

Grafikkort sætter Newton på speed

Hansen, Jesper Schmidt; Bailey, Nicholas; Dyre, J. C.; Schrøder, Thomas

Published in:
Aktuel Naturvidenskab

Publication date:
2013

Document Version
Også kaldet Forlagets PDF

Citation for published version (APA):
Hansen, J. S., Bailey, N., Dyre, J. C., & Schrøder, T. (2013). Grafikkort sætter Newton på speed. *Aktuel Naturvidenskab*, 2013(1).

General rights

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.

Take down policy

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



Foto: Carsten P. Kjaer.

Grafikkort sætter

Vores pc'ere gemmer i dag på grafikkort, der for de bedstes vedkommende byder på en enorm regnekraft set i forhold til prisen. Denne regnekraft udnytter forskerne nu i stor stil.

Forfattere



Jesper S. Hansen
jschmidt@ruc.dk



Nicholas P. Bailey



Jeppe C. Dyre



Thomas B. Schrøder

Grundforskningscentret for seje væskers dynamik "Glas og Tid", IMFUFA, Institut for Natur, Systemer og Modeller, Roskilde Universitet

Verden over forsøger videnskabsfolk at løse stadig mere komplicerede problemstillinger, hvor krævede computerberegninger er nødvendige. Desværre er udviklingen af computer-processorens regnekraft nu stort set gået i stå, og man anvender derfor flere processorer pr computer (også kaldet kerner). Grafikkortet har taget denne strategi til det ekstreme og er i rivende udvikling; grafikkort tilbyder i dag en regnekraft mange gange hurtigere end standard-computeren. Videnskabsfolk har dermed fået adgang til billige regneressourcer, hvilket kan flytte forskningen og vores forståelse af naturen.

Computerspil kræver meget stor regnekraft: Spilleren overflyver komplicerede og detaljerede landskaber, affyrer projektiler og forårsager eksplosioner – alt sammen med krævede grafik der løbende skal beregnes. Til dette benyttes computerens grafikkort, eller med mere teknisk sprog: Graphics-Processing-Unit (GPU).

En GPU er opbygget af simple men mange mikroprocessorer. Moderne grafikkort har 1.500 eller flere processorer indbygget – og det er til en pris på omkring 3.000 kr., eller cirka to kr. pr processor! Man kan – uden at der i øvrigt kan sammenlignes direkte – nævne, at Danmarks Meteorologiske Instituts to supercomputere hver har 512 standard-processorer, Central-Processing-Unit (CPUer) med hver fire kerner, til en pris på omkring 20 mio. kr. pr supercomputer.

Algoritmerne bag det hele

Desværre kan man ikke blot tage sit favoritprogram ned fra hylden og få grafikkortet til at lave beregningerne. Softwaren skal være programmeret til at udnytte grafikkortet. Den gode nyhed er, at det kan mange af de populære videnskabelige programmer i dag, eksempelvis MATLAB.

Da videnskabsfolk for godt 10 år siden begyndte at

Artiklen kommer fra tidsskriftet *Aktuel Naturvidenskab*: aktuelnaturvidenskab.dk

anvende grafik kort, var det anderledes. Programmeringen var ganske besværlig, og man skulle "snyde" grafik kortet til, at tro at fysiske størrelser som en partikels hastighed og position, var pixels på en skærm.

I 2006 lancerede den førende grafik kortproducent NVIDIA en meget nemmere måde hvorpå kommunikationen med grafik kortet kunne foregå – det såkaldte interface mellem programmør og grafik kort. NVIDIAs programmeringsmodel blev døbt CUDA (Compute Unified Device Architecture). Senere er en åben standard model, kaldet openCL, blevet udviklet, der skal gøre det muligt at regne på alle typer grafik kort.

De finere detaljer i programmeringsmodellerne er komplicerede, og skal man have fuldt udbytte af grafik kortet, skal disse detaljer kendes godt når man selv programmerer sin software. Derfor er programmerings erfaring og hardware forståelse nyttig her. På den anden side er det nu muligt for ikke-eks-

perter at udnytte de fleste muligheder GPU'en tilbyder og opnå betydeligt hurtigere beregningen med en ret lille indsats.

Er det besværet værd?

