

Erhvervssamarbejde indenfor effektive og high performance algoritmer

Professor Jørgen Bang-Jensen
Institut for Matematik og Datalogi (IMADA)
Syddansk Universitet

NB: Dette materiale er under udvikling. Det må forventes at flere emner kommer på, efterhånden som kontakten til erhvervslivet intensiveres.

1 Hvem er vi?

- Professor, Dr. Scient Jørgen Bang-Jensen, leder af erhvervssamarbejdet. Forskningsområde: kombinatorisk optimering (netværksoptimering, grafalgoritmer, metaheuristikker).
- Professor Peter Kornerup. Forskningsområde: computerarkitektur og computeraritmetik.
- Lektor Joan Boyar. Forskningsområde: online algoritmer, approksimationsalgoritmer, kompleksitetsteori, kryptologi, GRID computing.
- Lektor Dr. Scient Edmund Christiansen. Forskningsområde: kontinuert optimering, matematisk modellering, numerisk analyse.
- Lektor Rolf Fagerberg. Forskningsområde: datastrukturer (specielt for ekstern memory og “cache oblivious” datastrukturer), databaser.
- Professor Kim Skak Larsen, institutleder. Forskningsområde: online algoritmer, datastrukturer, databaser, GRID computing.
- Lektor Klaus Meer. Forskningsområde: kompleksitetsteori, kontinuert optimering, neurale netværk og machine learning.

- Lektor og direktør for Nordic GRID, Brian Vinter. Forskningsområde: High performance computing and Networking (distribuerede systemer, GRID computing, operativsystemer, computer arkitektur).
- Adjunkt Lene Monrad Favrholt. Forskningsområde: online algoritmer og bioinformatik.
- Desuden har vi 5 Ph.D. studerende indenfor algoritmik.

Mere information om IMADA: <http://www.imada.sdu.dk>

2 Hvad vil vi?

Der er en lang tradition indenfor Dansk industri for udvikling af IT-systemer med vægt på funktionalitet, såsom administrative systemer og brugerinterfaces. Derimod er der langt mindre IT-udvikling, som fokuserer på effektive beregninger og high performance computing. Det er vores mål, at bidrage til at ændre dette ved at etablere en vidensbase indenfor effektive algoritmer og high performance computing.

Meningen er, at vi vil dække hele spektret af kompetencer, som kræves for at løse beregningsmæssigt svære og beregningstunge problemer, lige fra teoretiske analyser til egentlige implementationer på forskellige platformer, inklusive cluster computers og GRID systemer. Kun ved at integrere viden og erfaring fra alle de involverede områder, er det muligt at opnå de mest effektive, højeste kvalitets løsninger, som kombinerer effektive algoritmer til at producere høj kvalitetsløsninger med high performance systemer.

En hindring for teknologioverførsel fra universiteterne til erhvervslivet er ofte, at de problemer, der er relevante for erhvervslivet ofte er noget anderledes end dem forskerne arbejder med. Derfor er det svært for industrien, at se relevansen af nye forskningsresultater til deres aktuelle fokus på mere praktiske problemer.

Vi ønsker at bryde denne barriere med følgende fremgangsmåde. Vores første mål er at få etableret et problemkatalog indeholdende problemer, som industrien finder det vigtigt at få løst, enten pga konkrete anvendelser, eller fordi de ønsker at være forberedt på fremtidige anvendelser. Kataloget vil blive lavet i samarbejde med de interesserede virksomheder. Med dette katalog som udgangspunkt vil studerende på alle niveauer, bachelor, kandidat og Ph. D have mulighed for og vil

blive opfordret til at bidrage til en bedre forståelse og til løsning af disse problemer. Dette vil ske under vejledning fra en eller flere af IMADAs forskere, samt, når det er muligt, en vejleder fra det relevante firma.

Der vil være en øjeblikkelig teknologoverførsel gennem kommunikationen af resultaterne fra disse projekter. Yderligere vil en del af vores forskning blive drejet mod problemer af aktuel interesse for erhvervslivet. Endelig vil vores studerende, gennem muligheden for at arbejde med erhvervsrelaterede problemer, blive bedre kvalificerede til at arbejde i erhvervslivet og der kunne bidrage til at højne vidensniveauet i virksomheden.

