



LUND UNIVERSITY

Programplan för ramprogrammet Datorbaserade hjälpmedel för utveckling av styrsystem (Computer Aided Control Engineering, CACE)

Mattsson, Sven Erik

1987

Document Version:
Förlagets slutgiltiga version

[Link to publication](#)

Citation for published version (APA):

Mattsson, S. E. (Red.) (1987). *Programplan för ramprogrammet Datorbaserade hjälpmedel för utveckling av styrsystem (Computer Aided Control Engineering, CACE)*. (Research Reports TFRT-3193). Department of Automatic Control, Lund Institute of Technology (LTH).

Total number of authors:
1

General rights

Unless other specific re-use rights are stated the following general rights apply:

Copyright and moral rights for the publications made accessible in the public portal are retained by the authors and/or other copyright owners and it is a condition of accessing publications that users recognise and abide by the legal requirements associated with these rights.

- Users may download and print one copy of any publication from the public portal for the purpose of private study or research.
- You may not further distribute the material or use it for any profit-making activity or commercial gain
- You may freely distribute the URL identifying the publication in the public portal

Read more about Creative commons licenses: <https://creativecommons.org/licenses/>

Take down policy

If you believe that this document breaches copyright please contact us providing details, and we will remove access to the work immediately and investigate your claim.

LUND UNIVERSITY

PO Box 117
221 00 Lund
+46 46-222 00 00

CODEN: LUTFD2/(TFRT-3193)/1-022/(1987)

Programplan för ramprogrammet

Datorbaserade hjälpmedel för utveckling av styrsystem

(Computer Aided Control Engineering, CACE)

Sven Erik Mattsson (red)

Institutionen för Reglerteknik
Lunds Tekniska Högskola
Juni 1987

Department of Automatic Control Lund Institute of Technology P.O. Box 118 S-221 00 Lund Sweden		<i>Document name</i> Final Report	
		<i>Date of issue</i> 1987-06-25	
		<i>Document Number</i> CODEN:LUTFD2/(TFRT-3193)/1-022/(1987)	
<i>Author(s)</i> Sven Erik Mattsson (red)		<i>Supervisor</i>	
		<i>Sponsoring organisation</i> Styrelsen för Teknisk Utveckling	
<i>Title and subtitle</i> Programplan för ramprogrammet Datorbaserade hjälpmedel för utveckling av styrsystem (Computer Aided Control Engineering, CACE)			
<i>Abstract</i> Denna rapport innehåller programplanen för ramprogrammet "Datorbaserade hjälpmedel för utveckling av styrsystem (Computer Aided Control Engineering, CACE)". En kort sammanfattning ges i början av programplanen. Ramprogrammet finansieras av Styrelsen för Teknisk Utveckling, STU och programplanen godkändes av STUs styrelse den 5 maj 1987.			
<i>Key words</i>			
<i>Classification system and/or index terms (if any)</i>			
<i>Supplementary bibliographical information</i>			
<i>ISSN and key title</i>			<i>ISBN</i>
<i>Language</i> Svenska	<i>Number of pages</i> 22	<i>Recipient's notes</i>	
<i>Security classification</i>			

The report may be ordered from the Department of Automatic Control or borrowed through the University Library 2, Box 1010, S-221 03 Lund, Sweden, Telex: 33248 lubbis lund.

Godkänt av STUs styrelse
den 5 maj 1987.

Programplan för ramprogrammet
Datorbaserade hjälpmedel
för utveckling av styrsystem
(Computer Aided Control Engineering, CACE)

Planen har utformats av styrgruppen:

Sven Gunnar Edlund, STFI, ordförande
Arne Otteblad, STU
Karl Eklund, Datema-Asea, Processdata AB
Claes Källström, SSPA
Eric Sandewall, LiTH
Gustaf Söderlind, ITM, KTH, LTH
Karl Johan Åström, LTH

med

Sven Erik Mattsson, LTH
Arne Otteblad, STU
Karl Johan Åström, LTH

som textförfattare.

Innehållsförteckning

Sammanfattning	3
1. Bakgrund	5
1.1 Verksamhet	5
1.2 Genomförda projekt	6
2. Motiv för fullföljande	7
3. Målsättning för det fortsatta ramprogrammet	8
3.1 Hur kan framtida CACE-system se ut?	8
3.2 Mål för CACE	9
4. Plan för verksamheten	10
4.1 Verksamhet	10
4.2 Budget	11
4.3 Styrgrupp	12
5. Nyttiggörande och spridning av resultaten	12
6. Förslag till beslut	13
Appendix 1 – Genomförda projekt	14
Appendix 2 – Externa kontakter	21

Sammanfattning

Användning av automation och avancerad styrning är av strategisk betydelse för svensk industri. Exempel på detta finns i hela spektret från traditionell processtyrning och elkraftindustri till rymdteknik. För att kunna utveckla system för automatisering och styrning behövs datorstödda verktyg för modellutveckling, simulering, analys, konstruktion och utprovning.

Existerande system har visat sig vara effektiva och arbetsbesparande. Erfarenheter från olika håll indikerar att ingenjörsarbetet i typiska fall kan reduceras med en faktor fyra. De system för datorstödd konstruktion av regler-system som används idag utvecklades för datorer med måttlig kapacitet och med primitiv utrustning för grafisk in- och utmatning. Den snabba utvecklingen inom mikroelektroniken och datalogin som ägt rum de senaste åren har öppnat nya möjligheter att förbättra verktygen. För att utforska dessa möjligheter startades i slutet av 1984 ett femårigt STU-finansierat ramprogram "Datorbaserade hjälpmedel för styrsystem (Computer Aided Control Engineering, CACE)" med institutionen för reglerteknik vid Lunds Tekniska Högskola som huvudaktör. Programmets budget har varit 1.5–2 MSEK per år.

Verksamheten har bedrivits i form av pilotprojekt och har fokuserats på användning av datorgrafik, symbolisk formelmanipulering och expertsystemteknik. Resultaten av projekten har motsvarat förväntningarna och visat att det är möjligt att väsentligt förbättra CACE-verktygen. Det är idag möjligt att förse varje ingenjörsarbetsplats med en kraftfull arbetsstation med högupplösande, snabb grafik. Snabb grafik till rimligt pris gör det möjligt att avsevärt förbättra kommunikationen med användaren. Denne kan rita blockdiagram för att beskriva sina modeller och använda animerad grafik för att presentera resultat. Den ökande datorkapaciteten gör det möjligt att utföra formelmanipulering. Nya landvinningar inom datalogin såsom expertsystem gör det möjligt att ge användaren bättre stöd och hjälp. Man är idag allmänt enig om att CACE-system skall vara utbyggbara. Det går inte att göra det kompletta CACE-systemet en gång för alla. Man måste ha ett flexibelt och modulariserat system med väldefinierade gränssnitt mellan modulerna. Nyckelorden är verktyg, verktygslåda och "verktygsmaskiner". Med det sista ordet menas att verktygen skall vara integrerade så att användaren slipper göra explicita konverteringar.

Ramprogrammet har medfört en betydande kunskapsutveckling, som bland annat finns dokumenterad i två licentiatavhandlingar och en snart fullbordad doktorsavhandling. Därutöver har ett antal vetenskapliga publikationer och internationella konferensbidrag publicerats. Antalet forskare som är verksamma inom området har ökat från 2 till 8 under perioden. Ett betydande internationellt samarbete med forskargrupper i England, Schweiz, Frankrike och USA har utvecklats.

Pilotprojekten i ramprogrammet har visat att det är möjligt att förbättra CACE-verktygen högst avsevärt. Den fortsatta insatsen bör genomföras i form av spjutspetsprojekt. Efter ingående överväganden har det befunnits lämpligt att man koncentrerar sig på verktyg för modellutveckling och simulering. Detta val innehåller de flesta viktiga problemställningarna för CACE-verktyg och passar bra in i det internationella samarbetet.