Dette spørgsmål kan ikke uden videre besvares generelt med et "ja" eller et "nej": Der skal det rigtige redskab til at løse et problem. Skal man fx gange to matricer sammen (beregninger der bl.a. anvendes i signalanalyse), vil beregningerne ofte være hurtigere at foretage på en helt almindelig CPU, hvis matricerne ikke er store. Er matricerne derimod store, vil beregningen på grafik kortet være mange gange hurtigere. Faktisk kan GPU'en være omkring 100 gange hurtigere end CPU'en for tilstrækkeligt store matricer – og det er uden de store anstrengelser ved programmeringen af koden (se boks).

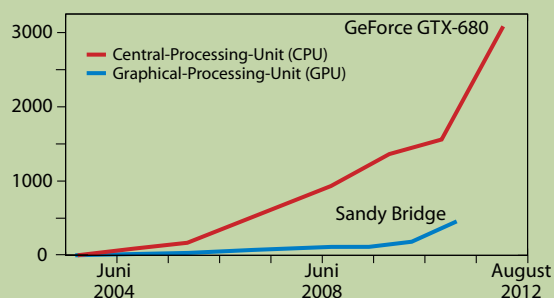
Ved grundforskningscentret "Glas og Tid" har vi siden 2008 benyttet NVIDIA grafik kort og CUDA-programmeringsmodellen.

Newton på speed

Udvikling af grafik kort

Udviklingen af grafik kort (GPU) går i dag meget hurtigere end udviklingen af standard CPU. GPU'en er specielt velegnet til at regne i enkeltpræcision, dvs. tal med op til 7 betydende cifre. Dette er en acceptabel præcision for de fleste videnskabelige beregninger.

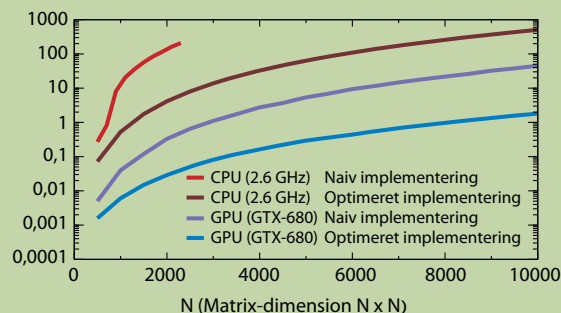
GFLOP/s - teoretisk estimat



Sammenligning af ydeevne for Intels flagskibe inden for standard CPU-processorer og NVIDIAs GPU-processorer. GFLOP/s betyder "Giga-Floating-Point-Operations-per-second", altså milliarder decimaltalsoperationer pr sekund. Jo flere GFLOP/s jo hurtigere er processoren. "Sandy Bridge" er pt. Intels high-end CPU-serie, og GeForce GTX-680 er NVIDIAs hurtigste spille-baserede grafik kort.

Kilde: NVIDIA - CUDA programming guide.

Eksekveringstid i sekunder



Figuren viser hvor lang tid det tager at gange to matricer sammen; bemærk den logaritmiske skala på y-aksen. Det "naive" CPU-program er skrevet i ISO-C99, det optimerede program benytter Basic Linear Algebra Subprograms (BLAS). Den naive GPU-kode er en meget simpel modifikation af den naive CPU-kode, som ikke benytter grafik kortets hurtigere hukommelse osv. Den optimerede GPU-kode benytter CUDA versionen af BLAS (CUBLAS). Det ses, at selv et simpelt GPU-program er cirka ti gange hurtigere end et fuldt optimeret CPU-program.

Newton's bevægelsesligning – Newtons 2. lov

- er givet ved

$$m_i \frac{d^2 r_i}{dt^2} = F_i,$$

hvor m_i er massen af partikel i , r_i dens stedvektor, og F_i er kraften på den fra de andre partikler i systemet. Newtons 2. lov bestemmer molekylernes bevægelse, når man foretager molekyledynamik-simuleringer; den er altså motoren bag det hele. For systemer med mange partikler er ligningen et højdimensionalt koblet differentiaalligningssystem, der ikke kan løses analytisk.

Man må derfor ty til numeriske metoder, altså til computersimuleringer. Når det sker, diskretiseres ligningssystemet med hensyn til tiden. I molekyledynamiske simuleringer er et diskret tidskridt typisk cirka 1-10 femto-sekunder (et femto-sekunder er 10^{-15} sekund).