For at kunne gøre dette erhvervsfremstød til en succes, er det vigtigt, at vi kan etablere en gruppe af Ph.D studerende (specielt erhvervsphd'er), som, under vejledning af relevante personer fra vores gruppe, kan arbejde på de problemer fra problemkataloget, der kræver en forskningsindsats. Gennem disse projekter og studerende vil vores forskere langt lettere kunne bidrage til løsning af erhvervsrelaterede problemer, end hvis de selv skulle sætte sig ind i problemet fra starten. Herved skabes inspirerende omgivelser for vores studerende og gennem de færdiggjaldede Ph.D. studerende vil den oparbejdede ekspertise spredes både til erhvervslivet og til andre universiteter i Danmark.

Da størsteparten af finansieringen af ovennævnte tiltag skal komme fra eksterne midler, er vi meget interesserede i at indgå samarbejdsaftaler med virksomheder, som kunne være interesserede i at ansætte en erhvervsphd, eller medvirke til finansieringen af post docs (forskningsassistenter med en Ph.D), der arbejder på problemer af direkte interesse for den pågældende virksomhed. Derigennem vil virksomheden bidrage til at oparbejde vigtig viden på området, samtidig med at den kan medvirke til uddannelse af højt kvalificeret arbejdskraft indenfor sit eget felt.

3 Eksempler fra de enkelte forskningsområder

Nedenfor gives eksempler på vores ekspertise og interesser indenfor de forskellige forskningsområder. Listen er repræsentativ, men ikke udtømmende.

3.1 Online-algoritmer

En algoritme er en "opskrift" på, hvordan man løser en bestemt type problem. Ofte er man nødt til at levere dele af løsningen, inden man kender hele problemet.

Da har man at gøre med et online-problem, som nødvendigvis må løses vha. en online-algoritme.

Eksempler på problemer, som er online af natur:

Pladsreservering Når man bestiller en pladsbillet i et tog, får man med det samme tildelt en bestemt plads i toget. Da man aldrig beder en passager om at skifte plads undervejs, betyder det, at man ofte får plads til færre passagerer, end man kunne, hvis man kendte alle bestillinger, inden man begyndte at fordele pladserne.

Bin Packing Bin packing er en meget generel betegnelse for problemer, hvor man ønsker at pakke nogle genstande i beholdere af en slags. Det kan være containere, lastbiler, dåser osv. Man kan være interesseret i at pakke en bestemt mængde af genstande i så få beholdere som muligt, eller man kan være interesseret i at pakke så mange genstande som muligt i et bestemt antal beholdere. Man kunne også være interesseret i at pakke 100 kg jordbær i så mange bakker som muligt, samtidig med at man sikrer, at der er mindst et halvt kilo i hver bakke. Hvis man ikke har lyst til at veje hvert enkelt jordbær og sortere dem efter vægt, inden man går i gang med at pakke, men bare vil håndtere dem et efter et i tilfældig rækkefølge, så har man et online-problem.

Planlægning af arbejdsopgaver Hvis man har en fabrik med produktionsmaskiner, er man sikkert interesseret i, at maskinerne bliver udnyttet så effektivt som muligt. Tilsvarende kan man have nogle computere, som skal udføre nogle opgaver. Man er normalt interesseret i, at opgaverne bliver udført så hurtigt som muligt - helst uden at man er nødt til at købe flere eller større computere, og hvis maskinerne ikke er ens, kan det have betydning, hvordan opgaverne fordeles på maskinerne. Kender man ikke alle opgaver i forvejen, er man nødt til at anvende en online-algoritme til at fordele opgaverne på maskinerne.