Avsikten är att man nu utvecklar en prototyp, som kan användas för att illustrera och utvärdera idéer och principer. En kommersiell produkt skulle däremot kräva väsentligt större arbetsinsatser. Tanken är i stället att programvaruföretag eller styr- och reglertekniska industriföretag skall utnyttja resultaten för att ta fram egna produkter. Erfarenheten från tidigare programpaket såsom Simnon och Idpac visar att det är mycket svårt att finna svenska företag som har resurser och marknadskanaler för att separat marknadsföra CACE-program. Istället bör grundidén vara att programmen utgör delar av eller komplement till annan utrustning. I detta sammanhang kan då påpekas att användningen av simulatorer sannolikt kommer av bli vanligare inom processindustrin både som styrhjälpmedel och utbildningsverktyg. Simulering och simulatorer har också en central roll i STU:s planerade insatsområde "Driftsutvecklingssystem för processindustrin" (DUP). Goda möjligheter finns att föra ut CACE-prototyper via DUP.

Styrgruppen föreslår att ramprogrammet fortsätter under ytterligare 2 år med följande omfattning:

Budgetåret 87/88 1.5 MSEK

Budgetåret 88/89 1.5 MSEK

1 Bakgrund

Automatisering och avancerad styrning är i många sammanhang en grundförutsättning för att man överhuvudtaget skall kunna uppnå sina mål. Exempel på detta finns i hela spektret från traditionell processindustri och elkraftindustri till avancerad rymdteknik. Ett annat viktigt skäl till automatisering och avancerad styrning är behovet av att framställa bättre och billigare produkter. För att kunna utveckla system för automatisering och styrning behövs datorstödda verktyg för modellutveckling, simulering, analys, konstruktion och utprovning.

De system för datorstödd konstruktion av reglersystem som används idag utvecklades för datorer med måttlig kapacitet och med primitiv utrustning för grafisk in- och utmatning. Den väsentliga funktionen i systemen är omfattande numeriska beräkningar och en kommandostruktur som gör det möjligt att enkelt formulera många analys- och syntesproblem. Existerande system har visat sig vara effektiva och arbetsbesparande. Erfarenheter från olika håll indikerar att ingenjörsarbetet i typiska fall kan reduceras med en faktor fyra.

Den snabba utvecklingen inom mikroelektroniken som ägt rum de senaste åren har öppnat nya möjligheter till att förbättra verktygen. För att utforska vilka möjligheter de nya teknikerna ger och för att utveckla några prototypsystem, startades i slutet av 1984 ett femårigt STU-finansierat ramprogram "Datorbaserade hjälpmedel för styrsystem (Computer Aided Control Engineering, CACE)".

1.1 Verksamhet

Verksamheten har i stort sett följt de linjer som drogs upp i programplanen av 1984-05-14 för ramprogrammet "Datorbaserade hjälpmedel för styrsystem (Computer Aided Control Engineering, CACE)" och har omfattat ett antal pilotprojekt och etablerande av nya och fördjupning av gamla internationella kontakter. Förändringar som gjorts är införandet av "Expertreglering" (STU-projekt 85-3084) i ramprogrammet och senareläggning av projektet "Implementeringsspråk".

Pilotprojekten har genomförts för att testa idéer och för att undersöka potentialen i de framsteg som gjorts inom mikroelektronik och datalogi. Det är idag möjligt att förse ingenjörsarbetsplatser med kraftfulla grafiska arbetsstationer. Symbolisk formelbehandling och expertsystemteknik är nya element som det är av intresse att introducera i CACE-system. Projekten har också inneburit utveckling av delprototypsystem, vilka är användbara för att visa idéer och principer. Projekten har uppfyllt förväntningarna och visat att det är möjligt att väsentligt förbättra CACE-verktygen. Projekten beskrivs översiktligt i Sektion 1.2 och längre beskrivningar av projekten och erhållna resultat ges i Appendix 1.

Resultaten har presenterats i åtta bidrag till internationella konferenser och de har där blivit positivt mottagna. Två licentiatavhandlingar har skrivits och en doktorsavhandling väntas bli klar under sommaren 1987. Antalet forskare som arbetar med en inriktning som svarar mot ramprogrammet är nu 8 mot 2 innan ramprogrammet startade.

Kontaktnätet med svenska och utländska universitet och företag som använder och konstruerar CACE-verktyg är väl utbyggt och fungerande. De internationella kontakterna är värdefulla och stor vikt har lagts vid att utvidga och fördjupa dem och kontakterna med de flesta betydelsefulla grupper är mycket goda. De externa kontakterna beskrivs i Appendix 2.

1.2 Genomförda projekt

Nedan ges en kort sammanfattning av genomförda projekt under CACE-ramprogrammet. En längre beskrivning ges i Appendix 1.

Högnivåspråk för problemlösning. I projektet "Högnivåspråk för lösning av reglerproblem" (STU-projekt 85-4808) har CACE-system betraktats som språk och miljöer för att lösa reglerproblem. Denna ansats ger ett enhetligt sätt att betrakta och analysera olika CACE-paket. Genom samarbete med utländska grupper kunde ett representativt urval täckas. Systembegreppet är fundamentalt i reglertekniken och representation av system är en nyckelfråga i CACE. Studien visade att här fanns mycket att önska. En specialstudie gjordes därför. Studien visade att en objektorienterad ansats är naturlig och lätt att arbeta med. Detta projekt har haft stort inflytande på planeringen av det fortsatta arbetet.

Grafik. Grafik ger stora möjligheter att förbättra kommunikationen mellan användare och dator. Ett bra och välkänt exempel är Apples Macintosh.

Grafik spelar en viktig roll i reglertekniken. Många metoder för analys och design är grafiska. Blockdiagram används för att beskriva system. Möjligheterna med att använda grafik för att beskriva systemstruktur har studerats i projektet "Nya former av man-maskin kommunikation" (STU-projekt 84-5069). En prototypsimulator kallad Hibliz för dynamiska system har implementerats. Hibliz tillåter definition av modellen i form av Hierarkiska Blockdiagram med Informations-Zoomning. Hibliz har fått ett mycket gott mottagande. Ett mindre projekt med tredimensionell animering av en ASEA-robot har också genomförts. Animering är ett intressant sätt att utnyttja grafik för att visa ett systems beteende.

Symbolisk manipulering. Den ökade datorkapaciteten gör det möjligt att utföra formelmanipulering. Symbolisk manipulering kommer att spela en stor roll. Användaren kan ge sina beskrivningar på ett för honom lämpligt format och från denna beskrivning kan sedan CACE-systemet vid lösning av specifika problem symboliskt förenkla problemet och generera beskrivningar som passar de numeriska verktygen. Analytiska svar kan ge bättre insikt än en tabell av numeriska värden.

Projektet "Kombination av formelbehandling och numerik" (STU-projekt 85-4809) har studerat möjligheterna att använda ett generellt formelmanipuleringspaket, MACSYMA. Erfarenheterna är att det är kraftfullt. Nackdelen är att det är stort och beräkningskrävande. För användning i CACE-program vore det önskvärt att ha modulariserade verktyg. Formelmanipuleringsrutiner har också skrivits i andra projekt som till exempel ekvationssortering i Pascal (STU-projekt 84-5069) och linjärisering i Lisp (STU-projekt 85-4808).

Expertsystemschnitt. Avsikten med projektet "Experiment med expertsystem-interface" (STU-projekt 85-3042) var att undersöka möjligheterna att använda expertsystemmetodik för att förse CACE-system med expertkunskap så att det kan ge målorienterad hjälp till användaren. Ett prototypexpertsystemschnitt till dataanalys och parameterskattningsprogrammet Idpac har designats och implementerats.

En slutsats är att expertsystemteknik är en möjlig väg att introducera expertkunskap i CACE-system. Svårigheten är att samla in och formalisera kunskapen. Bortsett från möjligheterna att introducera expertkunskap kan expertsystemteknik användas för att mera jordnära hjälpa användaren att hålla reda på vad han gjort och vad kommandona heter och ser ut. En intressant

slutsats från projektet att både användandet av Idpac och expertsystemet kunde förenklas om kommandona gjordes mera enhetliga till sin struktur.

