesse an dieser effizienten Art der Literatursuche haben, wenden sie sich bitte an:

EDV-Zentrum der Technischen Universität Wien,
Abteilung Hybridrechenanlage,
H. Hummer
Gußhausstraße 27-29, 1040 Wien
Tel.: 65-37-85/901 Durchwahl

680 COMPONENT OVERLAY

Informationen über bestimmte Zustände des Analogrechners und seiner Komponenten benötigt nicht nur der hybrid, sondern auch der rein analog arbeitende Benutzer. Es kann sich dabei etwa um das Auslesen von Verstärkern, digitalen Potentiometern oder Senselines handeln, um das Lesen des Interrupt Status oder das Setzen von D/A-Wandlern oder des digitalen Timers.

Ein Teil dieser Aufgaben kann zwar manuell am Keyboard bewältigt werden, dies stört jedoch den laufenden Rechenvorgang, besonders dann, wenn sich zwei Benutzer die Rechenkapazität des Analogrechners teilen, da das Keyboard die Zustände aller Komponenten steuert.

Dazu kommt, daß ein anderer, ständig wachsender Teil der Komponenten nur digital bedient werden kann und damit dem rein analog arbeitenden Benutzer nicht mehr direkt zugänglich ist.

Der hybrid arbeitende Benutzer kann, sofern er mit FORTRAN und den Hybrid Linkage Subroutines arbeitet, praktisch sämtliche Komponenten bedienen und Zustände bzw. Werte abfragen, nur für ganz spezielle Erfordernisse sind in Assembler geschriebene Programme notwendig. Er befindet sich daher in einer günstigeren Situation.

Es sollte also auch für den Analog-Benutzer ein ausbaufähiges digitales Unterstützungsprogramm zur Verfügung stehen. Wie die Praxis zeigt, haben auch fast alle Hybrid-Benutzer kleine (teils sogar interaktive) Teile zu ihren Programmen geschrieben, um mit den für sie wichtigen Komponenten komfortabel arbeiten zu können. Es handelt sich dabei im wesentlichen um dieselben Aufgaben, sodaß sie durch ein derartiges Hilfsprogramm entlastet werden könnten.

Das Hilfsprogramm wurde als Overlay konzipiert und kann daher im System JCS/TS 7 laufen, ohne eine Partition zu belegen.

Für den Hybrid-Benutzer bietet sich somit auch die Möglichkeit, die Aktionen seines in einer Partition laufenden Programmes zu prüfen und eventuell einzugreifen.

Der 680 COMPONENT OVERLAY wird von der Systemkonsole gestartet und benützt diese auch als Ausgabedevise. Die notwendige Eingabe besteht in Ein- oder Zwei-Zeichen-Codes, die am Bildschirm automatisch zu Texten ergänzt werden.

Derzeit können folgende Komponenten bedient werden:

DCA	Digitale Potentiometer	Setzen/Lesen
FR	Function Relay	Setzen
CO	Komparator	Lesen (auch alle 16 Komparatoren in Binärdarstellung)
SL	Senseline	Lesen (auch alle 8 Senselines in Binärdarstellung)
CL	Controlline	Setzen (auch alle 16 Control-lines in Binärdarstellung)
AD	A/D-Wandler	Lesen
DA	D/A-Wandler	Setzen (inklusive Bedienung des JAM-F/F)
GPI	General Purpose Interrupt	Lesen des Status aller GPI's in Binärdarstellung
RT	Real Time Clock	Lesen/Setzen

Derzeit können die Elemente unter ihrer Patch Panel Nummer angesprochen werden. An der Möglichkeit, einen durch unteres und oberes Limit definierten Bereich bzw. alle Elemente durch den Code * anzusprechen, wird gearbeitet.

Außerdem ist folgende Erweiterung vorgesehen:

Bedienung von Servopots/Handpots
Amplifier/Trunk
ABCD Timer
DMA Status

Durch Eingabe eines Blanks von der Konsole werden die wichtigsten Zustände des Analogrechners wie Logic Mode und Time Scale und Teile des Fault- und des Status-Wortes (Overload- und Overrange-Indikator) en bloc gelistet.

Alle Benutzer, die im Open Shop analog oder hybrid arbeiten, werden eingeladen, anhand der aufliegenden Beschreibung den 680 COMPONENT OVERLAY zu benützen. Um beim Arbeiten des Benutzers an der Systemkonsole negative Auswirkungen auf das laufende Betriebssystem durch Fehlbedienung, die trotz softwaremäßiger Absicherung denkbar sind, auszuschalten, ist bei Benützung des Overlays das Heranziehen eines Operators unbedingt erforderlich.

H. Hummer

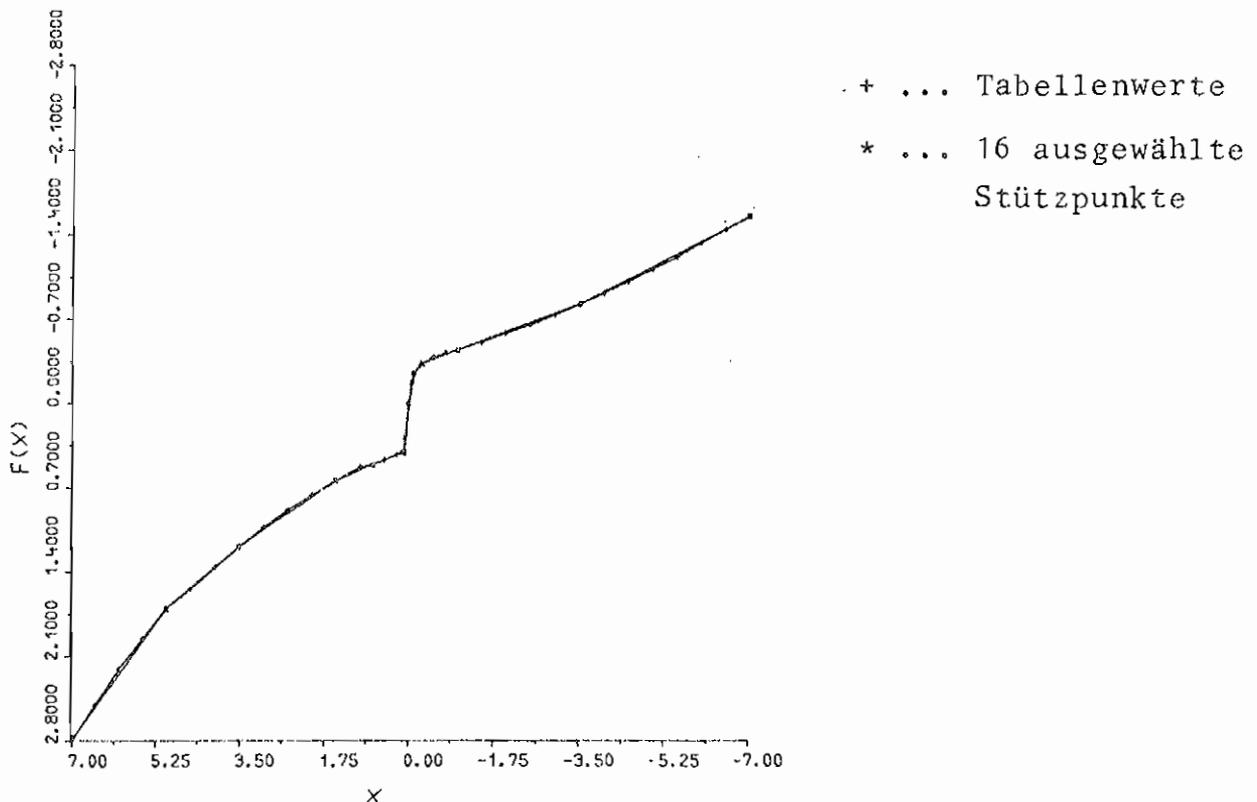
ANALOGES TIME-SHARING NICHTLINEARER ELEMENTE

Bei der Implementierung eines Modells am Hybridrechner kann man vor dem Problem stehen, daß der Komponentenbedarf des Rechenmodells mit den vorhandenen Nichtlinearitäten nicht gedeckt werden kann.

Falls eine "direkte" Implementierung des hybriden Modells wegen der zu geringen Anzahl von Nichtlinearitäten an dem zur Verfügung stehenden Analogrechner nicht möglich ist, so kann in vielen Fällen durch das im folgenden beschriebene Verfahren des analogen Time-Sharing nichtlinearer Elemente eine zufriedenstellende Lösung erzielt werden.

Das Prinzip des analogen Time-Sharing nichtlinearer Elemente soll an einem sehr einfachen Beispiel erläutert werden.

Gegeben sei eine nichtlineare Funktion (z.B. anhand von Tabellenwerten). Durch das Hybrid-Linkage-Unterprogramm DCFG wurden 16 Stützpunkte so ausgewählt, daß die Funktion möglichst gut approximiert wird.



In einer Analschaltung wird nun diese durch einen Funktions-
 geber realisierte Nichtlinearität $f(x)$ zweimal mit unterschied-
 lichem Argument benötigt, also die Funktionen

$$f(x_1), f(x_2),$$

wobei $x_1(t), x_2(t)$ beliebige zeitabhängige Verstärkerausgänge
 darstellen. Abbildung 1 zeigt z.B. für $t \in [0, T]$ und $f(x)$ die
 Funktionen $x_1(t) = 1-t$ und $f(-x_1)$, Abb. 2 die Funktionen $x_2(t) =$
 $= 1/4 * x_1(t)$ und $f(-x_2)$.

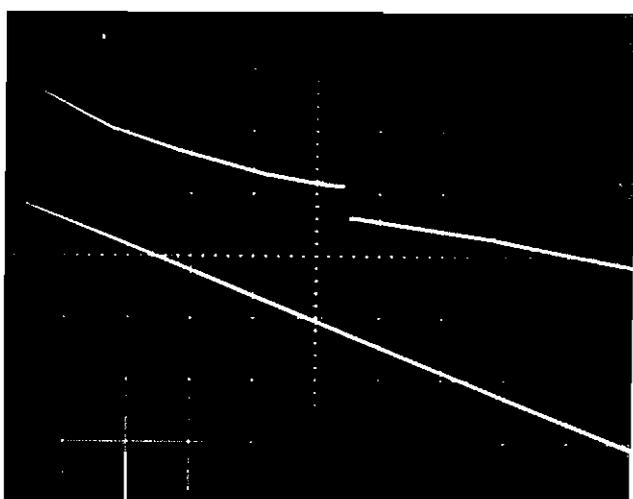


Abbildung 1

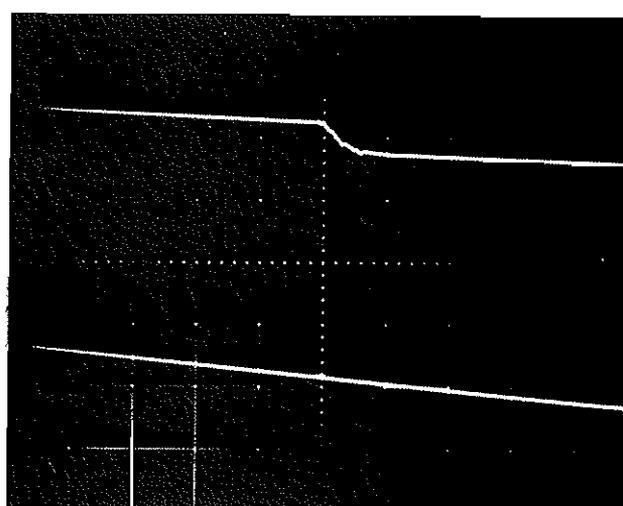
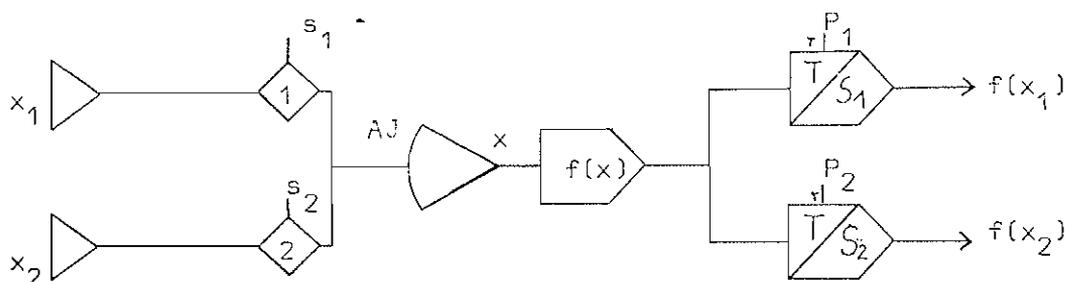
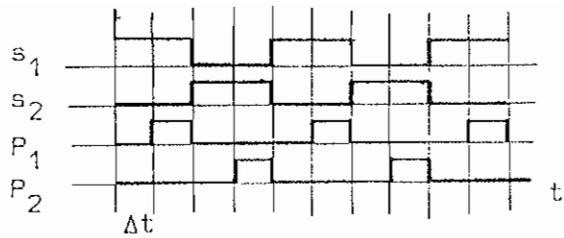


Abbildung 2

Dann kann man eine Ersatzschaltung mit nur einem Funktions-
 geber folgendermaßen aufbauen:





Zeitdiagramm
der Logikschaltung

$x_1(t)$ liegt an Schalter 1, $x_2(t)$ an Schalter 2 eines Multi-Switch-Summierers, der mit den komplementären logischen Signalen s_1 und s_2 gesteuert wird. Am Funktionsgebereingang liegt nun eine unstetige Funktion $x(t)$, die durch Umschalten im Abstand $2 * \Delta t$ zwischen $x_1(t)$ und $x_2(t)$ erhalten wird. Abb. 3 zeigt oben $x(t)$ und unten den Ausgang des Funktionsgebers $f(x)$, wobei $T = 2$ msec ist, was bei dem Δt von 50 μ sec genau die 20 Umschaltungen ergibt.