Paging Stort set alle computere har flere niveauer af hukommelse. Jo hurtigere en hukommelses-type er, jo dyrere er den typisk. Derfor har man en lille, meget hurtig hukommelse, en lidt større og lidt langsommere hukommelse osv. Det gælder selvfølgelig om, at de data, man bruger mest, er i den hurtigste hukommelse. Problemet er bare, at man ikke altid kan forudse, hvilke data man får brug for i fremtiden - dvs. det kræver en online-algoritme at beslutte, hvilke data man vil beholde i den hurtige hukommelse, og hvilke man vil smide ud for at få plads til andre.

Hvordan vælger man så den bedste algoritme til et givet problem? Først og fremmest må man vælge et kvalitetsmål. En mulighed er at måle algoritmerne på, hvor langt deres løsning kan risikere at ligge fra den bedst mulige løsning, som kunne opnås, hvis man kendte hele problemet fra starten. Det virker umiddelbart som en rigtig fornuftig måde at vurdere algoritmerne på. Men det viser sig, at man ofte får overraskende lidt information på denne måde. F.eks. kan dette mål ikke anvendes til at skelne mellem de mest kendte paging-algoritmer, selvom man ved, at de i praksis har meget forskellig performance. Derfor har en stor del af vores arbejde gået ud på at finde mere anvendelige kvalitetsmål og anvende dem på forskellige typer af online-algoritmer.

3.2 Kombinatorisk optimering, grafalgoritmer og metaheuristikker

Mange optimeringsproblemer som optræder i praksis er diskrete af natur og kan løses (enten optimalt eller "approsimativt") ved hjælp af teknikker, såsom heltalsprogrammering, strømning i netværk, matroidealgoritmer og metaheuristikker. Eksempler er skedulering af arbejdsopgaver, udførelse af transportopgaver med færrest mulige køretøjer, planlægning af kørselsruter for fejmaskiner, VLSI design, udvidelse af kapaciteten i telenetværk. Ved hjælp af metaheuristikker kan man ofte hurtigt opnå gode løsninger til en stor klasse af optimeringsproblemer (kaldet NP-hårde problemer), hvor den optimale løsning ikke kan forventes beregnet inden for den tid der er til rådighed. Da disse problemtyper optræder meget hyppigt i praksis er metaheuristikker et nyttigt værktøj.

Eksempler på tidligere projekter

- Bedre udnyttelse af kapaciteten i lysledernetværket til telefonsamtaler. Her var opgaven at finde en bedre måde til at rute telefonsamtaler på, således at specifikke krav om sikkerhed stadig var overholdt. Ved anvendelse metoder fra kombinatorisk optimering, lykkedes det at lave beregninger, der dels anviste en bedre måde at rute samtalerne på, dels viste at Fyns Telefon kunne vente mindst 5 år med at udlægge nye lyslederkabler på en bestemt strækning. Derved opnåede telefonvæsenet en betydelig besparelse.
- Planlægning af ruter for fejmaskiner, postomdeling, skraldafhentning, etc. En studerende har i sit speciale set på denne problemstilling, som kort går

ud på at finde f.eks. den mindste afstand, der skal tilbagelægges af en flåde af fejmaskiner, således at en forudbestemt delmængde af gaderne i en by alle bliver fejlet og den opsamlede mængde affald afleveret i et depot. Problemer af denne type tilhører klassen af NP-hårde problemer som omtaltes ovenfor. Man kan altså ikke håbe på at finde den absolut mindste afstand. Den studerende har derfor udviklet metoder, der tillader én at finde et system af ruter, som kan forventes at være rimeligt tæt på den bedste løsning. Hans resultater førte til et sæt af ruter, der var væsentligt bedre end dem man aktuelt anvendte i Fredericia. I dette speciale anvendte den studerende også metoder fra kombinatorisk optimering, avancerede datastrukturer, databaser med mere.

- Udskæring af plader i skibsproduktion. Dette projekt er lavet i samarbejde med Lindøværftet. Problemet, der er helt klassisk indenfor området, gik ud på at finde en rækkefølge at skære plader af givne længder (alle med samme bredde) ud fra stålplader med den samme bredde og en fast (stor) længde. En af vores specialestuderende udviklede en metaheuristik, baseret på genetiske algoritmer, som kunne finde en god rækkefølge at foretage udskæringerne i. Metoden er senere taget i anvendelse af værftet, som derved har opnået en pæn besparelse i materialeforbruget.