Projektet har lett fram till två licentiatavhandlingar (Jan Eric Larsson och Per Persson vid institutionen för reglerteknik, LTH).

Expertreglering. Expertreglering innebär användning av expertsystem för reglering i reell tid. I projektet "Expertreglering" (STU-projekt 85-3084) studeras möjligheterna till att använda expertsystemmetodik i den lokala regler-slingan. Regulatorns mål är att reglera bättre genom att successivt bygga upp mer kunskap genom att göra experiment och analysera resultaten. Expert-systemet används för att dirigera de olika algoritmerna på ett "intelligent" sätt. I en regulator är regleralgoritmen bara en bråkdel av den totala koden. Resten är ett logisk säkerhetsnät för att tillåta drift i olika operationsmoder. Detta nät lämpar sig väl för implementering med ett regelbaserat expertsys-tem. Expertsystemet kan också användas för att naturligt representera den processrelaterade information och de tumregler som en erfaren processopera-tör har. Ett prototypsystem har implementerats och visat att det är en möjlig och användbar ansats. Karl-Erik Årzén vid institutionen för reglerteknik, LTH, kommer under hösten 1987 att lägga fram en doktorsavhandling baserad på sitt arbete inom projektet.

2 Motiv för fullföljande

Användning av automation och avancerad styrning är av strategisk betydelse för svensk industri. Avancerad styrning och automation är livsviktigt för både traditionella processindustrier såsom stål, kemi, papper och massa, och nyare områden såsom bioteknologi, mikrobiologi och läkemedel. Även för verksamheter såsom elkraftindustri, verkstadsindustri, uppvärmning och ven-tilation kommer avancerad styrning att få en ökande betydelse. För att kunna utveckla system för automatisering och styrning behövs datorstödda verktyg för modellutveckling, simulering, analys, konstruktion och utprovning.

Kommersiella tillverkare och marknadsförare av CACE-programvara som till exempel Mitchell and Gauthier Associates (ACSL), Systems Control Tech-nology, Inc. (Ctrl-C) och Integrated Systems, Inc. (MATRIX) har så myc-cket investerat i sin programvara att det är svårt för dem att göra stora förändringar för att bättre utnyttja de nya möjligheter som ges av framste-gen inom datorområdet. Man gör förbättringar genom att lägga till nya op-erationer och införa bättre grafik. Nytänkandet kommer från universitet och forskningsinstitutioner där man inte har samma tvång att vara kompatibel bakåt.

CACE-ramprogrammet har inneburit kunskapsutveckling och undersök-ning av möjligheterna att förbättra CACE-verktygen. Projekten har genere-rat idéer och visat att det finns en potential att förbättra CACE-verktygen väsentligt. Stort arbete har också lagts ned på att bygga upp ett internationellt kontaktnät. Förbindelser med de flesta betydelsefulla grupper inom området är mycket goda. För att på bästa sätt utnyttja det som gjorts i CACE-projekten är det därför väsentligt att CACE-ramprogrammet får fullföljas.

CACE kommer i framtiden att vara utomordentligt viktigt för allt in-genjörarbete. Särarten hos de reglertekniska problemen är att de bygger på abstrakta modeller och avancerade matematiska metoder. De spjutspetspro-jekt som görs inom detta område kan förväntas få spin-off-effekter inom andra områden.

3 Målsättning för det fortsatta ramprogrammet

3.1 Hur kan framtida CACE-system se ut?

Framgångarna inom datortekniken och datalogin har öppnat nya möjligheter till att förbättra CACE-verktygen. Av speciell betydelse på hårdvarusidan är de nya kraftfulla arbetsstationerna. En arbetsstation har ett antal positiva egenskaper:

1. En-användarsystem; ingen konkurrens med andra användare
2. Avsevärd beräkningskapacitet
3. Hög interaktivitet mellan dator och användare
4. Grafik.

En idag vanlig konfiguration är en arbetsstation med en beräkningskapacitet på 1 Mips, 4–8 Mbyte minne, skärm med en upplösning på 1 Mpixel, flerprocessoperativsystem ("multiwindow") och en nätöverföringskapacitet på 1 Mbyte/s. Man måste komma ihåg att utvecklingen är snabb och prestanda ökar kraftigt samtidigt som priserna sjunker. Prisuppgifter blir snabbt inaktuella. Vad man kan säga är att ovanstående konfiguration idag kan köpas för under 200 000 kr samt att det finns tecken på att de inom den närmsta tiden kommer att ligga en bra bit under 100 000 kr. Problemområden är operativsystemen och kommunikation mellan olika arbetsstationer liksom avsaknaden av standard för grafik och fönsterhantering.

Man är idag allmänt ense om att CACE-system skall vara utbyggbara. Det går inte att göra det totala CACE-systemet. Man måste ha ett flexibelt och modulariserat system med väldefinierade gränssnitt mellan modulerna. Nyckelorden är verktyg, verktygslåda och "verktygsmaskiner". Med det sista ordet menas att verktygen skall vara integrerade så att användaren slipper göra explicita konverteringar. Användarsnittet skall vara enhetligt.

För få en flexibel verktygslåda är det viktigt att CACE-systemet är modulariserat på rätt sätt. Grovt sett kan ett CACE-system delas in i tre delar:

1. *Användarsnitt.*

Då olika användare har olika behov, är det viktigt att separera det från beräkningsverktygen. Man kan betrakta ett CACE-system som liggande i en läs-beräkna-skriv-slinga. I läs- och skriffaserna är användarsnittet aktivt. Man kan betrakta ett användarsnitt som bestående av ett språk och en miljö. Språket skall ha sådana begrepp som är kraftfulla och naturliga för användarna. För språk har vi sådana begrepp som syntax och semantik. Man kan säga att syntax definierar tillåtna former och att semantik har med tolkning och betydelse att göra. I en interaktiv miljö, där menyer och grafik används för inmatning, är gränsen mellan språk och miljö mindre markerad. Användaren kan likaväl se syntaxen som en del av miljön.

Användarsnittet kan göras flexibelt genom att tillåta olika syntax: textkommandon, menyer, grafik etc. Olika hjälp- och expertsystemsnitt kan placeras på denna nivå. Användarsnittet skall samla ihop användarens inmatning och sedan konvertera kommandot till en textmässig representation. När detta är gjort skall rätt beräkningsverktyg anropas. När användarsnittet erhållit resultatet skall detta presenteras på lämpligt sätt. Den textmässiga representationen av ett kommando är användbart till dokumentation och för programmering av nya sammansatta kommandon.

2. *Data.*

Databasen bör utgöra en gemensam del för de olika delarna så att verk-

tygen blir integrerade. Vidare bör all information vara tillgänglig för alla. Det är inte bara användaren som skall kunna komma åt all information även kommandoprocedurer, hjälp- och expertsystem måste kunna det.

3. *Beräkningsverktyg.*

Det är viktigt att beräkningsverktygen inte gör någon in- eller utmatning. All kommunikation med användaren måste göras via användarsnittet så att modulariseringen upprätthålls. Ett annat skäl är att all information tillgänglig för användaren måste vara tillgänglig från en kommandoprocedur så att den kan fånga upp felmeddelanden och undvika programkrascher.

3.2 Mål för CACE

Det fortsatta arbetet i CACE-projektet planeras att liksom förut bedrivs både på bredden och djupet. Största insatsen avses att göras i form av spjutspetsprojekt. Bredden kommer genom insamlande av idéer, erfarenheter, resultat och programvara från internationella kontakter och samarbetspartners. Resultaten från spjutspetsprojekten utgör här en nödvändig bytesvara.