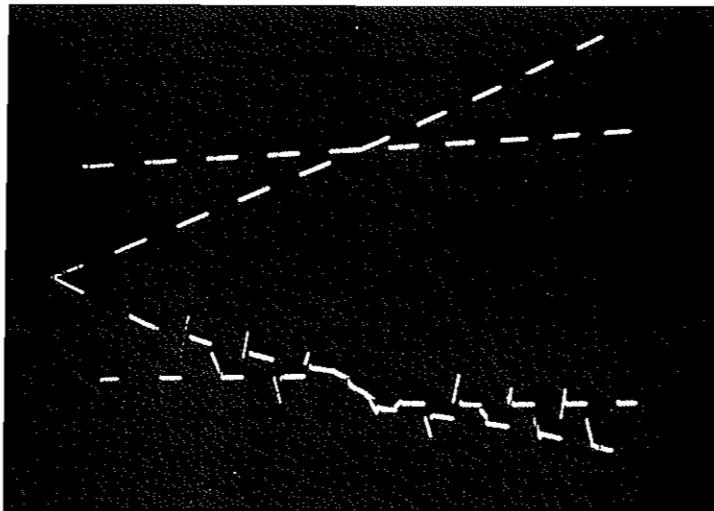


Abbildung 3

Dem Funktionsgeber nachgeschaltet sind nun zwei Track/Store-Summierer, die mit den logischen Signalen p_1 und p_2 gesteuert werden. Wie aus dem obigen logischen Diagramm ersichtlich, wartet die Schaltung zuerst Δt Zeiteinheiten, damit die T/S-Einheiten die Schaltspitzen des Funktionsgebers nicht übernehmen, dann übernimmt der T/S den Wert des Funktionsgebers, den er nach Δt Sekunden speichert. Abb. 4 zeigt den Ausgang von T/S 1 im Vergleich zum idealen $f(x_1)$, im gleichen Maß-

stab wie die vorigen Abbildungen (vertikal in eine Skaleneinheit zur besseren Übersicht verschoben). Abbildung 5 zeigt wieder T/S 1 und $f(x_1)$, aber diesmal mit $T = 0.2$ sec und $\Delta t = 50 \mu\text{sec}$, es ist praktisch kein Unterschied mehr festzustellen.

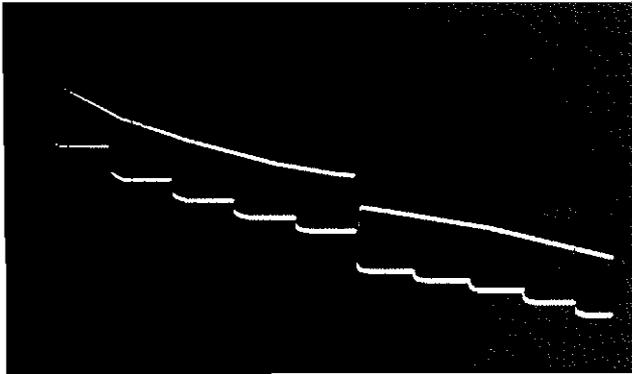


Abbildung 4

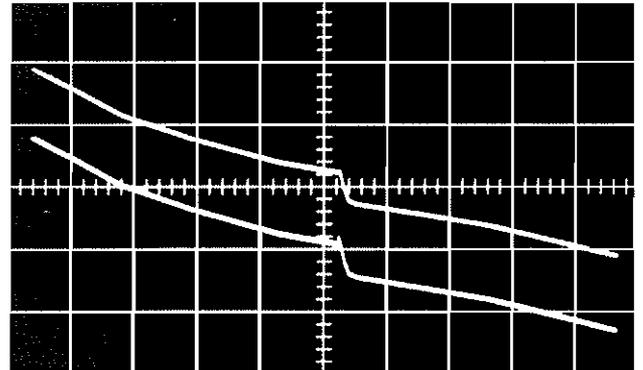


Abbildung 5

Abb. 6 ist das Analogon zu Abb.3, aber diesmal mit $T = 0.2$ sec. Die 2000 Umschaltungen erfolgen relativ zur Rechenzeit jetzt so schnell, daß praktisch der Eindruck von 4 Kurven vermittelt wird.

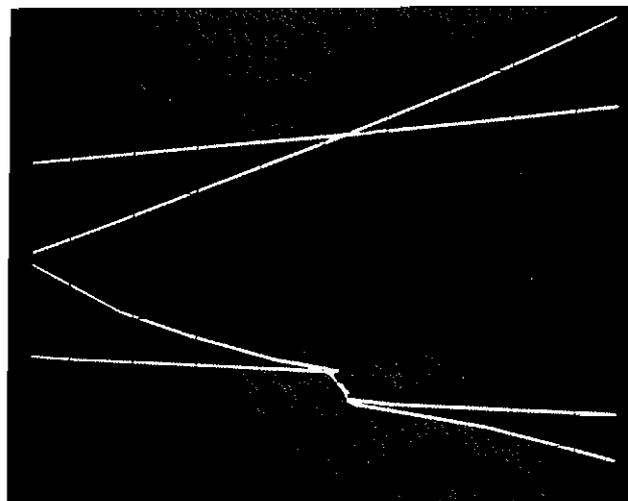


Abbildung 6

Bei diesem einfachen Beispiel war nur zwischen 2 Argumenten umzuschalten. Benötigt man Funktionsauswertungen von n ($n > 2$) unabhängigen Variablen, so kann die Schaltung leicht erweitert werden: man braucht dann n D/A-Schalter beim Multi-Switch-Summierer für x und n T/S-Einheiten. Die entsprechenden logischen Steuersignalen s_1, \dots, s_n und p_1, \dots, p_n können am besten mit einem Ringschieberegister erzeugt werden. Zu beachten ist auf jeden Fall, daß wegen des hardwaremäßig bedingten $\Delta t = 50 \mu\text{sec}$ die Funktionen $f(x_j)$ nur zu diskreten Zeitpunkten $t_j^{(i)} = t_{j-1}^{(i)} + 2n * \Delta t$ ausgewertet werden und daß $t_j^{(i)} = t_j^{(i-1)} + \Delta t$ gilt, falls nicht doppelt so viele T/S-Einheiten verwendet werden. Es sei darauf hingewiesen, daß die Updatefrequenz des analogen Time-Sharing ($1/(t_j^{(i)} - t_{j-1}^{(i)})$) nur von Δt und der Anzahl der Funktionsauswertungen n , aber nicht von der Komplexität der Funktionsauswertungen abhängt. Wenn also statt $f(x)$ mehrere Funktionen $f_k(x_1, \dots, x_m)$ von mehreren Veränderlichen vorliegen, kommt es wieder nur auf die Anzahl der Auswertungen \bar{n} , da die verschiedenen f_k parallel ermittelt werden.

Diese eben beschriebene Methode zur Einsparung nichtlinearer analoger Rechenelemente läßt sich nur dann anwenden, wenn die Schaltung mehrere gleichartige Schaltkreise enthält. Außerdem muß man sich darüber im klaren sein, daß durch das Einsparen von Nichtlinearitäten mehr lineare Rechenelemente (Track/Store-Einheiten, Multi-Switch-Summierer) benötigt werden.

Im oben beschriebenen Beispiel haben $x_1(t)$ und $x_2(t)$ immer das gleiche Vorzeichen, d.h. sie liegen im gleichen Quadranten. Dadurch ist gesichert, daß das Eingangssignal des DCFG keine Sprünge >1 aufweist und die schnellen Schaltzeiten realisiert werden können. Haben die Argumente verschiedenes Vorzeichen (d.h. es können Sprünge der Höhe 2 am Eingangssignal auftreten), kann es kurzzeitig zu Übersteuerungen kommen und der DCFG braucht länger, um den richtigen Funktionswert zu übernehmen. Das bedeutet aber, daß Δt größer sein muß und daher eine langsamere Rechengeschwindigkeit notwendig wird.

W. Kleinert

KÜNSTLICHE INTELLIGENZ

Das Zentrum für fortgeschrittene Studien in Belgrad (CAS) veranstaltet jährlich ein internationales Sommer-Seminar in Dubrovnik, das den Informationsaustausch und die Diskussion von aktuellen Entwicklungen auf dem Gebiet der System- und Computerwissenschaften zum Ziel hat. Diese zweiwöchigen Seminare bestehen aus einer Reihe von Vorlesungen anerkannter Wissenschaftler, die mit Arbeitsgruppen und freien Diskussionsrunden gekoppelt sind.

Das diesjährige Seminar befaßte sich mit Künstlicher Intelligenz, einer relativ jungen Wissenschaft, die die theoretischen Grundlagen, die Methoden, die Design-Kriterien und den Aufbau von intelligenten Systemen untersucht.

Die Vorträge wurden von Patrick H. Winston, der das Seminar leitete, R. Davis, K. P. Horn, W. Martin, G.J. Sussman vom Artificial Intelligence Laboratory des Massachusetts Institute of Technology, E. Sandewall vom Informatic Laboratory der Universität Uppsala und M. Somalvico vom Polytechnic Artificial Intelligence Project der Universität Mailand gehalten. Die 80 Teilnehmer setzten sich aus Studenten, Professoren und Fachleuten der Industrie aus 8 europäischen Staaten und der UdSSR zusammen.

Der erste Teil des Seminars brachte nach einer Einführung in die künstliche Intelligenz und die Programmiersprache LISP als eines der grundlegenden Hilfsmittel zur Beschreibung intelligenter Systeme einen Überblick über die Schwerpunkte und Methoden dieser Wissenschaft wie Problembeschreibung und Durchführung (Problem Representation), Lernen (Learning), das Verstehen natürlicher Sprachen (Natural Language Understanding), Bildanalyse und Verarbeitung (Machine Vision), Objektmanipulation, auf Sachkenntnis und/oder auf Regeln aufgebaute Systeme (Knowledge (Rule) Based Systems), Wissensverarbeitung u.ä.

Die Möglichkeiten dieser Methoden der Künstlichen Intelligenz wurden im zweiten Teil des Seminars an einigen konkreten Anwendungen wie DENDRAL, einem chemischen System, das aus Formeln, der Kenntnis der möglichen Verbindungsarten und der Massenspektren die Molekularstruktur berechnet, oder MYCIN, ein medizinisches System zur Diagnose von Infektionskrankheiten, das, ähnlich biologischen Bestimmungsbüchern, auf Regeln aufgebaut ist, die sukzessive von einer Grobbestimmung zur Feinbestimmung führen, veranschaulicht. Besonders interessant und auch didaktisch gut aufgebaut waren die Vorträge von K.Horn über Machine Vision und von W. Martin über natürliche Sprachen.