Eksempler på problemstillinger som vi er interesserede i at samarbejde omkring indenfor kombinatorisk optimering

- Ruteproblemer, f.eks. planlægning af ruter/køreplaner for en flåde af køretøjer, finde det mindste antal køretøjer der skal bruges for at levere en givet ordre. Finde approksimativt korteste ruter, f.eks. i computerspilanvendelser.
- Problemer hvor man søger at øge kapaciteten i et eksisterende (kommunikations-)netværk, f. eks. indenfor (mobil-)telefoni, til en given kapacitet (f.eks. målt ved en minimum punkt til punkt kommunikation på k enheder) og gøre dette så billigt som muligt.
- Skeduleringsproblemer, f.eks. skemalægning for skoler/universiteter og lignende, planlægning af rækkefølge for udførelse af arbejdsprocesser i et større projekt.
- Problemtyper der kan modelleres ved hjælp af grafteori/matroideteori og problemer der kan formuleres som lineære heltalsprogrammeringsproble-

mer. Dette dækker en lang række praktiske optimeringsproblemer, foreksempel pakning af containere, mandskabsplanlægning i fly og tog, beregning af det mindste antal servicemedarbejdere der kræves for at løse en givet opgave, som involverer tidsvinduer.

3.3 Kontinuert optimering, matematisk modellering og numeriske beregninger

Overskriften dækker over den del af anvendt matematik og matematisk modellering, som ligger uden for diskret matematik. Det vil først og fremmest sige problemer, som formuleres inden for rammen af følgende emner:

1. lineære og ulineære ligningssystemer, evt. løst online
2. ordinære differentiaalligninger, begyndelsesværdi- og randværdiproblemer
3. partielle differentiaalligninger, begyndelsesværdiproblemer, især diffusionsproblemer, og randværdiproblemer, især statiske beregninger
4. integralligninger

På instituttet har vi god erfaring med projekter fra den tekniske og naturvidenskabelige del af erhvervslivet. Vi er også interesseret i projekter fra den finansielle sektor, hvor vi har et godt samarbejde med Institut for Regnskab og Finansiering på Syddansk Universitet.

Eksempler på tidligere projekter

- Numerisk løsning af varmeledning ligningen anvendt til opdagelse af små utætheder i vandinstallationer. Samarbejde med Felson.
- Sammenhæng mellem tryk, temperatur og adsorption i et lukket termodynamisk system. Samarbejde med Danfoss.
- Beregning af strømning og tryk i et rørsystem. Kursusprojekt uden samarbejdspartner.
- Spline-repræsentation af kurver på digitale kort. Samarbejde med Rambøll.

- En matematisk model for vækst af pellets (svampekugler) i en næringsvæske. Hvis kuglerne bliver for store i forhold til koncentrationen, kommer der ikke næring til centrum, således at der udvikles et dødt hul. Derved bliver processen mindre effektiv. Samarbejde med Institut for Biokemi og Molekylærbiologi, som samarbejdede med Novo.
- En matematisk model for menneskets brystkasse (thorax) udsat for trykbølger. Samarbejde med forsvaret.
- Nedhæng af højspændingsledninger. For at beregne nedhæng og træk i højspændingsledninger må man løse et ulineært ligningssystem. Gammelt problem fra Jysk Telefon, som bliver brugt som kursusprojekt.
- Computer-registrering og -overvågning af nyrepatienter under langtidsbehandling. Samarbejde med Odense Universitetshospital.
- Afsløring af kaos i personers hjerterytme med henblik på prognostisering af patienter med akut hjerteinfarkt. Samarbejde med Biokemisk Institut og Odense Universitetshospital.
- Iltoptagelse i gæller. I fiskenes gæller passerer vand og blod hinanden i modstrøm, og herunder udveksles ilt ved diffusion. Hvor stor bliver iltkoncentrationen? Samarbejde med Biologisk Institut.
- Computerovervågning og -styring af en kemisk proces til afgiftning af spildevand. Samarbejde med med NKT i Middelfart.
- Brudstadieregning af et strukturelt element. Samarbejde med et konsulentfirma.