När det gäller spjutspetsdelen är nästa naturliga steg att utnyttja erfarenheterna från pilotprojekten till att bygga något slag av CACE-system. Efter ingående avvägningar i styrgruppen läggs tyngdpunkten på en verktygslåda för modellutveckling och simulering. Det är ett område av stor betydelse och det innehåller de flesta viktiga aspekter vid design av CACE-system såsom symbolisk manipulering, representation av system samt användarsnitt med grafik för att beskriva system och presentera resultat. Det är också ett område där forskarna i CACE-ramprogrammet har tradition och är starka. Valet passar också bra in i det internationella samarbetet. Flera grupper i bland annat England och USA arbetar på att göra verktyg för speciella analys och syntes problem.

Planen är att bygga en prototyp, som kan användas för att illustrera och utvärdera idéer och principer. En kommersiell produkt måste vara robust och vältestad. Det kräver stora arbetsinsatser. Enligt programplanen av 1984-05-14 är det inte heller avsikten att forskningsarbetet skall resultera i produkter utan överföringen till kommersiella produkter skall göras av programvaruföretag.

Skälen till att bygga en prototyp är flera. Med en prototyp som är någorlunda användbar kan man lättare få användarnas synpunkter och idéer till förbättringar. Genom att arbetet utförs i en universitetsmiljö kan man också snabbt få synpunkter från många användare såsom forskare, doktorander och teknologer.

Även själva konstruktionsarbetet ger nödvändig återkoppling. När man verkligen försöker bygga något får man svar på om det är möjligt och upptäcker lättare brister. En prototyp är också ett bra sätt att förmedla principer och idéer till de programvaruföretag som skall göra de kommersiella produkterna.

Vid implementeringen kommer flexibilitet att sättas före effektivitet så att det blir lättare att modifiera och testa olika sätt. Detta är speciellt viktigt när man gör användarsnittet. Det enda verkliga sättet att utvärdera en idé är att implementera och prova. Planen är att implementera prototypen i Common Lisp med utnyttjande av annan kod skriven i andra språk. Ett problem är avsaknaden av standard eller bra implementering av existerande standard för bland annat grafik och fönsterhantering, där de olika tillverkarna av arbetsstationer har sina egna system.

4 Plan för verksamheten

4.1 Verksamhet

Följande moment planeras ingå i verksamheten:

1. *Verktygslåda för modellutveckling och simulering.*
Som spjutspetsprojekt föreslår vi utveckling av verktyg för modellutveckling och simulering. Viktiga moment är representation och presentation av systemstruktur och systembeteende. Det skall finnas struktureringsmekanismer för att handha till exempel modelluppdelning och multipla representationer. Användaren skall kunna definiera modellstruktur genom att rita diagram. För att få en flexibel beskrivning av ett delsystems beteende som kan användas i olika sammanhang skall den vara på symbolisk form. Användaren kan då beskriva sitt system på ett för honom bra sätt. Från denna beskrivning skall CACE-systemet sedan kunna generera effektiv simuleringskod, kod för att beräkna stationära punkter, lineariserade representationer, effektiv reglerkod, beskrivningar som accepteras av andra paket etc. Resultat från analyser och simuleringar skall kunna presenteras på olika sätt.
2. *Integrering av tredjeparts programvara.*
Undersökning av möjligheterna till att integrera programvara som kan fås från utländska kontakter och samarbetspartners. Detta är en nyttig övning i sig själv då framtida system kommer att sammansättas av olika paket. Typexempel är matris- och polynompaket.
För de fall det är omöjligt eller mycket arbetssamt att introducera främmande programvara kan man göra konverteringsrutiner. Det ger inte integrerade verktyg, men ett sätt att kunna utnyttja kommersiell programvara där man inte har tillgång till källkod.
3. *Tillämpningsstudie.*
För att bättre kunna utvärdera verktygen som utvecklas för modellutveckling och simulering är det av intresse att utföra en tillämpningsstudie så att man får in "verklighetsbiten". I detta fall kan det vara möjligt att samarbeta med University of Maryland, där man bedriver projekt för modellering av kemiska processer. Sydkraft och Vattenfall finansierar en gästprofessur i reglering av kraftsystem vid institutionen för reglerteknik, LTH. Det närmaste året avser man att arbeta med modellering av kraftverk. Här finns också potentiella synergieffekter.
4. *Dokumentation.*
Dokumentation av utfört arbete är tidsödande och känns ofta tråkigt. Det är därför av stort intresse att studera möjligheterna till att utveckla hjälpmedel för rapportgenerering och automatisk dokumentation. Till exempel har det program som utvecklats för att konvertera MACSYMA-utskrift till ett format som accepteras av sättningsprogrammet $\text{\TeX}$ visat sig vara mycket användbart. Bland annat undviker man att göra de fel som lätt uppkommer vid manuell konvertering.
5. *Internationella kontakter och samarbeten.*
De internationella kontakterna och samarbetena är viktiga. Sveriges insats inom området utgör endast en liten del av den totala forskning och utveckling som görs i världen. Det är därför viktigt att hålla sig a jour och samla in och ta hem resultat. Det är viktigt att man gör studiebesök

hos varandra och driver gemensamma projekt. Artiklar och rapporter ger inte alls samma utbyte som riktiga demonstrationer med möjlighet till att själv få prova. Det är också viktigt att man kan förhandla sig till byte av experimentell programvara. Det är därför betydelsefullt att det avsätts resurser till resor så att forskarna i ramprogrammet kan göra studiebesök och deltaga på internationella konferenser. Det kan också vara av intresse att ha gästforskarutbyte. Flera utländska grupper har uttalat intresse för detta.

Som redogjorts för i sektion 1.1 är kontakterna goda med de flesta betydelsefulla grupperna. Dessa kontakter skall upprätthållas och nya skapas när det är av gagn. När det gäller närmare samarbete finns redan intresseavtal med de engelska grupperna och University of Maryland om samarbete. University of Maryland har liknande samarbetsavtal med INRIA i Frankrike. Samarbete med Lawrence Livermore Laboratory i Kalifornien är också av stort intresse.

De utländska kontakterna ger möjligheter till att ta hem olika former av CACE-programvara som, enligt moment 2, kan inkluderas i prototypsystemet. Ett annat sätt att utnyttja ny programvara är att prova den i forskarutbildningskurser. I forskarutbildningskurser om syntes av regler-system är det naturligt att utnyttja olika sorters programpaket. På detta sätt kommer fler personer att känna till och kunna använda olika paket. Detta ger värdefulla erfarenheter. Man kan också finna intressanta idéer som kan vara värdefulla att utveckla. När man hållit en sådan doktorandkurs har man en möjlighet att sprida kunskap till en större krets genom att hålla externkurser för folk verksamma i industrin.

6. *Speciella problem.*

Under arbetets gång kan det dyka upp speciella problem som i sig själva kan vara värda ett studium. Ett sådant exempel är kodgenerering till hårdvara som skall utföra regler- och styrfunktioner. Det vore bra om man då direkt kunde utnyttja den beskrivning som man erhållit från designen och använt vid simulering.

4.2 Budget

År	Löner, utrustning, material, resor
1987/88	1500 kkr
1988/89	1500 kkr
Summa:	3000 kkr

4.3 Styrgrupp

Styrgruppen för ramprogrammet har innevarande budgetåret bestått av följande personer:

Sven Gunnar Edlund, STFI, ordförande
Arne Otteblad, STU
Karl Eklund, Dätama-Asea, Processdata AB
Claes Källström, SSPA
Eric Sandewall, LiTH
Gustaf Söderlind, ITM, KTH, LTH
Karl Johan Åström, LTH

Avsikten är att denna sammansättning bibehålls de kommande budgetåren.

5 Nyttiggörande och spridning av resultaten

De vetenskapliga resultaten från ramprogrammet kommer liksom förut att göras allmänt tillgängliga genom presentationer på internationella konferenser och publicering av artiklar i internationella tidskrifter. Resultaten har och kommer också att publiceras som licenciat- och doktorsavhandlingar.