Aufgabe des letztgenannten Zweiges der Künstlichen Intelligenz ist es, einen tieferen Einblick in das Wesen der natürlichen Sprache als Informationsträger und Vermittler zu gewinnen und sie zur Kommunikation zwischen Mensch und Maschine heranzuziehen. Hier liegen die Probleme vor allem in einer geeigneten Darstellung und Behandlung der Syntax und Semantik bzw. semantischer Zusammenhänge, und auf einer weiteren Ebene in der Behandlung von Aussagen und Fragen, die erst die Kommunikation ermöglichen. Bei der Beschreibung der Syntax wird von der Annahme ausgegangen, daß es für alle Zeitwörter eine kleine Menge von Einsetzungsklassen gibt, sodaß die Wortgruppen aus diesen Klassen mit dem Zeitwort einen syntaktisch richtigen Satz ergeben, d.h. für jeden Satz können die darin vorkommenden Wortgruppen eindeutig einer Einsetzungsklasse zugeordnet werden, die man durch Satzanalyse erhält.

Beispiele für Einsetzungsklassen:

wer, was?		wen, was?		womit, wodurch?
AGENT		OBJEKT		INSTRUMENT
wo?	wann?		wohin, woher?	wielange
ÖRTLICHE	ZEITLICHE		ÖRTLICHE	ZEITLICHE
LOKALISIERUNG			ORIENTIERUNG	

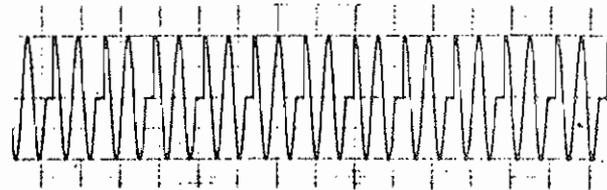
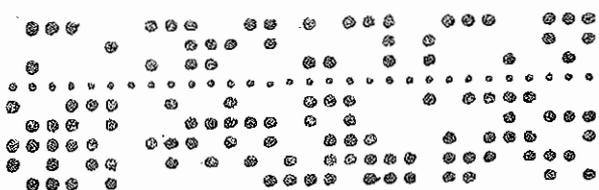
Die Machine Vision befaßt sich mit der visuellen Wahrnehmung in einer mehr oder weniger wohldefinierten Umwelt. Ihre Hauptaufgabe besteht in der Umweltbeschreibung aus den als Bild erhaltenen Informationen über die Umwelt und die Art der visuellen Wahrnehmung. Als wesentliche Probleme treten dabei das Erkennen von Strukturen und Bewegungen unter Ausschaltung optischer Täuschungen, das Erkennen von Objekten durch Vergleich mit vorgegebenen Prototypen und die Bestimmung von Objektposition und Orientierung auf. Die zur Lösung dieser Probleme angewandten Methoden reichen von der Wahl des Wahrnehmungssystems (Binäre (schwarz-weiße) und mehrstufige Mono- bzw. Stereowahrnehmung bei fester oder beweglicher, gleichmäßiger oder strukturierter Lichtquelle) bis zu komplexen, theoretisch fundierten Verfahren. So können z. B. räumliche Strukturen bei bekannter Oberflächenbeschaffenheit und bekannter Lage der Lichtquelle aus der Intensität des reflektierten Lichtes erkannt werden.

Abschließend wäre noch zur Organisation des Seminars kritisch zu bemerken, daß die rechtzeitige Zurverfügungstellung einer etwas detaillierteren Information über den Ablauf des Seminars sowie Angaben über die verwendete Literatur (siehe Lit [2]) eine große Hilfe gewesen wären.

O. Solar

Literatur:

- [1] Bertram Raphael: The Thinking Computer
W.H. Freeman and Company, San Francisco, 1976
(Eine einfach zu lesende Einführung)
- [2] Patrick H. Winston: Artificial Intelligence
Addison-Wesley Publishing Company, Reading, 1977
(Enthält ausführliche Literaturangaben)



benutzerforum

OPTIMALE TRASSEN FÜR EIN U-BAHN-NETZ

F. Berger, E. Wittek

Diplomarbeit - ausgeführt am Institut für Technische Mathematik
Technische Universität Wien

Betreuer: I. Troch

1. EINLEITUNG

Im folgenden soll über zwei Diplomarbeiten, die sich mit der energieoptimalen Trassierung eines U-Bahn-Netzes befassen, berichtet werden. Da die Energiekosten ständig wachsen und immer mehr Energie verbraucht wird, ist das Problem der optimalen Ausnützung der Energie und der Reduzierung des Verbrauchs sehr aktuell.

1975 haben sich Hoc.H.Hoang, Michael P. Polis und Alain Haurie [2] mit dem Problem der energieoptimalen Trassierung des Metro-Netzwerkes von Montreal beschäftigt. Die Autoren verwendeten für ihr Verfahren hauptsächlich heuristische Überlegungen und erzielten mit einem einfachen Verfahren sehr gute Resultate.

2. PROBLEMSTELLUNG

Eine U-Bahn-Trasse soll so festgelegt werden, daß die Züge in beiden Fahrrichtungen energieoptimal betrieben werden können, wobei gefordert wird, daß bei vorgegebener Trasse die Züge zeitoptimal gesteuert werden. Die Lage der Stationen kann man als fest vorgegeben annehmen, da die örtlichen Gegebenheiten (Ballungszentren, Verkehrsknoten, etc.) berücksichtigt werden müssen. Außerdem wird durch die Lage der Stationen, aus geologischen Gründen oder auf Grund tiefer Fundamente der Grundriß der Trasse bestimmt, d.h. die "horizontale" Länge der Trasse ist dadurch vorgegeben. Da der Zug in jeder Station halten muß, kann das Problem der energieoptimalen Trassierung des U-Bahn-Netzes auf das Problem der energieoptimalen Trasse zwischen zwei Stationen zurückgeführt werden. Zusätzlich wird noch die Annahme getroffen, daß der Zug für beide Fahrrichtungen dieselbe Trasse durchfährt. Die Länge der Trasse wird vom Mittelpunkt der ersten Station (A) bis zum Mittelpunkt der zweiten Station (B) gemessen. Die Station kann nicht als Punkt angenommen werden, da das Verhältnis von Stationslänge und Länge der Trasse zwischen den beiden Stationen zu groß ist. Da jeweils die Hälfte der Station zur Trasse gerechnet wird, ist es möglich, die Trassen einer U-Bahn-Strecke aneinanderzureihen und so den gesamten Energieverbrauch und die reine Fahrzeit zu berechnen.

Ausgehend von zwei festen Stationen mit bekannter Entfernung LT , gegebener Höhendifferenz H und beliebig vorgegebener Starttrajektorie, soll mit

Hilfe des im folgenden beschriebenen Modells und einer zu wählenden Fahrstrategie durch Trassenvariation die energieoptimale Trasse berechnet werden.

Der Energieverbrauch hängt wesentlich von der Fahrweise der Züge ab. Eine für eine U-Bahn übliche Forderung ist die nach möglichst kurzen Fahrzeiten. Es wird daher angenommen, daß die Fahrstrategie des Zuges stets auf dieses Ziel ausgerichtet ist, also durch die zeitoptimale Steuerung bestimmt ist.

Die Trasse soll durch einen Polygonzug mit n Stützpunkten approximiert werden. Die Fahrt von B nach A wird mit denselben Gleichungen berechnet, und daher muß für die Rückfahrt die Trasse gespiegelt werden.

3. MODELL

Aus Gesetzen der Mechanik lassen sich folgende Gleichungen ableiten:

$$\frac{dv}{dx} = \frac{1}{v} [u^+ - u^- + g(x, v, \theta)]$$

$$\frac{dE}{dx} = \frac{1}{v} [\beta v u^+ - \alpha \beta v u^-]$$

$$\frac{dt}{dx} = \frac{1}{v}$$

$$g(x, v, \theta) = -\mu \int_{x-\frac{LT}{2}}^{x+\frac{LT}{2}} \sin\theta(s) ds - v \int_{x-\frac{LT}{2}}^{x+\frac{LT}{2}} \cos\theta(s) ds - av^2$$

wobei die Variablen folgende Bedeutung haben:

Zustandsvariable:

- $v(x)$ Geschwindigkeit des Zuges, gemessen in m/s
- $E(x)$ Energie, die vom Zug verbraucht wird, gemessen in $kg \cdot m^2/s^2$
- $t(x)$ Fahrzeit des Zuges, gemessen in s
- $\theta(x)$ Neigungswinkel der Trasse

Steuervariable:

- $u^+(x)$ Antriebskraft des Zuges, gemessen in m/s^2
- $u^-(x)$ Bremskraft des Zuges, gemessen in m/s^2

Parameter:

LT	Zuglänge, gemessen in m
β	Gewicht des Zuges, gemessen in kg
α	prozentueller Anteil der rückzuführenden Energie beim Bremsen ($0 \leq \alpha < 1$)
μ	Faktor für die Gravitation, $[\mu] = 1/s^2$ mit $\mu = 9.81/LT$
ν	Faktor für die Reibung, $[\nu] = 1/s^2$ mit $\nu = \tau$, wobei τ der mittlere Reibungskoeffizient für Schienenfahrzeuge ist ($\tau = 0.0035$)
a	Faktor für den Luftwiderstand, $[a] = 1/m$ mit $a = 3.66/\beta$

Als zugrundeliegendes Maßsystem wird das MKS-System verwendet:

Länge	Meter
Zeit	Sekunden
Masse	Kilogramm
Kraft	Newton
Arbeit	Joule
Leistung	Watt

Da für x die Integrationsgrenzen bekannt sind, nämlich $[A, B]$, für t hingegen die obere Integrationsgrenze unbekannt ist, wird der Einfachheit halber nicht die Zeit t , sondern der momentane Abstand x des Zuges von der Station als unabhängige Variable gewählt.

Die Funktion $g(x, v, \theta)$ berücksichtigt den Einfluß von Gravitation, Reibung und Luftwiderstand auf die Geschwindigkeit. Bei negativem Neigungswinkel (fallende Trasse) wirkt die Gravitation als Antriebskraft, bei positivem Neigungswinkel als Bremskraft. Der Neigungswinkel der Trasse wird jeweils in Fahrtrichtung gemessen. Reibung und Luftwiderstand wirken immer als Bremskraft. Da der Zug nicht als Punktmasse angenommen werden kann (Verhältnis von Zuglänge und Trassenlänge zu groß), müssen Gravitation und Reibung mit Hilfe des Integrals über die Zuglänge gemittelt werden. Klarerweise ergeben sich aus mechanischen Gründen und aus Gründen der Fahrtechnik und Sicherheit sehr viele Beschränkungen für die einzelnen Größen des Modells wie maximale Geschwindigkeit, maximaler Neigungswinkel und Einschränkungen für die Antriebs- und Bremskraft.

Randbedingungen

Wie man aus den Differentialgleichungen leicht erkennen kann, beschreiben sie die Wirklichkeit nur für $v \neq 0$, da das Problem für $v=0$ singular wird (die rechte Seite der Differentialgleichung wird unendlich). Deshalb muß die Annahme getroffen werden, daß der Zug in den Stationen eine gewisse Mindestgeschwindigkeit hat, die möglichst klein angesetzt werden soll.

Randbedingungen für die Fahrt von A nach B:
 $y(A) = Y_A$ $\theta(A) = \theta_A$ $v(A) = v_{\min}$ $t(A) = 0$

$y(B) = Y_B$ $\theta(B) = \theta_B$ $v(B) = v_{\min}$ $E(A) = 0$

Für die Fahrt von B nach A ändern sich nur die Randbedingungen für t und E :

$t(B) = 0$ $E(B) = 0$

sowie der Term θ in der Differentialgleichung, da ja die Trasse zu spiegeln ist.

Gütekriterium

Das Gütekriterium erhält man aus der Summe des Energieverbrauchs für beide Fahrtrichtungen:

1. Richtung: $E(B) = E_B$

2. Richtung: $E(A) = E_A$

$$J = E_A + E_B \quad \text{! Min}$$

4. FAHRSTRATEGIE

Der Energieverbrauch, der minimiert werden soll, hängt sehr stark von der Fahrweise des Zuges ab. Deshalb ist es sehr wichtig, eine einheitliche Strategie in der Fahrweise für alle Trassen fest zu wählen, da sonst kein Vergleich der verbrauchten Energie möglich wäre.