En mere udførlig beskrivelse af projekterne kan findes på de følgende links:

<http://www.imada.sdu.dk/~edc/Bachelor/>

<http://www.imada.sdu.dk/~edc/Specialer/>

3.4 Algoritmer for store datamængder

I mange situationer arbejder man med datamængder som er langt større end størrelsen af computerens RAM. Eksempler inkluderer meteorologiske data, finansielle data, telekommunikationsdata, data fra geologi, fysik og astronomi, geografiske informationssystemer (GIS), samt computergrafik. I mange tilfælde arbejder

man med datamængder i Terabyte (10^{12}) og endda Petabyte (10^{15}) klassen, mens de fleste computers RAM er omkring en Gigabyte (10^9). At flere og flere data i vore dage genereres automatisk af computeriserede enheder er med til at øge forekomsten af så store datasæt.

I sådanne situationer er flaskehalsen for beregninger ikke selve beregningsarbejdet, men derimod tiden som går med at transportere data mellem RAM og harddisk. Ved at bruge algoritmer som er designet til at minimere denne transport, kan betydelige besparelser i køretid opnås.

På instituttet har vi stor ekspertise i at designe sådanne algoritmer, og har gennem både teoretiske og praktiske resultater vist betydningen af denne tilgangsvinkel. Blandt de praktiske resultater kan nævnes datastrukturer for databaser med væsentligt forbedrede køretider for indsættelser af elementer, samt nye algoritmer for sortering som slår klassiske algoritmer på grund af mere effektiv datatransport.

Vi er interesserede i at samarbejde med virksomheder såvel som forskningsenheder som bearbejder datamængder af størrelsen Gigabyte og opefter.

3.5 Effektive realisationer af beregningstunge problemer:

Indenfor real-tids signal- og billedbehandling indgår ofte beregningstunge delopgaver, hvor data skal forbehandles, filtreres, transformeres og/eller komprimeres. Disse beregninger er som oftest meget simple, men skal udføres meget hurtigt og udgør tit en flaskehals. Det er selvfølgelig væsentligt at de algoritmer der skal udføres, er tilpassede til den konkrete opgave, og er optimale hertil. Her er i sig selv en opgave i at analysere problemet og finde passende problemtransformationer og algoritmeformuleringer som kan minimere tidsforbruget. Men dette er ofte ikke tilstrækkeligt til at opgaverne kan løses med den krævede hastighed med dagens processor teknologi. Udover processorer som "hyldevarer" i form af standard PC'er, findes der specielle signalbehandlingsprocessorer, der er optimerede med henblik på nogle af de beregningsprimitiver, som er fælles for mange af disse applikationer.

En anden oplagt mulighed for at effektivisere udførelsen af et program er at isolere det beregningstunge delproblem, og lave en hardware implementation af dette, dvs. et kredsløb realiseret som en VLSI-chip. Her kan man langt bedre end i sædvanlig software udnytte fx. muligheder for at gøre noget parallelt med noget andet, og at gøre præcis det der er behov for og intet overflødigt. En sådan chip kan så monteres som en co-processor i fx. en PC, og derved understøtte det egentlige applikationsprogram. Men design og produktion af en sådan chip kræver at der er behov for meget store stykantal for at dette er rentabelt.

Der findes imidlertid i dag en alternativ teknologi, i form af såkaldte FPGA kredse (Field Programmable Gate Arrays). Disse består af et meget stort antal “gates”, dvs. simple kredse, som kan “programmeres” til at udføre en specifik simpel opgave, og indbyrdes kan forbindes til at udgøre et større logisk kredsløb. Denne programmering kan til enhver tid laves om, og kredsen kan endog dynamisk ændres til at udføre en ny beregning, hvis der måtte blive behov herfor. Antallet af “gates” på disse FPGA kredse er så stort, at der ligefrem er plads til også at lægge hele PC’ens hardware ned på kredsen, og dog have plads til at den også kan indeholde et ekstra kredsløb til løsning af det specielt beregningstunge delproblem. Der findes udmærkede værktøjer til udvikling af disse “programmer”, ligesom der er omfattende biblioteker med løsninger til delproblemer som kan indkorporeres, herunder også førnævnte total-designs af hele standard-processorer.