Forskningen leder till att kunskaperna hos institutionernas forskare och doktorander ökar. Forskarutbildningskurser inom området har givits och kommer att ges. Examensarbeten ger också möjlighet till informationsspridning. Kunskaps- och metodöverföring till industrin via anställning av duktiga och välutbildade forskare och ingenjörer har visat sig vara ett bra sätt. En ökning av kunskapsnivån på universitetsinstitutionerna gör det möjligt att ge kurser inom området för folk verksamma inom industrin. Att sprida resultaten till ingenjörer verksamma i industrin är centralt därför att den kanske viktigaste konsekvensen av CACE-ramprogrammet är att öka den enskilde ingenjörens produktivitet och kvaliteten på de lösningar han presterar.

Informationsöverföring till industrin och andra forskningsinstitut kan också ske genom studiebesök och samarbetsprojekt. Goda kontakter finns redan med till exempel STFI, SSPA, ASEA, Alfa-Laval, SattControl och Sydkraft. Som ett exempel kan nämnas att Sydkraft avser att bygga upp modellbibliotek för kraftverkskomponenter och har i samband därmed haft diskussioner med forskarna i ramprogrammet och följt med på en studieresa till CEGB i England.

Programvaruföretag eller industri inom styr- och reglerområdet kan utnyttja resultaten för att generera egna produkter. En svårighet är att den kommersiella marknaden för CACE-produkter är relativt liten och det är svårt att göra dem självfinansierande. Den svenska marknaden är för liten. Man måste satsa på större marknader. Erfarenheterna från programpaketen Simnon och Idpac visar att det är mycket svårt att hitta något företag i Sverige som har resurser, reglerteknisk kunskap och marknadskanaler för att marknadsföra CACE-program. Ett sätt att lösa problemet vore att starta samarbete med utländskt företag. Vissa delar kan också spinnas av till att göra en förbättrad version av Simnon.

Förutom att framställa generella CACE-verktyg kan man utnyttja resultaten indirekt. Specialiserade CACE-verktyg kan vara väsentliga för att ett företag skall kunna förbättra och förbilliga sina produkter både i design och produktionsledet och kan motivera egen eller branschvis satsning.

Ett annat sätt att utnyttja forskningsresultaten är att bygga simulatorer eller virtuella processer. Genom att kombinera simulering med animering och ett bra användarsnitt kan man skapa realistiska simulatorer som beter sig så gott som en verklig process. Inom till exempel kraftindustrin och flygväsendet har man länge använt simulatorer för att utbilda sin personal. Tyvärr har kostnaden varit för hög för att använda sådana simulatorer inom andra områden, men det håller nu på att ändras. Man kan här nämna att programpaketet Simnon har använts för att göra en kompaktsimulator till lågt pris för operatörsutbildning på Barsebäck. Användning av simulatorer kommer av bli vanligare inom processindustrin. Dels för att utbilda personal i hantering av nödlägen och dels för att ge dem en möjlighet att lära sig processen genom att simulera effekten av olika ingrepp i normal drift. Man kan också utnyttja simulatorer eller virtuella processer inom ingenjörsutbildningen och på detta sätt få flexibla, realistiska, lättunderhållna och säkra labprocesser. Simulering och simulatorer har en central roll i det av STU föreslagna insatsområdet "Driftsutvecklings system för processindustrin"(DUP). Det finns goda möjligheter att föra ut prototyper och verktyg via DUP. När CACE-programmet avslutas har DUP varit i gång i 2 år.

6 Förslag till beslut

Styrgruppen föreslår att styrelsen godkänner föreliggande planering av ramprogrammet "Datorbaserade hjälpmedel för utveckling av styrsystem" med följande medelsdisposition:

Budgetåret 87/88 1.5 MSEK

Budgetåret 88/89 1.5 MSEK

Appendix 1 – Genomförda projekt

Detta appendix beskriver de projekt som genomförts under ramprogrammet "Datorbaserade hjälpmedel för styrsystem (Computer Aided Control Engineering, CACE)".

Avsikten med projekten har varit att testa idéer och att undersöka hur de framsteg som gjorts inom mikroelektronik och datalogi kan utnyttjas för att förbättra CACE-verktygen. Projekten har fokuserats på användning av grafik, symbolisk formelmanipulering och expertsystemteknik. Resultaten har uppfyllt förväntningarna och visat att det är möjligt att väsentligt förbättra CACE-verktygen. Projekten har också inneburit utveckling av delprototypsystem, vilka är användbara för att visa idéer och principer. Resultaten har presenterats i åtta bidrag till internationella konferenser och de har där blivit positivt mottagna. Projektet "Experiment med expertsysteminterface" (STU-projekt 85-3042) har lett fram till två licentiatavhandlingar (Jan Eric Larsson och Per Persson, institutionen för reglerteknik, LTH). Karl-Erik Årzén, institutionen för reglerteknik, LTH kommer under hösten 1987 att lägga fram en doktorsavhandling baserad på sitt arbete i projektet "Expertreglering" (STU-projekt 85-3084).

De olika projekten och resultaten presenteras nedan och indelningen är där gjord efter ämnesområde och ej efter pilotprojekt.

Högnivåspråk för problemlösning

Ett CACE-system kan betraktas som ett språk och en miljö för att lösa reglerproblem. Språket skall vara mer än ett medel för att instruera datorn att utföra olika operationer. Det skall vara en grundstomme på vilken man kan bygga sina idéer. Vid konstruktion av ett språk skall speciell uppmärksamhet ägnas medlen för att kombinera enkla idéer till mera komplexa. Ett kraftfullt språk skall tillhandahålla primitiva datatyper, metoder för att kombinera dem till sammansatta objekt och möjligheter till abstraktion med vilka sammansatta objekt kan namnges och manipuleras som enheter.

Valet av primitiva objekt innebär identifiering av grundläggande och kraftfulla begrepp. En reglertekniker arbetar på flera abstraktionsnivåer, så detta bör också avspglas i språket. Det skall ha de grundläggande matematiska datatyperna boolean, integer, real och complex. Vektorer, matriser och uttryck med and, or, =, +, -, *, /, ^, if-then-else, matematiska standardfunktioner och användardefinierade funktioner är användbara sätt att komponera nya objekt. Ett CACE-system bör kunna handha uttryck på symbolisk form. Det skall vara möjligt att använda matriser i uttryck liksom att ett matriselement skall kunna vara ett uttryck. Symbolisk manipulering av uttryck diskuteras mera nedan.

Systembegreppet är fundamentalt i reglertekniken. Det är önskvärt att ha mera sofistikerade datatyper än tal, fält och uttryck för att beskriva system så att man kan se dem som enheter. Representation av system har studerats i projektet "Högnivåspråk för lösning av reglerproblem" (STU-projekt 85-4808). Ett prototypsystem för en objektorienterad representation av system har implementerats i ExperLisp på en Macintosh.

Representation av system har också behandlats i projektet "Nya former av man-maskin kommunikation" (STU-projekt 84-5069). Detta projekt behandlas närmare under punkterna datorgrafik och symbolisk manipulering.

Man har funnit att systembegreppet i reglertekniken är komplext och är inte trivialt att formalisera. Man kan betrakta ett system på så många olika

sätt. Man kan ha modeller av olika komplexitet. En modell med partiella differentialekvationer kan diskretiseras till en modell med olinjära ordinära differentialekvationer, som i sin tur kan linjäriseras. Modellen kan samplas. Linjära modeller kan transformeras till frekvensplanet. Relationerna mellan olika modeller av ett system kan alltså bli komplexa.

De flesta av dagens CACE-program accepterar bara rena matematiska beskrivningar. Det vore önskvärt att också kunna diskutera kvalitativt om system och att kunna handha ofullständig och osäker information.

En användare måste också kunna introducera nya operationer. Han skall kunna definiera procedurer med parametrar och styrsatser som *if-then-else*, *while-do* etc. Det är viktigt att all information såsom systemparametrar som beskriver hur användaren vill att CACE-systemet skall bete sig, dimensioner på matriser etc är tillgängliga. Det måste vara möjligt att fånga upp och behandla fel så att urspårningar undviks.