In der Problemstellung wurde gefordert, die Züge zeitoptimal zu steuern, das bedeutet, daß die Geschwindigkeit möglichst rasch den Maximalwert erreichen soll, der dann bis zu jenem Zeitpunkt gehalten wird, ab dem mit maximaler Bremskraft gebremst werden muß, sodaß der Zug in der Station v_{\min} erreicht. Daraus folgt für die Steuerungen u^+ und u^- , daß sie keine stetigen Funktionen sind, sondern nur die zwei Werte 0 und u^+_{\max} bzw. 0 und u^-_{\max} annehmen können.

Begonnen wird mit maximaler Beschleunigung ($u^+ = u^+_{\max}$, $u^- = 0$). Wird die Höchstgeschwindigkeit überschritten, so muß gebremst werden ($u^+ = 0$, $u^- = u^-_{\max}$). Wird die Geschwindigkeit kleiner als die Maximalgeschwindigkeit, so muß beschleunigt werden ($u^+ = u^+_{\max}$, $u^- = 0$). Um ein Triggern der Steuergrößen zu verhindern, wurde obige Strategie dahingehend abgeändert, daß in einem gewissen Toleranzintervall $[v_0, v_{\max}]$ weder beschleunigt noch gebremst wird. Außerdem werden die Steuersignale für das Beschleunigen und das Bremsen um eine bestimmte Zeit Δt verzögert (durch monostabile Kippstufen realisiert).

5. VORBEREITUNG ZUR ANALOGSCHALTUNG

Bei einem hybriden Problem stellt sich die Frage, was sinnvollerweise analog und was digital berechnet werden soll. Da die Differentialgleichungen sehr oft gelöst werden müssen, kommt für deren Berechnung nur der Analogrechner in Frage. (Lösen des Randwertproblems in ca. 15 msec). Die Steuerung des Analogrechners und das Optimierungsverfahren werden am Digitalrechner realisiert. Die Integrale für den Einfluß der Gravitation und der Reibung müssen zu jedem Zeitpunkt ausgewertet werden. Durch Umformung der Integrale kann die Berechnung rein digital erfolgen und durch Verwendung von DCFGs kann die Auswertung der Integrale rein analog durchgeführt werden, wodurch die Werte kontinuierlich am Analogrechner zur Verfügung stehen und den Schwierigkeiten der Datenübertragung ausgewichen wird. Da die Trasse durch einen Polygonzug mit n Stützpunkten approximiert wird, ist der Neigungswinkel θ_i zwischen zwei Stützpunkten, also im Intervall $[x_i, x_{i+1}]$, konstant. Das bedeutet, daß die Berechnung des Integrals

$$\int_{x_0}^{x_m} (\mu \sin \theta(s) + v \cos \theta(s)) ds =$$

$$\sum_{i=0}^{m-1} (\mu \sin \theta(s) + v \cos \theta(s)) (x_{i+1} - x_i)$$

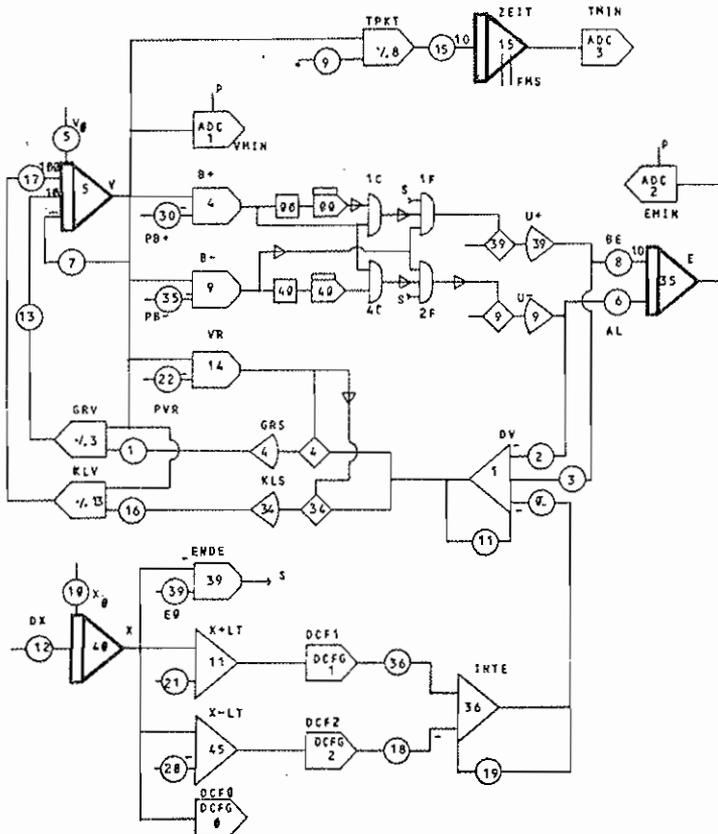
digital sehr einfach ist. Weiters gilt, daß der Wert des Integrals

$$\int_{x_0}^x (\mu \sin \theta(s) + v \cos \theta(s)) ds$$

zu einem Punkt $x \in [x_i, x_{i+1}]$ die Linearkombination der Werte der Integrale zum Punkt x_i und zum Punkt x_{i+1} ist. Speichert man nun in ein Feld für jeden Stützpunkt den digital berechneten Wert des Integrals obiger Form, transformiert diese in DCFG-Format und setzt zwei DCFGs auf diese Werte, so kann man direkt zu jedem Punkt x den Wert des gewünschten Integrals am Analogrechner auswerten.

$$\int_{x - \frac{LT}{2}}^{x + \frac{LT}{2}} f(s) ds = \underbrace{\int_{x_0}^{x + \frac{LT}{2}} f(s) ds}_{\text{DCFG 1}} - \underbrace{\int_{x_0}^{x - \frac{LT}{2}} f(s) ds}_{\text{DCFG 2}}$$

6. ANALOGSCHALTUNG



Um genauere Resultate zu erhalten, wurden für die Division durch v zwei Dividierer verwendet, die verschieden skaliert sind (1. Dividierer für den Bereich, in dem v sehr klein ist, 2. Dividierer für das Restintervall). Durch die Verwendung von 4-Quadranten-DCAs konnten in der Schaltung Inverter vermieden werden. Mit ADC1, ADC2 und ADC3 werden Geschwindigkeit, Energie und Zeit zum Digitalrechner übertragen. Die Steuerung der Integrierer und die Übertragungssignale für die ADCs werden vom 4-Intervall-Repop-Timer geliefert.

7. VERFAHREN ZUR TRASSEN-VARIATION

Die Trassenvariation wurde einerseits mit einem Verfahren [1], das Ideen aus dem Verfahren des adaptiven Suchschlauches [3] verwendet, andererseits mit einem Verfahren [5], das von der zyklischen Parametervariation hergeleitet wurde, durchgeführt.

Beim ersten Verfahren wurde die Trasse durch einen Polygonzug mit äquidistanten Stützstellen approximiert. Die zu variierenden Parameter sind die y -Koordinaten der Stützpunkte. Beim zweiten Verfahren wurden als zu variierende Parameter sowohl die x -Koordinaten als auch die y -Koordinaten der Stützpunkte gewählt.

8. REALISIERUNG AM HYBRIDRECHNER

Die Kontrolle und Auswertung des analogen Modells wurde von der an der Hybridrechenanlage vorhandenen Unterstützungssoftware für hybride Simulation (HYBSYS [4]) durchgeführt. Durch die Verwendung von HYBSYS wurde das Digitalprogramm in einzelne Moduln (z.B. Initialisierung, Änderung der Parameter des Modells, Verfahren 1, Verfahren 2, Ausgabe,...) zerlegt, die zu einem beliebigen Zeitpunkt in beliebiger Reihenfolge exekutiert werden können.

9. ERGEBNISSE

Für beliebig vorgegebene Trassen wurde die Optimierung mit beiden Verfahren durchgeführt, wobei auch die Einflüsse von Gravitation, Reibung, Luftwiderstand, Länge des Zuges, Gewicht des Zuges, Länge der Trasse und prozentuellem Anteil der rückzuführenden Energie überprüft wurden.

An Hand einer vorgegebenen Trasse sollen die am Analogrechner berechneten Funktionen für die Geschwindigkeit, Beschleunigung, u^+ und u^- gezeigt werden.

Bei der Abbildung für die Trasse (Abb. 1) ist das Verhältnis des Maßstabes von x zu dem von y zu beachten, wodurch die Neigungswinkel verzerrt werden und größer als möglich zu sein scheinen.

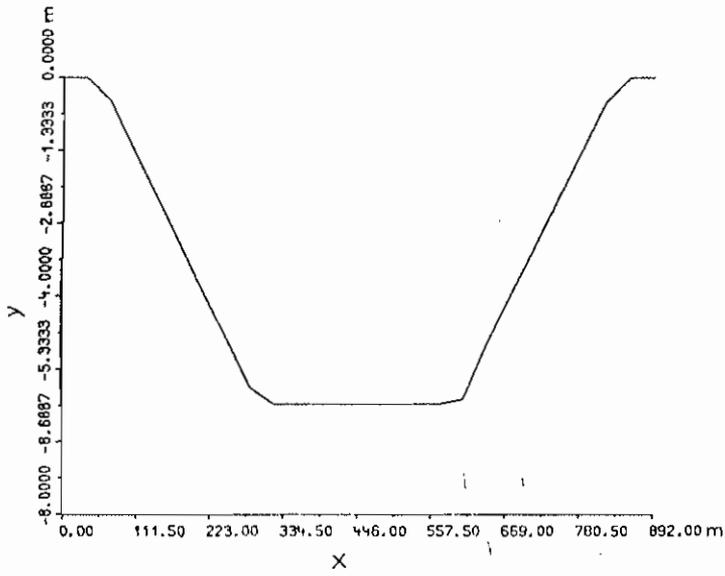


Abbildung 1

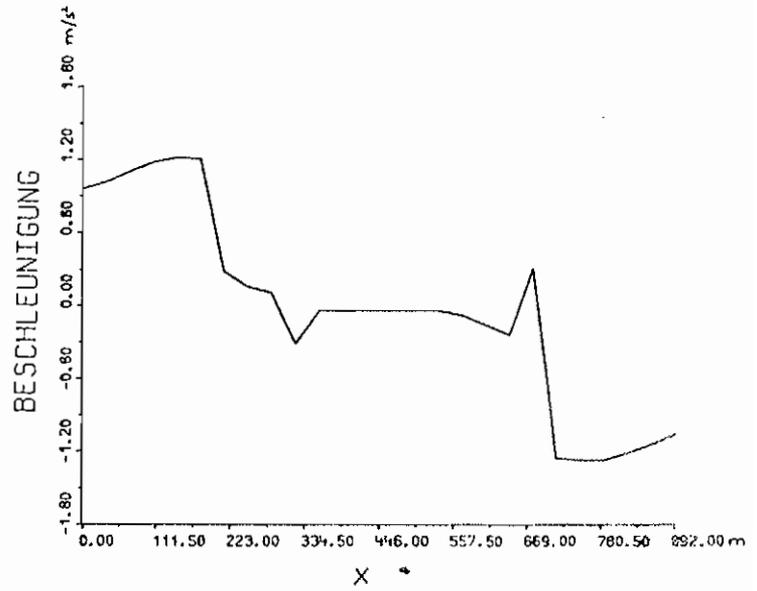


Abbildung 3

Die gewählte Fahrstrategie - möglichst rasch die maximale Geschwindigkeit zu erreichen und diese möglichst lange zu halten - kann in den Abbildungen von v (Abb.2), u^+ (Abb.4) und u^- (Abb.5) sehr gut erkannt werden. Die maximale Beschleunigung, die erreicht wird, ist 1.2 m/sec^2 .