Vi har udviklet specialdesignede algoritmer, som udnytter de forskellige muligheder grafikortet tilbyder. Vi har omkring 90 grafikort i vores "super-computer", hvilke giver en beregningskapacitet på den pæne side af 90 TeraFLOP/s, dvs. 90×10^{12} decimalberegninger (FLOP: "floating point operations") pr sekund.

Asfalt og molekulære simuleringer

Har det været besværet værd? Bestemt! Overgangen til brug af grafikort var et kvantespring for vores forskning, idet vi med ét kunne lave 20 gange hurtigere simuleringer. For eksempel benytter vi grafikort til at forstå seje væskers dynamik, nanovæsker, og til simulering af molekylemodeller af raffineret bitumen (den sorte tjæredel i asfalt). Studierne er baseret på molekyle-dynamiske beregninger, hvor man numerisk løser Newtons bevægelsesligning, dvs. et stort sæt koblede differentiaalligning – typisk i størrelsesordenen 6.000-30.000 styk – der skal løses over den tid, som er karakteristisk for det problem, man studerer.

Seje væsker er karakteriserede ved en meget langsom dynamik. I "Glas og Tid" og i projektet "From Micro to Nano" opstiller vi modeller for disse væsker, som studeres ved computer-simuleringer af molekyledynamik. For at kunne teste vores teorier er det nødvendigt at "ramme" den interessante dynamik, og derfor skal simuleringerne foretages over en (for molekyledynamik) enormt lang tidsskala. Grafikortet gør os i stand til dette.

Computersimuleringer indgår også som en vigtig del af det strategiske forskningsprojekt Cooee.

Coee-projektets formål er at etablere det videnskabelige grundlag for en reduktion af rullemodstanden (modstanden mellem dæk og vej), herunder at forstå ændringerne i de mekaniske egenskaber af bitumen efterhånden som en vej ældes. Bitumen kan bestå af mere end en million forskellige typer molekyler. Disse grupperes på baggrund af deres opløselighed i organiske opløsningsmidler.

Ud fra en sådan klassificering har vi opstillet en molekylemodel bestående af fire strukturer, der hver repræsenterer én gruppe. Vi studerer fænomener på en tidsskala svarende til omkring 2 mikrosekunder, hvilket er den karakteristiske tidsskala for molekylernes rotation.

På basis af resultaterne fra computer-simuleringer foretaget inden for det seneste år har vi identificeret en afgørende årsag til ændring af asfalts mekaniske egenskaber med tiden. Dette kunne ikke have været gjort med CPU-baserede simuleringer, da de kun kan foretage beregninger af meget kortere tidsrum på cirka 10 nanosekunder; på så kort en tidsskala kan mange egenskaber ikke studeres.

Ikke kun "Newton er på speed"

Grafikort benyttes ikke kun til løsning af Newtons bevægelsesligninger. Eksempler på andre anvendelser er Computational Fluid Dynamics (CFD), optimeringsproblemer og signalanalyse. Det første interface med grafikortet var baseret på programmeringssproget C. I dag findes der interfaces til fx sprogene Python, Java og Fortran, så man nu kan programmere software, der kan køre på GPU-kort i sit favoritsprog.

Mange softwarepakker understøtter faktisk GPU-beregninger, uden at brugeren skal have kendskab til GPU-programmering (som nævnt ovenfor er MATLAB et eksempel herpå) – faktisk kan brugeren være helt uvidende om, at programmet benytter grafikortet. Grafikortets ressourcer er dermed nu tilgængelig for alle, selvom man stadig selv må programmere nede "på bit-niveau", hvis den allerhurtigste kode skal opnås. ■

Et moderne gerafikkort repræsenterer en betydelig regnekraft.

Foto: Thomas Schröder



Artiklen kommer fra tidsskriftet *Aktuel Naturvidenskab*: aktuelnaturvidenskab.dk

Cooee-projektet

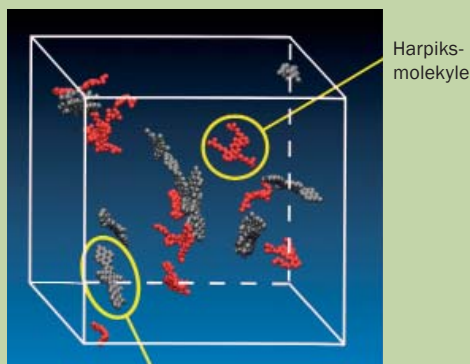


I projektet undersøges eksperimentelt hvilke parametre, der bestemmer en vejs rullemodstand. Ud fra målinger af en vejs tekstur opstilles matematiske modeller for rullemodstanden. Billedet viser en såkaldt TUG-trailer til måling af rullemodstand. Cooee-projektet har bl.a. til forskningsformål udført rullemodstands-målinger på landingsbanen for den nu nedlagte flyvestation Værløse.