För språk har man begreppen syntax och semantik. Man kan säga att syntaxen definierar de tillåtna formerna på satserna och att semantiken har att göra med tolkning och betydelse. I en interaktiv miljö, där menyer och grafik används för inmatning, är gränslinjen mellan språk och miljö mindre markerad för användaren. En användare bör kunna definiera nya operationer genom att visa medan CACE-systemet loggar. Han skall sedan kunna introducera sin operation i en meny. För loggning och dokumentation är det värdefullt att också ha en textmässig beskrivning av objekt och operationer. En textmässig-(ASCII)-beskrivning är även mer portabel. Skall man definiera en mera komplex operation med *text if-then-else* är editering av text troligen både lättast och effektivast. Följaktligen behövs textrepresentationer för objekt och operationer. Den kan användas som en gemensam bas, på vilken det kan byggas olika användarsnitt med grafiska representationer och expert-systemfaciliteter som anpassas till användarens behov och önskemål.

Datorgrafik

Utvecklingen inom mikroelektronikområdet har gjort det möjligt att förse ingenjörsarbetsplatser med kraftfulla grafiska arbetsstationer. Detta är viktigt, ty grafik ger stora möjligheter att förbättra kommunikationen mellan användare och dator. Ett bra och välkänt exempel är Apples Macintosh. Menyer och ikoner minskar kravet på memorering. Användaren väljer och behöver inte skriva in ett kommando enligt någon invecklad syntax.

Grafik spelar en viktig roll i reglertekniken. Många metoder för analys och design är grafiska och utnyttjar *text* trendkurvor, fasplan, rotorter, Bode- och Nyquistdiagram. Det är också naturligt att använda blockdiagram för att beskriva system och modeller. En grafisk beskrivning av en uppdelning av ett system i delsystem är mer begriplig än en textmässig sådan.

I projektet "Nya former av man-maskin kommunikation" (STU-projekt 84-5069) implementerades en prototypsimulator för dynamiska system. Den kallas Hibliz och tillåter definition av modellen i form av Hierarkiska BLockdiagram med Informations-Zoomning. Användaren kan panorera och zooma kontinuerligt i diagrammet. På översta nivån ser han bara namnförsedda rektanglar som är förbundna med linjer vilka beskriver växelverkan mellan delsystemen. Börjar han zooma in öppnar blocken upp sig och visar sin inre struktur, som kan vara ett nytt blockdiagram eller en matematisk beskrivning av delsystemets beteende i form av ekvationer. Detta sätt att representera systemstruktur har fått ett mycket gott mottagande. Användning av block-

diagram för att beskriva systemstruktur rekommenderas, men det bör också påpekas att det finns en rad obesvarade detaljfrågor rörande utseende och användarsnittet för att editera dem. Hibliz har implementerats på den kraftfulla grafiska arbetsstationen IRIS 2400 från Silicon Graphics. Men man kan klara sig med enklare hårdvara om man minskar kravet på att panorering och zoomning skall kunna göras kontinuerligt.

För presentationer av Hibliz finns det diabilder och en sju minuter lång videofilm.

Animering av system är ett spännande och bra sätt att utnyttja grafik om det görs på rätt. Arbetsstationen IRIS 2400 kan t ex animera en industrirobot i tredimensionell, skuggad färggrafik. Ett av arbetsstationens demonstrationsprogram har modifierats till att visa ASEAs industrirobot IRB 6/2. Programmet kan läsa koordinater för lederna från en fil, som kan vara resultatet från en simulering eller en mätning av ett verkligt experiment. Medan roboten utför sina rörelser kan man med användning av musen röra sig runt roboten och studera dess beteende från olika vinklar. Animeringen känns mycket verklig. På Deutsche Forschungs- und Versuchanstalt für Luft- und Raumfahrt, München tänker man sig att använda denna typ av animering vid styrning av robotar i rymden. Det är svårt att styra en robot i rymden från jorden på grund av tidsfördröjningar. Deras idé är att operatören tittar på en bild där man visar en animerad robot utan tidsfördröjning. De har också använt animering för att studera hjulens rörelse hos tåg. Animering är också användbar för att visa idéer och resultat för kunder och lekmän.

Ett besvärligt problem med grafik är avsaknad av en bra och heltäckande grafikstandard. Den nuvarande standarden GKS omfattar bara 2D-grafik och det saknas bra implementeringar. Ett grundläggande svaghet är att de flesta implementeringar inte stöder samplad och händelsestyrd inmatning. Det innebär att inmatning bara kan ske när programmet begär så och det blir då hängande tills användaren svarat. I det engelska ramprogrammet, där man rekommenderas att följa GKS har det lett till att man utnyttjat andra grafikimplementeringar och försökt skriva egna GKS-liknande skal. En annan aspekt är att det arbetas på en mera omfattande standard PHIGS, som skall ha kraftfullare primitiver. PHIGS stöds kraftigt av IBM. Ett annat problem är fönsterhantering. Standard kan inte väntas före 1989. Man kan gissa att under tiden kommer X-windows att bli en de facto standard. Den nuvarande pragmatiska ansatsen är att isolera grafik och fönsterhantering med ett antal rutiner som anses vara möjliga att implementera på de flesta arbetsstationer.

Symbolisk manipulering

Den ökade beräkningskapaciteten hos datorer möjliggör symboliskt formelbehandling. Det finns två huvudskäl till symbolisk formelbehandling. För det första så är ett analytiskt uttryck ett mera komplett svar. Struktur är viktigt och ett analytiskt uttryck kan ge bättre insikt än en tabell av numeriska värden. Detta gäller speciellt om uttrycket beror på flera variabler. Med ett analytiskt uttryck är det lättare att visa begränsningar och möjligheter. För det andra kan symbolisk formelbehandling användas till att ge användaren större frihet vid definition av sina problem. Man kan generera kod för numerisk lösning från en högnivåbeskrivning. Numeriska problem kan elimineras och problem kan transformeras till en form som passar bättre för numerisk lösning.

Det är viktigt att matematiska beskrivningar kan ges på en naturlig form,

utan att behöva bearbetas manuellt. Vid härledning av fysikaliska modeller erhåller man ofta system av differentialekvationer och algebraiska ekvationer (DAE-system). Följaktligen bör verktyg för modellutveckling och simulering kunna hantera DAE-system. Hibliz gör det, men de flesta simuleringspaket kräver att användaren anger hur derivator och algebraiska variabler skall beräknas. Manuell utlösning är arbetssamt och inte trivialt. Fel kan lätt introduceras. Om man kräver utlösta variabler är det svårt att utnyttja modellbibliotek, eftersom det är ett globalt problem att avgöra vilken variabel som skall lösas från vilken ekvation. Man kan inte göra det individuellt för varje delsystem utan måste också ta hänsyn till hur de är kopplade. Tyvärr har de numeriska DAE-lösarna en del brister, som i princip kan elimineras med symbolisk förbehandling. Vid simulering gör Hibliz en del formelbehandling före numerisk lösning, men det återstår mycket att göra.

Den i projektet "Högnivåspråk för lösning av reglerproblem" utvecklade prototypen för systemrepresentation som är skriven i Lisp kan utföra en del formelmanipulering såsom linjärisering.

För att få erfarenhet av att använda verktyg för symbolisk formelbehandling startades projektet "Kombination av formelbehandling och numerik" (STU-projekt 85-4809). Som tillämpning valdes analys och design av multivariabla, linjära system och som verktyg valdes MACSYMA, som är ett stort formelbehandlingsprogram utvecklat vid MIT Laboratory for Computer Science.