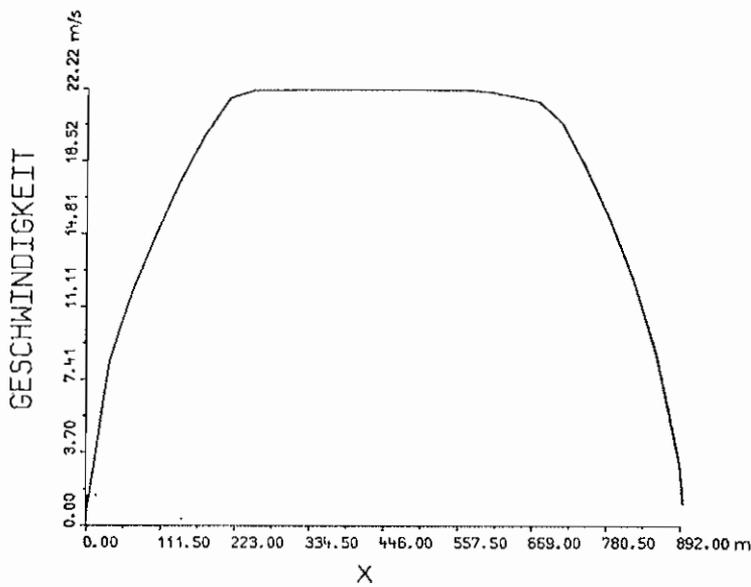


Abbildung 2

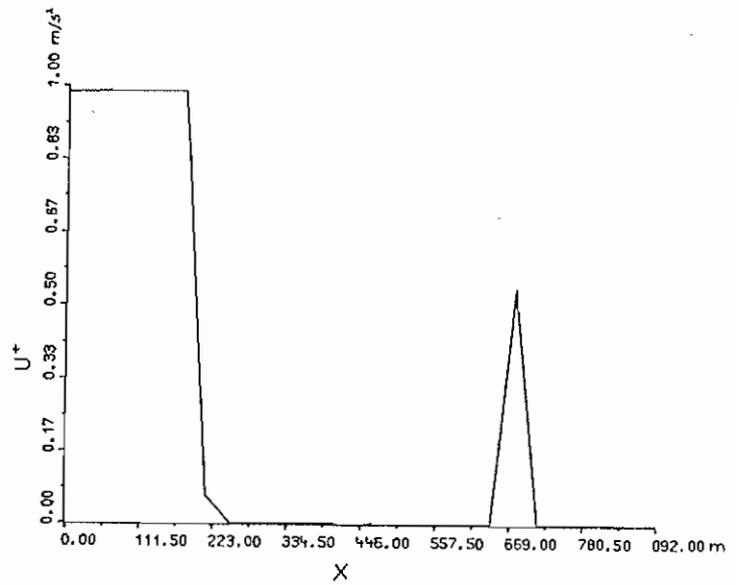


Abbildung 4

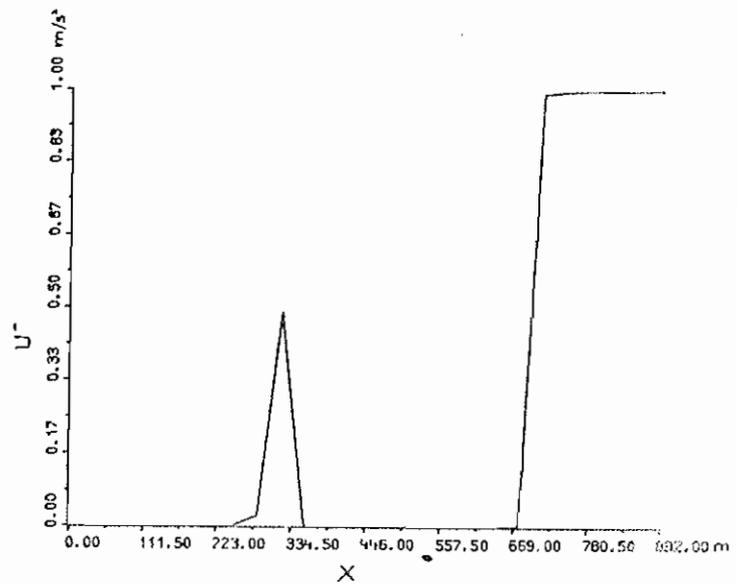


Abbildung 5

Der prozentuelle "Energiegewinn" hängt natürlich sehr stark vom Stationsabstand und dem Stationshöhenunterschied ab. Bei kurzen Abständen und keinem Höhenunterschied zwischen den Stationen kann natürlich am meisten Energie gespart werden. Bei größerem Höhenunterschied unterscheidet sich die optimale Trasse immer weniger von der geradlinigen Verbindung.

Der Energieverbrauch für die direkte Verbindungs-trasse zwischen zwei Stationen, die auf gleicher Höhe liegen und 892 m voneinander entfernt sind, beträgt 20.45 kwh bei einer Fahrzeit von 123.51 s. Bei der optimierten Trasse (Abb. 6) verringert sich der Energieverbrauch um 22% und die Fahrzeit um 4.3%. Hingegen verringert sich der Energieverbrauch bei der Trasse, bei der die Stationen einen Höhenunterschied von 25 m aufweisen, nur um 6.8% und die Fahrzeit um 3.8% (Abb. 7).

Wie man an Hand der Abbildungen 8 und 9 leicht erkennen kann, hat eine Änderung der Zuglänge, des Gewichtes des Zuges oder des prozentuellen Anteils (α) der rückzuführenden Energie keinen sehr großen Einfluß auf die Gestalt der optimalen Trasse.

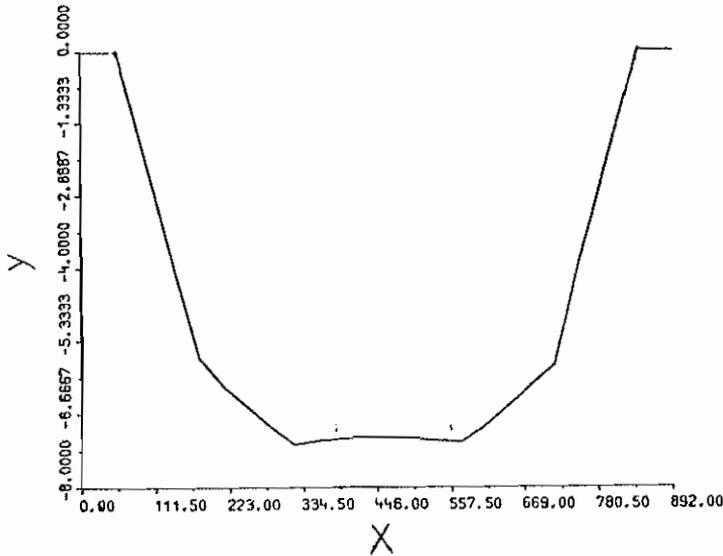
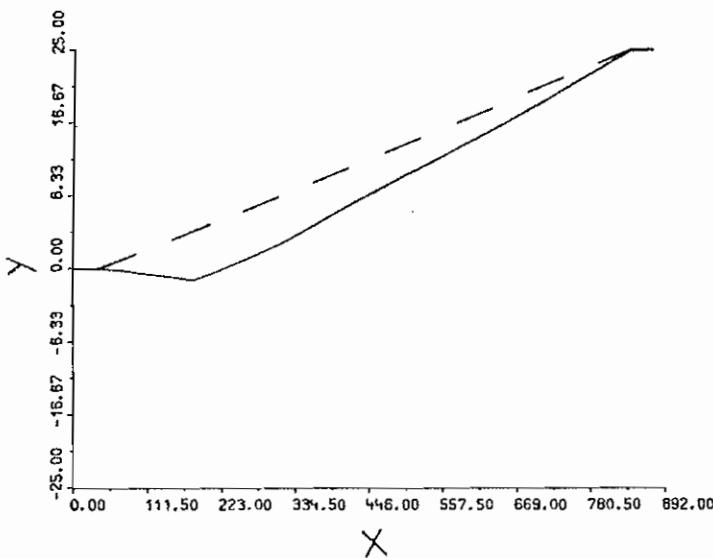


Abbildung 6



--- Ausgangstrasse: Energieverbrauch: 27.54 kwh
Zeitbedarf: 125.71 s
— optimierte Trasse: Energieverbrauch: 25.67 kwh
Zeitbedarf: 121.03 s

Abbildung 7

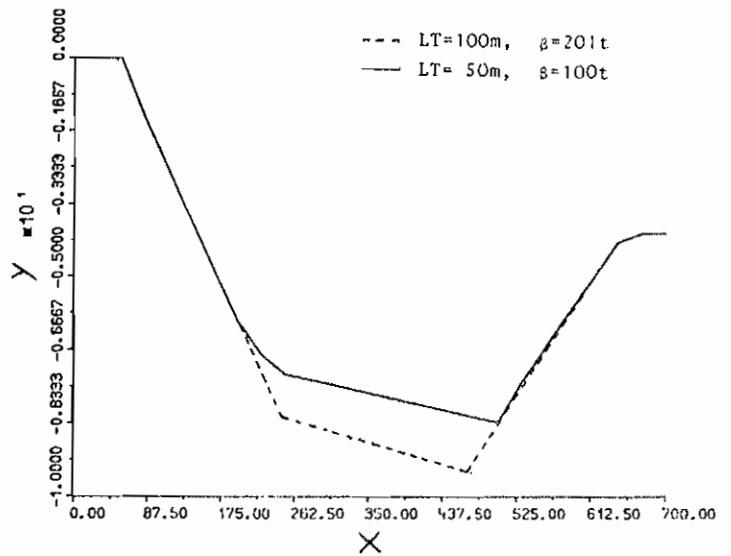


Abbildung 8

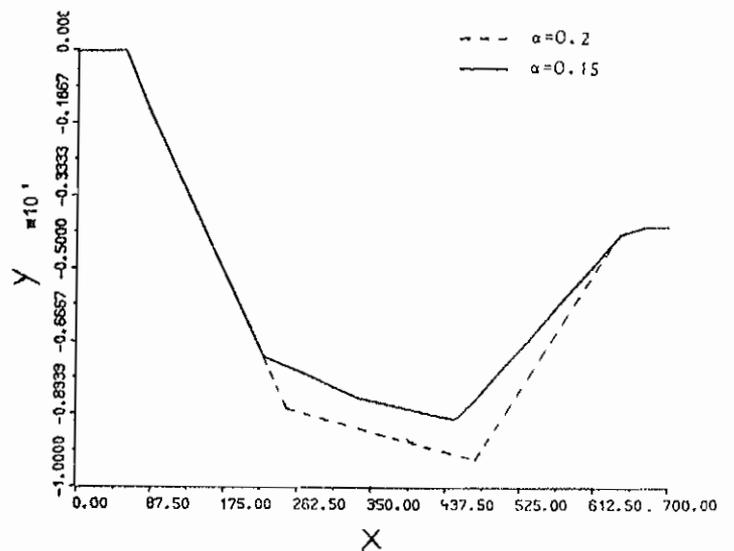


Abbildung 9

Es zeigt sich, daß die Gravitation die ausschlaggebende Größe für die Gestalt der optimalen Trasse ist. Das bedeutet für die Gestalt der Trasse, daß der erste bzw. der letzte Abschnitt des Polygonzuges den größten erlaubten Anstieg hat. (Falls der Höhenunterschied der Trasse so groß ist, daß am Ende der Trasse dieser Anstieg nicht realisiert werden kann, so wurde der größtmögliche realisiert.) Diese Abschnitte sind so lang, daß der Zug am Ende dieser Teilstücke gerade die zulässige Höchstgeschwindigkeit erreicht hat.

Im Hinblick auf die Baukosten sei darauf hingewiesen, daß bei einer Trassenlänge von 1000m und auf gleicher Höhe liegenden Stationen die optimale Trasse um weniger als 1m länger als die gerade Verbindungstrasse ist. Weiters verringert sich bei den berechneten Trassen nicht nur der Energieverbrauch, sondern auch die Fahrzeit. In [2] wird die Richtigkeit dieser Rechnung insofern bestätigt, als mit den dort durch heuristische Überlegungen erhaltenen (qualitativ identischen) Resultaten beim U-Bahn-Netz in Montreal der Energiebedarf um durchschnittlich 16% gegenüber der geradlinigen Verbindung gesenkt werden konnte.

Summary

In this article a model is proposed which enables the simulation of a train in a Metro Network. This model is described by three differential equations (for velocity, energy and time) and is simulated on the analog computer with the support of the hybrid processor HYBSYS. The aim of this work is to reduce the energy consumption through optimization of the tunnel trajectories. The optimization was realized by two different methods.