Eksempler på tidligere projekter

- Design af et system til støjfiltrering af talesignaler ved telefonsamtaler: Telefonsamtaler er ofte behæftet med en del støj som man ønsker at fjerne, herunder hyletoner der opstår ved tilbagekobling, når en abonnent har medhør på. Opgaven var formuleret af det daværende Fyns Telefon, som ønskede et system til at forbedre arbejdsmiljøet for telefonisterne. Den studerende udarbejdede en software realisation, der faktisk kunne foretage den ønskede filtrering, samt et oplæg til en hardware realisation, der ville kunne løse problemet i praksis.
- Hastighedsforbedring af algoritmer til kryptering: I forbindelse med et PhD-projekt ved Århus Universitet, i samarbejde med det daværende Jysk Telefon, blev der udviklet en forbedret version af den såkaldte RSA algoritme til kryptering. Denne algoritme indgår i de fleste systemer til sikring af digitale transmissioner, digitale underskrifter, mm., men kræver et meget stort antal beregninger på tal med flere hundrede decimale cifre (binært på fx. 512, 1024 eller endnu flere bit), idet krypteringssikkerheden stiger med antallet af cifre. Algoritmen blev af den PhD-studerende senere udlagt som et VLSI design, og et mindre antal chips blev fremstillet. På daværende tidspunkt var det den hurtigste enkelt-chip realisation af RSA algoritmen, den kunne kryptere/dekryptere en ukomprimeret ISDN digital telefonsamtale på 64 Kbits/sek. Senere er flere endnu bedre RSA algoritmer blevet udviklet, herunder en som tillader et meget simpelt VLSI design.

- **Signalbehandling:** Et formål med mange billedbehandlingsopgaver er at genkende objekter, at sammenligne scener eller klassificere billeder. Et værktøj hertil kan være beregninger af såkaldte diskrete momenter, som ved deres definition kræver et meget stort antal multiplikationer på værdierne af de enkelte billedpunkters lysværdier (pixelvalues). Vi har udviklet algoritmer, der erstatter disse med et væsentligt mindre antal additioner, og således kan reducere beregningsarbejdet betragteligt. Algoritmen kan ydermere realiseres ekstremt simpelt i hardware, dvs. det vil være meget simpelt at understøtte sådanne billedbehandlingsopgaver med en co-processor i FPGA teknologi, jfr. ovenfor omtalt.

3.6 High Performance Computers and Networks (HPCN)

High Performance Computers and Networks, HPCN, er det værktøj som skal muliggøre overgangen til e-Science som pt. har meget høj bevågenhed i vores nabolande. Begrebet e-Science dækker over mange videnskabs og ingeniør felter hvor man ønsker at erstatte laboratorieforsøg med computer modelleringer og simuleringer felter hvor e-Science er aktivt inkluderer kemi, biologi, miljø, FEM, CFD samt mange andre.

HPCN gruppen arbejder med en række aktiviteter til at muliggøre e-Science, fra design af supercomputere, specielt klynge-computere, med optimale egenskaber til de ønskede beregningsområder, net-protokol design om implementering til at overkomme begrænsningerne i TCP/IP, distribueret delt lager, operativsystemer, GRID computing og parallelisering af applikationer.

Eksempler på tidligere projekter

- Horseshoe, Danmarks kraftigste computer, har hjemme på SDU og er designet og implementeret af HPCN gruppen.
- Tuple-based distributed memory; udviklingen af PastSet er et eksempel på et fuldt funktionelt distribueret delt lager som tillader programmører meget simpel tilgang til distribuerede resurser. PastSet er senere blevet kommercialiseret.
- Message Passing Interface, MPI, er en standard for programmering af store parallel-maskiner, inklusive klynge-computere. HPCN gruppen udviklede en række nye algoritmer til mere effektiv udnyttelse af netværkslaget i sådanne maskiner og leverede en MPI implementering der er målt så meget

som 13 gange hurtigere end de bedst kendte versioner. MPI implementationen er senere blevet kommercialiseret.