För analys och design av flervariabla system finns det en klass av metoder som baseras på polynommatriser. Tyvärr så har dessa metoder i allmänhet dåliga numeriska egenskaper. Metoder som baseras på tillståndsrepresentationer har bättre numeriska egenskaper. Emellertid är det i många fall önskvärt att kunna arbeta i frekvensplanet. MACSYMA är bra på att hantera polynom och rationella funktioner. I projektet utvecklades funktioner i MACSYMA för linearisering, stabilitetsanalys, sampling och olika faktoriseringar och realiseringar. Det är också möjligt att skriva ut systemen på en fil, vilken kan läsas av analys- och designprogrammet CTRL-C för vidare numerisk behandling eller filtreras genom ett program som genererar en modell som kan användas för simulering i Simnon. Då utskriften från MACSYMA är i radskrivarformat och kan vara svår att läsa, utvecklades ett program som konverterar utskriften till ett format som accepteras av sättningsprogrammet T_EX. Detta ger också ett sätt att bekvämt inkludera resultat i rapporter.

Erfarenheterna från denna tillämpning och annan användning är att MACSYMA är ett kraftfullt verktyg. Nackdelen är att MACSYMA är ett stort och beräkningskrävande program. Tyvärr är inte MACSYMA modulariserat. För användning i CACE-program vore det önskvärt att ha modulariserade verktyg för symbolisk formelbehandling.

Expertsystemsnitt

Om man vill använda ett CACE-program för att lösa ett realistisk problem, måste man ge en hel sekvens av kommandon för att nå målet. Hjälpssystemen kan idag typiskt bara lista tillgängliga kommandon och en beskrivning av dessa. De kan inte ge målorienterad hjälp. De kan inte föreslå lämpliga kommandosekvenser. Användaren måste därför känna till metoderna och helst också den underliggande teorin. Tyvärr så räcker det inte med att gå en kurs. Man måste också ha erfarenhet. Man måste känna till en hel del mer än vad som står i läroböckerna för att nå ett verkligt bra resultat.

För att undersöka möjligheterna till att använda expertsystemteknik för att hjälpa användaren startades projektet "Experiment med expertsystem-interface" (STU-projekt 85-3042). Som tillämpning valdes konstruktion av ett expertsystemschnitt till dataanalys- och parameterskattningsprogrammet Idpac. Det uppskattades som ett problem av lämplig storlek, varken trivialt eller för stort. Vidare är dataanalys till en stor del ett hantverk och kvalitén på resultaten beror på användarens skicklighet.

Ett prototypexpertsystemschnitt till Idpac har designats och implementerats. Det är skrivet i FranzLisp med Flavours och YAPS. Koden är cirka 5000 rader lång och kan köras under VMS på en VAX 11/780. Expertsystemsnittet residerar i två olika processer som kommunicerar via brevlådor. På detta sätt behövde man anmärkningsvärt nog inte göra några modifieringar i Idpac.

En grundläggande idé är kommandospionkonceptet. Expertsystemet hjälper bara till på begäran. Det kan köras i två moder: expert och nybörjare. I expertmod kör man Idpac precis som vanligt och expertsystemet svarar då bara på explicita frågor från användaren. För att kunna besvara frågor håller den reda på vad som händer och försöker lista ut vad användaren försöker göra utan att ställa frågor. I nybörjarmod ger den varningar och råd vid varje kommando och ställer frågor för att få en bättre uppfattning av vad användaren försöker uppnå.

Expertsystemet försöker ge hjälp på tre nivåer. För det första ger det varningar och råd om vad som är lämpliga kommandon i nästa steg. För det andra accepterar det kortformer på kommandon och kan fråga efter parametrar eller fylla i dem själv utgående från föregående kommandon. För det tredje så håller det reda på de filer med modeller och data som skapas av Idpac. Det kan ge en halvgrafisk representation av relationen mellan filerna.

För att nå ett mål måste användaren utföra en sekvens av kommandon. Om man kan anta att de flesta målen kan klassificeras i ett relativt litet antal klasser, kan kunskap representeras som kommandosekvenserna för att nå detta mål. För att beskriva denna kunskap finns det ett speciellt spåk för att beskriva kommandosekvenser blandat med fakta. Det innehåller element för att beskriva alternativa vägar, repetition och nästning av sekvenser. Varje kommandosekvens beskrivs också av ett antal regler som evalueras när nya fakta påträffas vid matchning mot användarens kommandosekvens.

En slutsats är att expertsystemteknik är en möjlig väg att introducera expertkunskap i CACE-system. Svårigheten är att samla in och formalisera kunskapen. Expertsystemet har endast försetts med kunskap om ML-skattning. Det omfattar cirka 60 regler. För Idpac uppskattar man att det behövs ett tusental regler. Vid konstruktion av reglerna stöter man på frågor som man inte kan besvara direkt. Förhoppningsvis kan detta också inspirera till ny forskning.

Bortsett från möjligheterna att introducera expertkunskap kan expertsystemteknik användas för att mera jordnära hjälpa användaren att hålla reda på vad han gjort och vad kommandona heter och ser ut. En intressant slutsats från projektet att både användandet av Idpac och expertsystemet kunde förenklas om kommandona gjordes mera enhetliga till sin struktur.

I förlängningen kan man tänka sig att använda en del idéer från projektet för att göra vertyg så att man enklare kan göra flexibla och personligt anpassade användarschnitt.

Expertreglering

Existerande CACE verktyg understödjer för det mesta enbart analys- och designfasen av regulatorkonstruktionen. Själva implementeringen av styrlagen förbises ofta. En mycket viktig beståndsdel i varje praktiskt användbar regulator är det logiska säkerhetsnät som behövs runt själva regleralgoritmen. Detta säkerhetsnät handhar den logik som behövs för t.ex. omställning mellan olika moder, parameterbyten, alarmtillstånd etc. Erfarenheten har visat att detta system ofta är den dominerande delen både vad det gäller storlek och utvecklingstid. I projektet "Expertreglering" (STU-projekt 85-3084) undersöks möjligheten att implementera detta säkerhetsnät med hjälp av expertsystemteknik.

Namnet "Kunskapsbaserad regulator" har införts för att beteckna en regulator bestående av ett kunskapsbaserat system och ett antal olika regler-, identifierings- och övervakningsalgoritmer. Funktionellt sett liknar denna regulator en auto-tuner. Regulatorns mål är att successivt bygga upp processkunskap genom olika dynamiska experiment som t.ex. reläexperiment och stegsvarsexperiment. När nödvändig processkunskap har extraherats skall en lämplig regulator designas och startas. Skillnaden mot en traditionell auto-tuner är att den kunskapsbaserade regulatorn är betydligt mer flexibel. Detta innebär t.ex. att den skall ha möjlighet att identifiera olika processklasser och att välja regulator och designa regulatorparametrar utefter detta. Regulatorn skall också till viss del ha möjlighet att utnyttja tillgänglig processkunskap. Erfarna processoperatörer har ofta kvalitativ kunskap om processen som kan vara svår att utnyttja i konventionella regulatorer.

De expertsystemegenskaper som främst utnyttjas är den explicita, deklarativa kunskapsrepresentationen och den modulära uppbyggnaden. Den modulära uppbyggnaden möjliggör att systemet används som en testbänk för experiment med nya regulatorstrukturer. Den explicita kunskapsrepresentationen möjliggör bl.a. att omfattande utfrågningsmöjligheter inkluderas i regulatorn. Några exempel på frågor som är relevanta är "Vilken processmodel används?", "Vilken reglerdesignmetod har använts och varför?" etc.

Betraktat som en expertsystemapplikation innehåller en kunskapsbaserad regulator flera distinkta problemtyper. Några exempel är klassificering, övervakning, design etc. Det är också en realtidstillämpning. Realtidsexpertsystem är ett mycket intressant område där mycket återstår att göra. Grundläggande teoretiska problem, som t.ex. icke-monotont resonemang, uppstår och försvårar ett strikt teoretiskt angreppssätt. I detta projekt betraktas i stället expertsystemtekniken som en högnivåprogrammeringsteknik.

En programmeringsomgivning för kunskapsbaserad reglering har tagits fram på en VAX. Det består av tre parallella subprocesser; det kunskapsbaserade systemet, de numeriska algoritmerna och användargränssnittet. Dessa kommunicerar genom meddelandesändning.