The main result is the fact that an optimal trajectory should have maximum slope at both ends because the most important parameter for the reduction of energy by variation of tunnel trajectory is this which brings the influence of gravity into equation for velocity.

Finally it shall be pointed out that only the use of a hybrid computer allows to perform the search for the optimal trajectory within reasonable computation time. This is due to the large number of integrations which can be performed on the analog computer more quickly than on a digital computing system.

LITERATUR

- [1] BERGER F.: Energieoptimale Trassierung eines U-Bahn-Netzes. Diplomarbeit, Institut für Technische Mathematik, TU Wien, 1979.
- [2] HOANG H. H., POLIS M. P., HAURIE A.: Reducing Energy Consumption through Trajectory Optimization for a Metro Network. IEEE Transaction on Automatic Control, Vol. AC-20, No 5, Oct 1975.
- [3] SCHULZE H. K.: Die Methode des adaptiven Suchschlauchs zur Lösung von Variationsproblemen mit Dynamic Programming-Verfahren. Elektronische Datenverarbeitung, 3/1966.
- [4] SOLAR D.: HYBSYS - interaktiver hybrider Prozessor, Version 3.2. Hybridrechenanlage TU Wien, 1979
- [5] WITTEK E.: Energieoptimale Trassierung eines U-Bahn-Netzes. Diplomarbeit, Institut für Technische Mathematik, TU Wien, 1979.

RECHNERERZEUGTE PUNKTBILDER AUF X/Y-OSZILLOGRAPH - ANGEWANDT FÜR DAS LIFE-SPIEL *)

Herbert Wehlan
Institut für Systemdynamik und Regelungstechnik
Universität Stuttgart

Einleitung

Man kann mit einem Hybridrechner PACER 600 stehende oder bewegte Punkt-Bilder, bestehend aus bis zu 4096 Punkten in einem 64 x 64 Raster, auf einem gewöhnlichen X/Y-Oszillographen sehr einfach erzeugen, wenn das Hybrid-Interface und damit der D/A-Wandler über einen DMAC (Direkter Speicherzugriff) ansteuerbar sind. Ein Programm dafür wird hier beschrieben.

Mit dem bekannten "Life"-Spiel von Conway wurden von "Computer-Cracks" bisher weltweit sicherlich viele tausend Stunden Rechenzeit verbraten. Das Spiel ist für die o.e. graphische Darstellungsart ein geeignetes Demonstrationsbeispiel mit nur halb-ernstem Hintergrund: Simulation einer Populationsdynamik.

In der hier beschriebenen Version des Spiels wird eine Verallgemeinerung der Spielregeln ermöglicht, die zu interessanten Bildern führt.

1. Punkt-Bilder auf dem X/Y-Oszillographen

1.1 Prinzip

Die Koordinaten der darzustellenden Punkte sind in einem Speicherbereich ("DIS-Liste") beliebiger Länge $\leq 8K$ in beliebiger Reihenfolge folgendermaßen anzugeben: X- und Y-Wert jeweils in aufeinanderfolgenden Speicherworten, als 6-bit-Werte linksbündig ohne Vorzeichen, d.h. in Bit 1 bis 6.

Mit dem Programmteil INIDIS werden zwei benachbarte DACs auf Transfer-Mode initialisiert, Anfangs- und Endadresse der DIS-Liste werden in ein Record-Packet geschrieben und der DMAC gestartet.

Legt man nun an die DMAC-Buchse auf dem Patchpanel des 680 einen Puls mit der Frequenz $10^4 \dots 10^5$ Hz und einer Pulsdauer, die genau 2 DMA-Zyklen auslöst, so erscheinen mit dieser Frequenz an den beiden DACs die Koordinaten der darzustellenden Punkte als Spannungen im Bereich 0...-10V. Die beiden DACs werden an die X- und Y-Eingänge des Oszillographen angeschlossen. Dunkeltastung zwischen den Punkten ist wegen der schnellen Settling-Time der DACs nicht nötig. Der erforderliche Puls kann mit dem 680 erzeugt werden; siehe 1.4. (Bem.: Mit dem Sichtgerät des 680 läßt sich im CROSS-PLDT kein flackerfreies Bild erhalten, daher ist ein Oszillograph besser.)

*) Dieser Beitrag wurde der Redaktion von einem Benutzer eines Hybridrechners PACER 600 an der Universität Stuttgart freundlicherweise zur Verfügung gestellt. Listing und Binärlochstreifen des beschriebenen Programms sind beim Autor erhältlich.

Ist die DIS-Liste vom DMAC abgearbeitet, so löst der DMAC-Interrupt die Interruptserviceroutine DISP aus, welche den DMAC neu initialisiert und startet.

Da die Bildausgabe über den DMAC geschieht, kann bei laufender Ausgabe die DIS-Liste per Programm verändert und somit ein bewegtes Bild erzeugt werden.

1.2 Datenbeschreibung

Für die meisten Anwendungen ist es sinnvoll, mit 3 verschiedenen Listen zu arbeiten (siehe Bild 1):

- DIS-Liste: für den Oszillographen (Display); siehe 1.1. Zwei Speicherworte entsprechen einem vorhandenen Bildpunkt.
- RE-Liste: zum Berechnen des nächsten Bildes. Jedem Rasterelement entspricht ein Speicherwort. Negatives Vorzeichen bedeutet: Punkt vorhanden. Die restlichen 15 Bit stehen zur Berechnung des nächsten Bildes zur Verfügung.
- IO-Liste: zur benutzerfreundlichen Ein/Ausgabe. Ein Speicherwort entspricht einem vorhandenen Bildpunkt.

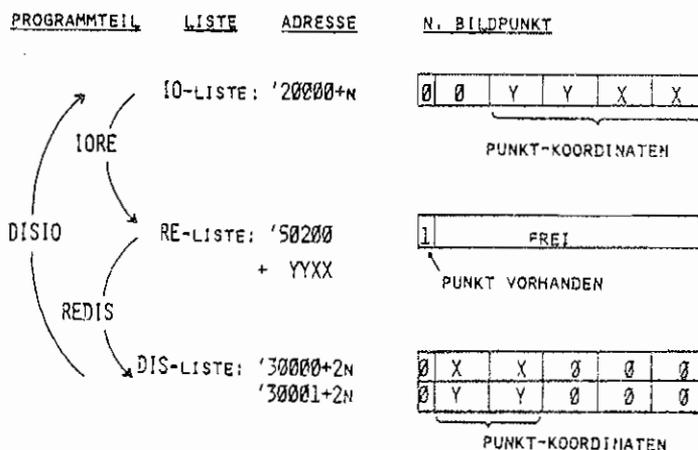


Bild 1
Daten-Listen und Listen-Umwandlung

Zur Umwandlung und Übertragung der einzelnen Listen ineinander dienen die Programmteile IORE, REDIS und DISIO; siehe Bild 1.

Die Verwendung separater RE- und DIS-Listen ermöglicht einen (fast) schlagartigen Bildwechsel, sobald REDIS aufgerufen wird.

1.3 Programm-Beschreibung

Aus Zeitgründen wurde das Programm in ASSEMBLER geschrieben. Ein Flußdiagramm zeigt Bild 2. Das Programm arbeitet mit OEDIPUS zusammen, zur Manipulation der ID-Liste. Die Programmlänge beträgt etwa 500 Speicherworte.

1.3.1 Senseswitch-Abfragen

Mit B kann SETUP übersprungen werden; siehe 1.3.2.

Mit C kann der Bildwechsel angehalten werden. Mit D erfolgt ein Rücksprung zum DED. Mit E,F,G,H wird die Bildwechsel-Geschwindigkeit verzögert; siehe 1.3.2 (TIME).

1.3.2 Bedeutung der einzelnen Programmteile

Nach dem Programmstart erfolgt, je nach Senseswitch B eventuell über den Umweg SETUP, ein Sprung zum "Supervisor"-Hauptprogramm SUP. SUP ruft dann die einzelnen Programmteile auf, gemäß dem Flußdiagramm Bild 2.

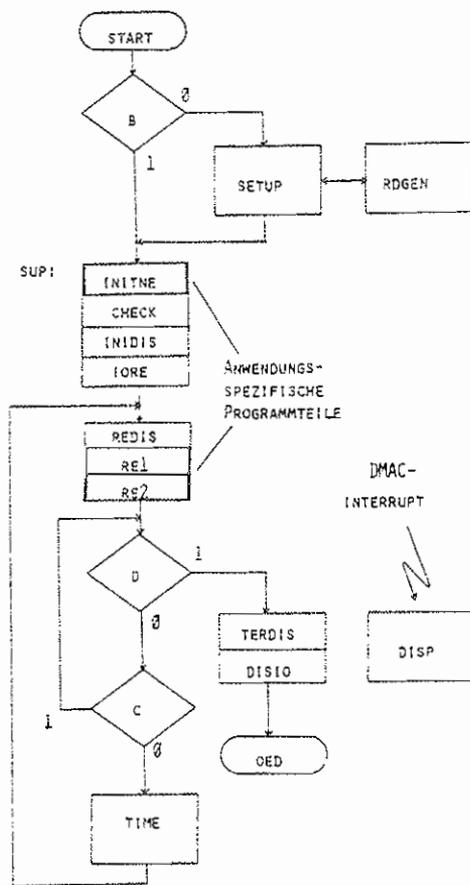


Bild 2
Flußdiagramm

INITNE, RE1 und RE2 sind anwendungsspezifische Programmteile. In INITNE werden irgendwelche Anfangsberechnungen durchgeführt. RE1 berechnet aus den Vorzeichen der RE-Listenelemente irgendwelche Informationen und speichert diese in den restlichen 15 Bit ab. Daraus errechnet RE2 die neuen Vorzeichen und damit das neue Bild.

Mit SETUP kann beim Kaltstart des Programms der Oszillograph eingestellt werden. Es werden zwei diagonale Eckpunkte des Darstellungsfeldes im schnellen Wechsel ausgegeben, sodaß eine Linie entsteht. SETUP ruft außerdem RDGEN auf, wodurch die ID-Liste per Zufallsgenerator gefüllt wird; dies kann für viele Anwendungen entfallen.

CHECK begrenzt die Elemente der ID-Liste auf 12 Bit und eliminiert die doppelt vorhandenen.

INIDIS initialisiert den DMAC und die DACs.

IORE, REDIS und DISIO dienen zur Listen-Umwandlung und -Übertragung; siehe 1.2. REDIS geht zurück zum DED, falls das Bild leer ist.

TIME verzögert den Bildwechsel, wobei eine In-Core-Timer-Zelle mit einem 100Hz-Takt benutzt wird. (Man könnte statt des In-Core-Timers auch eine NOP-Schleife verwenden, die 10msec dauert.) Senseswitch H ergibt 0,1 sec Verzögerung; E, F und G verlängern die Verzögerung um den Faktor 2^n , wobei n als Binärzahl an E, F, G eingestellt wird. Die Maximalverzögerung beträgt also 12,8 sec.

DISP ist die Interruptserviceroutine. Sie schreibt die aktuelle Endadresse der DIS-Liste in das Record-Packet und startet den DMAC erneut.

TERDIS fängt zur Beendigung der Bildausgabe den nächsten DMAC-Interrupt ab, stoppt den DMAC und schickt den Oszillographen-Strahl mit +10V/+10V nach außerhalb des Bildschirms, um ein Einbrennen zu vermeiden.

1.4 Programmlauf

1) Erzeugung des Trigger-Pulses für den DMAC mit dem Analogrechner 680: 10^4 Hz (10^3 bei komplexen Bildern) auf ein Mono-Diff; dessen Ausgang auf die DMC-Buchse. Die Zeit des Mono-Diff auf Null stellen, das Einstellrad ganz nach oben, anschließend 10 Zähne nach unten. CLOCK RATE CONTROL auf 10^6 und DIGITAL MODE CONTROL auf R stellen.

2) Sollen zur Zeitverzögerung des Bildwechsels mit TIME die Senseswitches E,F,G,H benutzt werden, so muß der In-Core-Timer '746 mit 100Hz laufen; siehe 1.3.2.

3) Programm und DED laden, DED starten.

4) In '17776 und '17777 die Anfangs- und Endadresse der ID-Liste schreiben ('20000 und >'20000).