- OpenMP er en standart, der tillader simpel udvikling af parallelle applikationer ved simple sproglige udvidelser. OpenMP antager et underlæggende delt-lager og findes derfor kun til klassiske SMP arkitekturer. HPCN gruppen har udviklet metoder, der tillader OpenMP kompilering til distribuerede hukommelses arkitekturer, fx klynge-maskiner. Metoderne fungerer også på delt hukommelse og er målt til 26 gange hurtigere end OpenMP fra Intel på sådanne maskiner. OpenMP versionen er under kommercialisering.
- PATHS er et projekt der muliggør meget høj-opløselig profilering af kommunikationen i en applikation, hvilket muliggør detaljeret optimering af applikationer. Teknologien i PATHS er blevet inkluderet i produktet VAMPIR fra Pallas GMBH (nu Intel).
- GRID front-end er et projekt, der gør det muligt at koble et system til GRID (NordGRID mere specifikt), uden at installere GRID middleware på systemet, men i stedet repræsenterer front-end systemet maskinen på GRID. Front-end systemet er i operation på flere af de største supercomputere i Norden og er basis for den kommende GRID for Dansk Center for Scientific Computing.
- GRID BLAST. Applikationen BLAST, som er en flaskehals i megen bio-forskning, er blevet paralleliseret til at køre på løst koblede systemer (GRID). Løsningen er meget effektiv og har demonstreret reduktion af køretiden fra døgn til minutter ved tilstrækkeligt antal CPU'er. GRID BLAST er senere kommercialiseret.
- DALTON Portal. En applikations specifik GRID portal der skjuler GRID for brugeren.
- GRID VCR. Det første kendte eksempel på et GRID instrument. Med GRID VCR er det muligt for en GRID bruger at optage en udsendelse fra TV og få resultatet leveret som en MPEG film. Meget populært blandt brugere der ikke har tilgang til dansk TV.

3.7 Polynomial systems solving

Many problems in real life involve polynomial systems over real or complex numbers. For example, in engineering the design of certain mechanisms often is closely related to solving a corresponding polynomial system. Think about the construction of a garage door that has to move from the closed position to the open one. The movement of the door from one of these two positions to the other of course has to satisfy certain requirements; for example, during the motion process the door cannot use the space where a car is parked. One special mechanical structure with which such a task can be accomplished is the so called Stephenson mechanism. Determining the correct measures of the different parts (angles, lengths of linkages etc) of such a mechanism typically leads to a polynomial system of equations.

From the theoretical point of view, computing all (or even a single) solutions of such a system can be difficult already in low dimensions. There are basically two families of algorithms attacking such questions: symbolical and numerical algorithms. Among the latter we are working with homotopy methods. These are numerical path following algorithms based on a (deep) understanding of the solution structure of polynomial systems. In case of Stephenson (or easier) mechanisms, such homotopy methods have given a complete solution catalogue.

3.8 Optimization

We are interested in all kinds of optimization problems, in particular linear and convex programming. Here, for the last 15-20 years interior point methods IPM have become more and more important and currently are even more than just an alternative to the well known Simplex method. Theoretically, IPMs have a polynomial running time that outperforms the Simplex method in a worst case scenario. This behaviour as well is reflected when using IPMs in practice. We have applied successfully interior point algorithm for fault detection tasks in geodesy.

3.9 Neural Networks and machine learning

Among alternative models of computation Neural Networks play an important role in relation to problems in artificial intelligence. They are the main model for questions related to computational learning tasks. We are interested in the study of different network models and their use in learning theory. Depending on the

precise model used different techniques turn out to be important. Examples are numerical steepest descent methods (used in the well known Backpropagation algorithm), convex optimization (main problem to solve using Support Vector Machines), Markov processes (related to statistical processes modeled through Boltzmann machines or Hopfield networks) and many more. Our interest is in using such techniques and models in relation to task like image analysis, pattern recognition, data analysis, supervised learning etc.