Foto: Niels Dujardin

I bitumen (den sorte klistrede del af asfalten) vil de største af molekylerne (asfaltenerne) med tiden danne nano-aggregater. Disse kan muligvis medføre nedsat vedhæftning til stenene i asfalten og derved give et dårligere vejgreb. På baggrund af resultater fra molekyledynamik-simuleringer mener vi, at man kan kontrollere formationen af aggregater ved fx at tilsætte harpiks. Simuleringerne af bitumen er foretaget med "Glas og Tid"s molekyledynamik-pakke RUMD designet og optimeret for GPU-computing (<http://rumd.org>).

Coee-bitumen modellen



Asfaltmolekyle

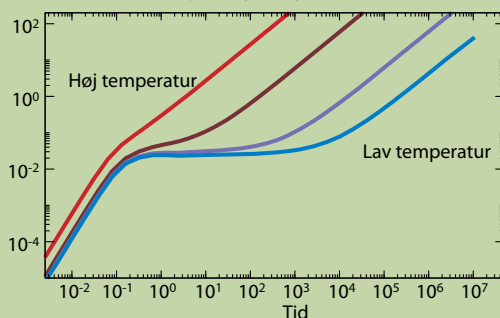
Snapshot af Coee-bitumen modellen i en molekyledynamiksimulering. Kun asfaltenmolekylerne og harpiksmolekylerne er vist her. I alt simuleres et par hundrede molekyler.

Kassen angiver simuleringsboksen, dvs. størrelsen af hele systemet.

Seje væsker

Figuren viser kvadratet af den gennemsnitlige afstand (middelvejs-forskydningen) et molekyle har bevæget sig

Kvadratet af middelvejsforskydningen ($\langle \Delta r^2(t) \rangle$)



som funktion af tiden. Tid og afstand er angivet i karakteristiske simuleringsenheder. Afstandsenheden svarer til ca. 1 Ångstrøm, tidsenheden til ca. et picosekund. I seje væsker er det specielt overgangen (angivet ved plateauet) fra den såkaldte ballistiske bevægelse ved meget korte tider til den diffusive bevægelse ved længere tider, der er interessant.

Med GPU-baserede simuleringer kan vi studere denne overgang i betydelig længere tid end med CPU-baserede simuleringer. Ved den laveste temperatur kommer systemet først i ligevægt efter tre måneders simulering. Hvis en tilsvarende simulering var foretaget på en CPU-baseret computer, ville det have taget omtrent seks år med den hurtigste CPU, man har til rådighed i dag!

Coee-projektet

Projektet er støttet af Det Strategiske Forskningsråd, og er et samarbejde mellem "Glas og Tid" (RUC), Vejdirektoratet, DTU samt entreprenør-virksomheden NCC. Projektet "From Micro to Nano" er endvidere støttet af Lundbeckfonden (projekt R49-A5634).

Videre læsning

J. Sanders og E. Kandrot, CUDA by Example: An Introduction to General-Purpose GPU Programming, NVIDIA Corporation (2011).

R. Tsuchiyama et al., Programming Book, Fixstarts corporation (2010).

"Glas og Tid" hjemmeside: <http://glass.ruc.dk>.

Coee-projektets hjemmeside: <http://www.coee-co2.dk>.

B. Schmidt og J. C. Dyre, CO₂ emission reduction by exploitation of rolling resistance modelling of pavements, *Procedia* 48, 311 (2012); J. S. Hansen et al., Four-Component united-atom of bitumen, *J. Chem. Phys.* (i trykken).

L. Bøhling et al., Do the repulsive and attractive pair forces play separate roles for the physics of liquids?, *J. Phys.: Condens. Matter* 25, 032101 (2013); T.S. Ingebrigtsen et al., What is a simple liquid?, *Phys. Rev. X* 2, 011011 (2012).