5) Bei Bedarf: Ab '20000 das Anfangsbild in die IO-Liste schreiben. (vgl. Bild 3e). In diesem Fall muß in SETUP der Aufruf von RDGEN durch einen NOP ersetzt werden.

6) Programm starten. SETUP erzeugt nun eine Diagonale zur Oszillographen-Einstellung. Mit Senseswitch B wird danach das eigentliche Programm gestartet. Für Programm-Restarts bleibt Senseswitch B oben. Bei komplexen Bildern dauert es nun bis zu 10 sec bis das erste Bild erscheint; in dieser Zeit prüft CHECK die IO-Liste auf Doppelbelegungen.

7) Der Bildwechsel erfolgt mit einer Frequenz von ca. 3 Bildern/sec bei der hier geschilderten Anwendung. Er kann mit Senseswitch C angehalten werden oder mit E,F,G,H verzögert werden.

8) Programmabbruch und Rücksprung zum OED erfolgen durch kurzzeitiges Einschalten des Senseswitch O.

2. Das Life-Spiel

2.1 Definition

Zur primitiven Simulation einer Populations-Dynamik werden Individuen dargestellt durch Punkte in einem quadratischen Raster (hier: 64x64 Raster-elemente), das den Lebensraum symbolisiert. Jedes Raster-element ist am Anfang entweder mit einem Punkt belegt oder leer; diese Anfangskonfiguration soll als erste Generation bezeichnet werden.

Der Generationswechsel erfolgt nun zu diskreten Zeitpunkten, wobei sich das neue Bild jeweils aus dem alten nach folgenden Spielregeln errechnet:

Jedes Raster-element hat 8 benachbarte Elemente.

1) Überlebenskriterium: Ein Punkt überlebt nur dann, wenn er vor dem Generationswechsel genau 2 oder 3 Nachbarn hatte; andernfalls stirbt er (z.B. wegen Vereinsamung oder Überbevölkerung).

2) Geburtskriterium: In einem leeren Raster-element entsteht ein Punkt, wenn das leere Element vorher genau 3 belebte Nachbarn hatte (was z.B. als optimales Wachstumsklima gedeutet werden könnte).

Der Spieler legt also lediglich die Anfangskonfiguration fest und beobachtet, wie sich die Population entwickelt (z.B. Bild 3a).

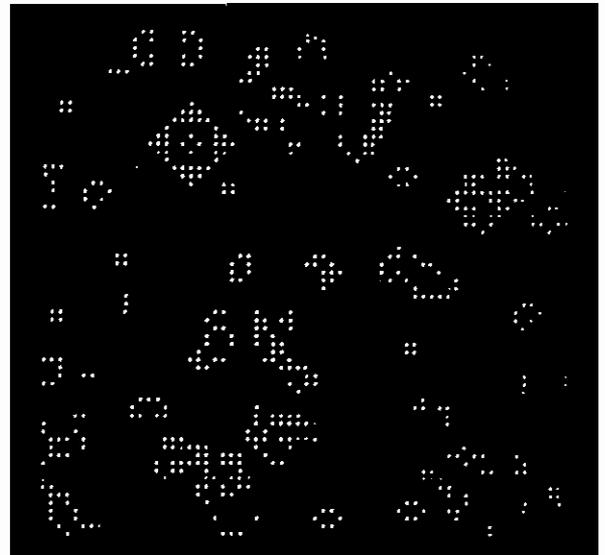
2.2 Ergebnisse

Nach wenigen oder sehr vielen Generationen ergibt sich einer der folgenden Endzustände:

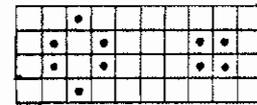
- A) Die Population stirbt aus.
- B) Es ergibt sich ein stehendes Bild (z.B. Bild 3b).
- C) Das Bild ändert sich weiterhin, wiederholt sich aber jeweils nach n Generationen (z.B. Bild 3c: sog. "Blinker" mit n=2), wobei n recht groß sein kann.

Man kann das Spiel so programmieren, daß das Rasterfeld an den waagrechten und/oder senkrechten Kanten zyklisch geschlossen zu denken ist, sodaß topologisch ein Torus oder ein Zylinder entsteht.

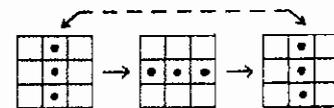
Es gibt Figuren, die sich nach n Generationen wiederholen, jedoch um ein Raster-element seitlich oder diagonal versetzt (z.B. Bild 3d); sie wandern also über das Bild. Bei zyklischer Geschlossenheit der entsprechenden Kanten bilden sie einen Sonderfall des Endzustandes C, nämlich 2 Wiederholungszyklen ineinander: einen mit Periode n und einen mit Periode 64n. Die Jagd nach dem Entdecken solcher Figuren ist ein beliebtes Betätigungsfeld der Spieler.



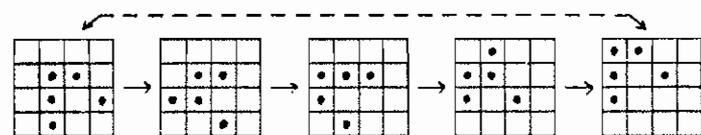
a) Wachstumsbeispiel



b) Stabile Figuren



c) Blinker (2er-Zyklus)



d) Läufer (4er-Zyklus mit Wandern)

17776: 20000	ergibt	42
17777: 20002		41
20000: 4040		40
20001: 4041		37
20002: 4140		37404142

e) Beispiel zur Bild-Eingabe mit OEDIPUS

Bild 3
Populationsbeispiele

Vor vielen Jahren, als das Spiel populär wurde, erschienen im Scientific American etliche Artikel, in denen weitere interessante Aspekte des Spiels behandelt werden.

2.3 Erweiterte Spielregeln

Die 2 Spielregeln aus 2.1 geben 2 Kriterien für das Überleben (2 oder 3 Nachbarn) und 1 Kriterium für Geburten (3 Nachbarn).

Das Programm hier ermöglicht dem Benutzer die Wahl von bis zu 3 beliebigen Kriterien je Spielregel; z.B. Überleben bei 2,3 oder 4 Nachbarn, Geburt bei 1,3 oder 5 Nachbarn.

Interessante Bilder in der Art rechteckiger Labyrinth entstehen beispielsweise mit den Kriterien 2,3,4 Nachbarn für Überleben und 3 Nachbarn für Geburten. Bild 4b zeigt einen Endzustand und 4a ein Zwischenstadium, bei dem man chaotisches Wachstum am Rand und beginnende Strukturbildung im "Hinterland" erkennen kann.

2.4 Programmbeschreibung (zusätzlich zu 1.3)

INITNE berechnet einige Konstanten; RE1 benutzt diese zur Adressierung der 8 Nachbarn eines Punktes.

RE1 zählt in der RE-Liste für jedes Element diejenigen der 8 Nachbarn, die negatives Vorzeichen haben, und trägt diese Anzahl rechtsbündig in jedes Element ein. RE1 ist zur Rechenzeiterparnis so programmiert, daß die senkrechten Kanten zyklisch geschlossen sind (vgl. 2.2), jedoch um eine Bildzeile versetzt.

RE2 korrigiert die Vorzeichen anhand dieser Anzahl und anhand der je 3 Kriterien für Überleben und Geburt.

2.5 Programmablauf (zusätzlich zu 1.4)

Es gibt folgende Möglichkeiten, die Anfangskonfiguration in die ID-Liste einzutragen:

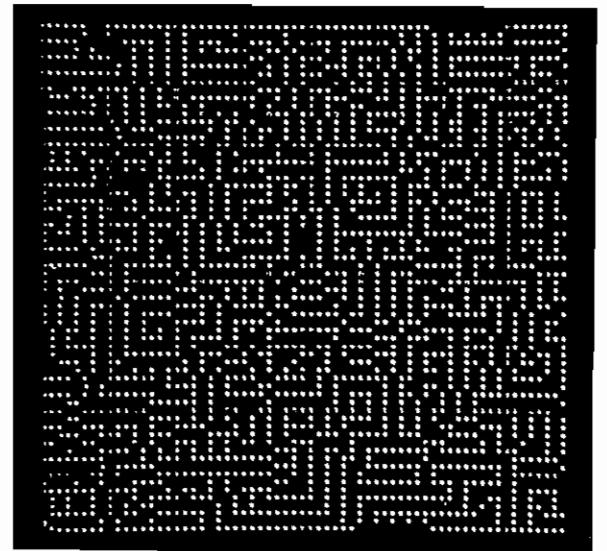
- a) Man tippt sie mit DED ein (für kleine Bilder)
- b) Man erstellt einen Lochstreifen und liest diesen mit OED ein (für interessante Bilder).
- c) Man erstellt und lädt einen Plattenfile (wie b).
- d) Man benutzt den Zufallsgenerator ROGEN über SETUP und begrenzt die Endadresse der ID-Liste (in '17777) auf einen Wert, der die gewünschte Populationsdichte ergibt.
- e) Man nimmt das, was gerade zufällig im Speicher steht (oder lädt sich vorher irgendein Programm von der Platte, das im ID-Liste-Bereich liegt) und gibt nur die Anfangs- und Endadresse der ID-Liste an.

Mit DED können im Speicher die Kriterien CR01,2,3 für Überleben und CRN1,2,3 für Geburt geändert werden; vgl. 2.3.

Der Experimentierfreude des Benutzers sind keine Grenzen gesetzt.



a) Zwischenzustand



b) Endzustand

Bild 4
Labyrinth-ähnliche Struktur

Summary

A PACER 600 Hybrid Computer can easily be used to generate fixed or moving pictures on any X/Y-oscilloscope, if the Hybrid Interface is connected to the CPU via Direct Memory Access Channel. The pictures consist of up to 4096 points in a 64 by 64 raster. Part one gives a program description.

The well known Life Game by Conway is presented in part 2 as an application example for this display method. It can be viewed as a simple simulation of population dynamics. A generalization of Conway's rules of the game is included, which leads to interesting pictures.

VERGLEICH VERSCHIEDENER KLASSIFIZIERER FÜR DIE AUTOMATISCHE ERKENNUNG VON AKUSTISCHEN SIGNALEN

R Gudonawitschius
Politechnische Hochschule in Kaunas, UdSSR
I. Husinsky
Hybridrechenanlage
Technische Universität Wien

EINFÜHRUNG

Es gibt viele Verfahren zur automatischen Analyse und Erkennung von akustischen Signalen (Sprache, Geräusche, etc.), aber es gibt sehr wenige Arbeiten, in denen diese Methoden miteinander verglichen werden. Für diesen Zweck haben wir einige lineare Klassifizierer empfohlen und zwar: die euklidische, die gewichtete euklidische und die Machalanobis-Abstandsmessung [1]. Im folgenden werden diese Klassifizierer beschrieben.

THEORETISCHE BESCHREIBUNG DER KLASSIFIZIERER

a) Euklidische Abstandsmessung (EAM)

Einer der einfachsten Klassifizierer beruht auf der Abstandsmessung mittels euklidischer Metrik. Ein zu klassifizierender Vektor von Merkmalen X wird derjenigen Klasse ω_j zugeordnet, zu deren Schwerpunkt $M^{(j)}$ er den kleinsten Abstand d_j hat, mit

$$d_j = (X - M^{(j)})^T (X - M^{(j)}) \quad (1)$$

wobei $M^{(j)} = E\{X^{(j)}\}$... Mittelwert der Merkmale X

und $X \in \omega_j$, wenn $d_j = \min_{j=1}^L \{d_j\}$ (2)

b) Gewichtete euklidische Abstandsmessung (GAM)

Man kann die Leistungsfähigkeit der EAM durch eine Gewichtung verbessern. Die Gewichtung besteht aus der Varianz σ_N^2 der Merkmale X , und die Abstandsmessung wird nach folgender Formel berechnet:

$$d_j = (X - M^{(j)})^T G (X - M^{(j)}) \quad (3)$$

wobei

$$G = \begin{pmatrix} \frac{1}{\sigma_1^2} & 0 & \dots & 0 \\ \vdots & \frac{1}{\sigma_2^2} & \ddots & \vdots \\ 0 & \dots & \frac{1}{\sigma_N^2} & \end{pmatrix}$$

mit $\sigma_N^2 = E\{(X_i^{(k)} - M_i^{(k)})^2\}$

Nur bei wenig korrelierten Merkmalen kann die Gewichtung eine Verbesserung erzielen, bei starker Korrelation bewirkt diese Gewichtung wenig.

c) Machalanobis-Abstandsmessung (MAM)

Der Machalanobis-Abstandsklassifizierer berücksichtigt im Gegensatz zu den bisher genannten Klassifizierern die Korrelation zwischen den Merkmalen X_i .