De expertsystemskal som finns tillgängliga är i regel inte avsedda att användas i realtid. I projektet har ett realtidsskal tagits fram. Detta system bygger på "blackboard"-idéer. En global databas är tillgänglig för ett antal olika kunskapskällor (knowledge sources). Dessa implementerar domänkunskap för något delproblem. De är ofta knutna till en viss numerisk algoritm. Olika metoder för att representera kunskap kan användas. Några exempel är regelbaserad representation med framåtsökning eller bakåtsökning och procedurorienterad representation. En separat "scheduler"-modul bestämmer vilken kunskapskälla som för tillfället skall utnyttjas. Systemet kan liknas vid ett van-

ligt realtidsexpertsystem. Olika realtidsprimitiver har implementerats. Dessa möjliggör att en kunskapskälla väntar en viss tid eller på ett visst databas-element. Det är också möjligt att implementera regler som testas med en viss tidsperiod och att knyta giltighetstider till databaselement. Uppstartsfasen i den kunskapsbaserade regulatorn består av en sekvens, med parallella inslag, av olika kunskapskällor. Detta kan liknas vid en plan och angripas med "plan-ning" metodologi.

De experiment som hittills har utförts har främst rört intelligenta PID-regulatorer. En annan möjlig tillämpning är automatisk uppstart av adaptiva regulatorer.

Implementeringsspråk

Ett CACE-system har många olika saker att utföra: In/utmatning, kommandoavkodning, grafik, symbolisk formelbehandling, numerik, databashantering, expertsystem etc. För att implementera detta finns det många språk att välja mellan: Fortran, C, Pascal, Ada, Lisp, Prolog etc. Numerisk programvara är till övervägande delen skriven i Fortran och kommer att så förbli åtminstone ett tag till. Kanske Ada kommer att överta. Expertsystem är typiskt skrivna i Lisp. En slutsats är att CACE-system kommer att vara implementerade i ett antal språk. Olika problem kräver olika verktyg och man väljer det språk som passar bäst. De flesta arbetsstationer har Fortran, C och ofta Pascal som standardspråk och Common Lisp som tillbehör. Det finns normalt möjligheter att anropa procedurer skrivna i andra språk.

Ett oväntat och viktigt resultat för forskarna i ramprogrammet var att Common Lisp är användbart i de flesta tillämpningar och inte bara för att skriva till exempel expertsystem. Genom sin interaktivitet ger Common Lisp en mycket mer lättarbetad och mer flexibel miljö jämfört med den som konventionella språk av typen C, Fortran och Pascal har.

I CACE-projektet behövs en mycket flexibel och interaktiv miljö för att man lätt och snabbt skall kunna få upp prototyper för att testa idéer. Både erfarenheterna från pilotprojekten och andras erfarenheter indikerar att Common Lisp är ett lämpligt implementeringsspråk i CACE-projektet. Det ger en flexibel interaktiv miljö. Implementering går fortare och koden blir kortare. Vid ändringar slipper man ifrån långa kompilerings- och länkningstider. Man kan se ett CACE-system som ett högnivåspråk för att lösa reglerproblem. Lisp är väl ägnat för att implementera andra språk. Om man inte finner ett sådant högnivåspråk tillräckligt för sina tillämpningar kan man lätt byta till Lisp ett tag.

Appendix 2 – Externa kontakter

Detta appendix behandlar de externa kontakterna inom CACE-ramprogrammet.

Kontaktnätet med svenska och utländska universitet och företag som använder och konstruerar CACE-verktyg är väl utbyggt och fungerande. Det är krävande att bygga upp ett brett kontaktnät. Den breda kontaktyta och det goda rykte som institutionen för reglerteknik vid LTH byggt upp i tidigare projekt och genom försäljning av licenser för programpaketen Simnon och Idpac har varit till stor fördel.

Redogörelse för resor och besök har gjorts vid styrgruppsmötena och har dokumenterats i mötesrapporterna. I Sverige har bland annat Alfa-Laval, ASEA, Ericsson, SAAB, SattControl, SSPA, STFI, Sydkraft, Telelogic, LiTH och SICS besökts eller gjort besök.

De internationella kontakterna är värdefulla och stor vikt har lagts vid att utvidga och fördjupa dem. Forskarna inom CACE-ramprogrammet har haft åtta bidrag på internationella konferenser. De internationella konferenserna inom området ger också bra tillfällen till att utbyta idéer och information och att lära känna nya personer och grupper med gemensamma intressen så att kontaktnätet kan vidgas.

De flesta personer inom området som fått en presentation av verksamheten inom CACE-ramprogrammet har funnit den både omfattande och spännande. Många vill också etablera samarbete i någon form. Resultat såsom Simnon och Idpac från äldre projekt har varit värdefullt handelsgodis vid byte av programvara.

Kontakterna är goda med de flesta betydelsefulla grupper såsom: Frankrike: INRIA, Tyskland: Deutsche Forschungs- und Versuchsanstalt für Luft- und Raumfahrt (DFVLR), Schweiz: ETH, USA: Rensselaer Polytechnic Institute, Lawrence Livermore National Laboratory, University of Tennessee, University of California Berkeley, University of California Santa Barbara, General Electric, Systems Control Technology Inc., Integrated Systems Inc., MathWorks Inc. och Reasoning Systems.

Samarbetet med de engelska grupperna är speciellt gott. I England bedrivs ett motsvarande femårsprogram med en budget på 2.8M GBP av the Science and Engineering Research Council (SERC). I programet deltar grupper från bl a University of Manchester (UMIST), Cambridge University, Imperial College, Kingston Polytechnic, University College Swansea och University of Salford. Formella kontakter har etablerats mellan STU och SERC. En gemensam workshop "Graphical Front Ends for CACE" hölls på UMIST 14–18 juli 1986. Den var mycket givande med presentationer av olika projekt och ett ömsesidigt utbyte av idéer och tankar. En workshop med expertsystem som tema genomfördes den 6–10 april 1987 i Cambridge. Engelmännen har också deltagit i ett styrgruppsmöte och gjort kortare och längre besök.

Kontakt har knutits med och ett studiebesök har gjorts hos en grupp vid Central Electric Generating Board (CEGB). Vattenfall kan sägas vara Sveriges motsvarighet till CEGB. CEGB har en lång erfarenhet av att utveckla och använda CACE-program. Deras specialitet är detaljerad simulering av kraftsystemkomponenter. De har utvecklat effektiva och robusta numeriska rutiner för att göra detta. De har uttalat stort intresse för samarbete.

Samarbetet med Lawrence Livermore National Laboratory, Kalifornien har givit tillgång till preliminära versioner av EAGLES som är en grafikbaserad miljö med CACE-verktyg.

Vid Maryland Universitet har ett nytt forskningscentrum Systems Research Center (SRC) skapats. Idéen med centrat är att ha en bred tvärvetenskaplig verksamhet inom det systemtekniska området med nära samverkan mellan universitet och industri. Bland de forskningsprogram som bedrivs märks, intelligenta servomekanismer och kemisk processreglering som båda har element av CACE. Samarbetet med gruppen i Maryland är bra. Bl.a. har Andre Tits i Maryland utvecklat ett program Delight MaryLin som bygger på Intrac, som utvecklats på institutionen för reglerteknik, LTH i tidigare STU-finansierade projekt. Programvara för symbolisk problemlösning har bytts. Maryland har fått den programvara som utvecklats i projektet "Kombination av formelbehandling och numerik" (STU-projekt 85-4809) för analys av flervariabla system. De har givit program för analys av olinjära system och för automatisk generering av robotbanor. Odd Asbjörnsen har ett projekt för modellbygge av kemiska processer där CACE-ramprogrammets ansatser till systembeskrivningar och simulatorer skulle passa bra.

Gästforskare. Två utländska gästforskare har deltagit i CACE-ramprogrammet: Wolfgang Kreutzer, University of Canterbury, Christchurch, Nya Zeeland (2 månader) och Mike Denham, Kingston Polytechnic, UK (1 månad).