Setze man voraus, daß die Kovarianzmatrizen durch folgende Formeln berechnet werden,

$$\Sigma_i = E\{(X^{(i)} - M^{(i)})^T (X^{(i)} - M^{(i)})\},$$

dann läßt sich die Machalanobis-Abstandsmessung definieren als

$$d_j = (X - M^{(j)})^T \Sigma^{-1} (X - M^{(j)}) \quad (4)$$

und nach Gleichung (2) der Vektor X zuordnen. Bei der numerischen Berechnung des Machalanobis-Abstands nach der Formel (4) muß die Inverse der Kovarianzmatrix berechnet werden. Durch Transformation gelangt man zu einer Formel, die die Eigenwerte und Eigenvektoren der Kovarianzmatrix verwendet.

Die Eigenwerte der Kovarianzmatrix Σ seien durch

$$\Lambda = \begin{pmatrix} \lambda_1 & 0 & \dots & 0 \\ 0 & \lambda_2 & & \vdots \\ \vdots & & \ddots & \\ 0 & \dots & & \lambda_N \end{pmatrix} \quad (5)$$

und die Matrix der Eigenvektoren Φ_i durch

$$\Phi = (\Phi_1, \Phi_2, \dots, \Phi_N) \quad (6)$$

gegeben. Dann wird ein Merkmal X durch Y abgebildet

$$Y = \Lambda^{-1/2} \Phi^T X$$

mit

$$\Lambda^{-1/2} = \begin{pmatrix} \frac{1}{\sqrt{\lambda_1}} & 0 & \dots & 0 \\ 0 & \frac{1}{\sqrt{\lambda_2}} & \ddots & \vdots \\ \vdots & & \ddots & \\ 0 & \dots & & \frac{1}{\sqrt{\lambda_N}} \end{pmatrix} \quad (7)$$

und die euklidische Abstandsmessung (1) durchgeführt, d.h.

$$d_j = (Y - M^{(j)})^T (Y - M^{(j)}), \quad (8)$$

wobei

$$M^{(1)} = E\{Y^{(1)}\}$$

die Mittelwerte der transformierten Merkmale sind.

Es wurden experimentelle Untersuchungen mit den beschriebenen Klassifizierern bei denselben Testsignalen durchgeführt.

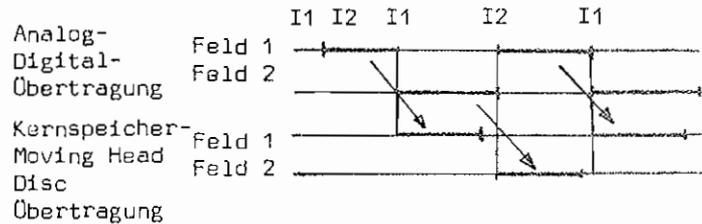
REALISIERUNG AM HYBRIDRECHNER

a) Datenübertragung

Das Phasenbildverfahren und die Bestimmung der Kreuzpunktdichten für akustische Signale am Analogrechner wurden in [2] beschrieben. Die Dichten für 4 Strahlen wurden mit einer Frequenz von 100 Hz über den direkten Datenkanal zum Digitalrechner überspielt und von dort blockweise auf Datenfiles abgespeichert. Die Testsignale kommen von einem Tonband. Der Beginn und das Ende des gesamten Übertragungsvorganges wird durch Push Buttons am Analogrechner gesteuert.

Zu Vergleichszwecken wurde auch das gesamte akustische Signal (Originalsignal) zur Abstandsmessung zum Digitalrechner übertragen. Dafür ist jedoch eine wesentlich höhere Übertragungsfrequenz notwendig (12 kHz). Der digital setzbare 4-Stufen Timer wird auf den Abstand zwischen zwei Übertragungsimpulsen gesetzt. Das differenzierte Timer-Ausgangssignal liefert dann die Übertragungsimpulse für den direkten Datenkanal (DMC-Buchse). Der Start der Übertragung wird mit einem Push Button gesteuert. Sollen nun Übertragen werden, so müssen die Daten aus dem Kernspeicher möglichst gleichzeitig mit dem Analog-Digital-Transfer auf einen Massenspeicher gebracht werden, da der Kernspeicher rasch überläuft (12 k Worte pro Sekunde). Diese gleichzeitige Datenübertragung wird durch das Vorhandensein von zwei direkten Datenkanälen am PACER 600 ermöglicht, die voneinander und von der CPU unabhängig arbeiten. Zu diesem Zweck wurde mit zwei Feldern im Kernspeicher gearbeitet (je 12 k Worte = 1 Sekunde). Ein Feld dient zur Analog-Digital-Übertragung über einen direkten Datenkanal, das andere Feld dient zur gleichzeitigen Übertragung von bereits im Kernspeicher vorhandenen Daten zur Moving Head Disc über den anderen direkten Datenkanal. Zu Beginn des Übertragungsprogramms werden zwei Analog-Digital-Transfers initialisiert (Unterprogramm DF688 der Hybrid Linkage Software). Jeder Transfer wird für eine Sekunde definiert. Dann erfolgen abwechselnd in einer Schleife die Kernspeicher-Datenfile- und die Analog-Kernspeicher-Übertragungsbefehle. Die Übertragung vom Kernspeicher auf den Datenfile muß aus Zeitgründen im Assembler mit System Traps programmiert werden, wobei der Abschluß des Transfers im Programm nicht abgewartet werden darf. Tests haben ergeben, daß auf diese Art eine Frequenz von bis zu 30 kHz gewählt werden kann. Beim Wechsel von einem Hilfsfeld auf das andere (in unserem Fall jede Sekunde) gibt es jedoch eine Übertragungslücke von ungefähr 1 Millisekunde, bedingt durch die Neuinitialisierung.

Zeitablaufdiagramm für hochfrequente Datenübertragung:



- I1 Initialisierung des A/D-Transfers für Feld 1
- I2 Initialisierung des A/D-Transfers für Feld 2

Nach derselben Methode ist auch eine hochfrequente Übertragung von Daten, die auf der Moving Head Disc gespeichert sind, zum Analogrechner möglich. So kann man z.B. ein digital gespeichertes akustisches Signal wieder hörbar machen. Das am Digital-Analog-Wandler anliegende Signal braucht dann nur, entsprechend verstärkt, über das angeschlossene Tonbandgerät auf einen Lautsprecher oder Kopfhörer gelegt werden.

b) Digitale Auswertung

Zur weiteren Verarbeitung der auf Datenfiles abgespeicherten Übertragungsdaten wurden einige Digitalprogramme geschrieben.

Die Berechnung der Merkmal-Koeffizienten (Autokorrelations-, Prädiktor- und Parcorrkoeffizienten) erfolgte wie schon in [3] beschrieben. Die Koeffizienten werden ebenfalls auf Datenfiles abgespeichert, auf die das Programm zur Abstandsmessung zugreift. Die Abstände zu verschiedenen Testsignalen wurden nach den Formeln (1), (3) und (8) programmiert. Zur Berechnung der Eigenwerte und Eigenvektoren der Kovarianzmatrix wurde das Bibliotheksprogramm EIGSYM verwendet. Dann wird der minimale Abstand gesucht und es wird eine Erkennungstabelle ausgegeben.

EXPERIMENTELLE KLASSIFIZIERUNGSERGEBNISSE

Die Untersuchungen wurden folgendermaßen durchgeführt:

Zuerst wird das zu analysierende Sprachsignal mit einer Frequenz von bis zu 10 kHz stark reduziert, d.h. mit Hilfe des Analogrechners werden die Dichten der Kreuzpunkte und die Zeitdauer im Phasenbild nach dem sogenannten Phasenbildverfahren bestimmt und weiters werden für diese Dichtefunktionen die Autokorrelations-, Prädiktor- und Parcorrkoeffizienten berechnet, die als Merkmale für die Klassifikation dienen [2,3]. Die Klassifizierungsergebnisse für 4 Sprachsignale, die als Testsignale dienen, von denen jedes 30 Analyseintervalle mit etwa 1 sec Dauer hat, sind für verschiedene Phasenbilder (x, x) , (x, δ) , (x, δ^{-1}) in Tabelle 1 zusammengefaßt [4].

Mittlere Erkennungsrate in %	K L A S S I F I Z I E R E R						Merkmale
	EAM		GAM		MAM		
Phasenbild	K	Z	K	Z	K	Z	I=38 Intervalle, N=9
(\hat{x}, x)	64.06	80.75	58.80	65.00	50.31	71.81	Autokorrelations- koeffizienten
(x, δ)	73.38	80.62	70.25	77.25	72.38	79.25	
(x, δ^{-1})	64.06	55.63	56.63	51.06	57.00	50.87	
Originalsignal	78.25		76.00		76.00		
(\hat{x}, x)	76.00	88.06	70.88	53.00	73.75	70.87	Prädiktorkoeffi- zienten
(x, δ)	76.19	80.25	71.00	75.75	56.69	77.43	
(x, δ^{-1})	68.81	62.25	56.69	55.75	51.69	53.06	
Originalsignal	59.25		28.75		20.50		
(\hat{x}, x)	76.00	84.75	62.75	64.31	69.56	72.12	Parcorkoeffizienten
(x, δ)	79.75	79.87	70.56	77.12	67.06	72.31	
(x, δ^{-1})	69.00	61.12	56.81	52.81	56.38	55.93	
Originalsignal	91.00		53.50		59.50		

Tabelle 1

K Kreuzpunktdichten
Z Zeitdauerdichten

Die Ergebnisse zeigen, daß das Phasenbildverfahren, das das zu analysierende Signal in einer zweidimensionalen Ebene betrachtet, für die Bestimmung der Merkmale und Anwendung für die Erkennung von akustischen Signalen günstig ist: erstens ist die mittlere Erkennungsrate für Phasenbildverfahren besser als für das Originalsignal, zweitens ist die Informationsreduzierung 125 mal größer und deswegen der Rechenaufwand kleiner als bei der Verarbeitung des Originalsignals.

Nach den Angaben der Tabelle kann man für die Analyse und Erkennung von akustischen Signalen die Parcorkoeffizienten für die Kreuzpunktdichte bzw. Zeitdauerdichte des Phasenbildes (\hat{x}, x) mit der euklidischen Abstandsmessung empfehlen. Es ist selbstverständlich, daß dies nur die ersten Ergebnisse im Vergleich von Verfahren von Merkmalbestimmungen und ihrer Klassifizierung nach verschiedenen Abstandsmessungen sind.

SUMMARY

Three methods for classifying random acoustic signals by using the phase path $\{\hat{x}(t), x(t)\}$ are discussed: euclidean distance, weighted euclidean distance, and Mahalanobis distance. Digital-analog transfer of the phase path data (100 Hz) and the original signal (12kHz) is described in detail. The results of the classification experiments for three types of coefficients are summarized in a table. The euclidean distance derived from the paracor coefficients of the phase path data showed the best recognition.

LITERATUR

- [1] JESORSKY P.: Ein Vergleich verschiedener Analyseverfahren der automatischen Spracherkennung. Techn. Bericht No.194, Berlin 1976
- [2] GUONAWITSCHIUS R., HUSINSKY I.: Klassifikation von akustischen Signalen durch Phasenbildverfahren. INTERFACE Nr. 13 (1979)
- [3] GUONAWITSCHIUS R., HUSINSKY I.: Ein Vergleich einiger Verfahren der Merkmalbestimmung für die Spracherkennung. INTERFACE Nr. 10 (1977)
- [4] GUONAWITSCHIUS R.: Theoretische und experimentelle Untersuchungen von akustischen Signalen. Tech. Bericht, TU Wien (1979)

INTERFACE Dezember 1979
An das Interfakultäre EDV-
Zentrum d. Techn. Univ. Wien
Abt. Hybridrechenanlage

Gußhausstr. 27-29
1040 Wien