Computerarchitectuur

Niet-grafische toepassingen voor GPU's

ofwel ***GPGPU***

Door:
Haroen Viaene

1ste fase bachelor Elektronica-ICT

2014-2015

Inhoud

Inhoudsopgave

1 Inleiding

Omdat onze grafische kaarten alsmaar groter worden, lijkt het logisch om nieuwe toepassingen te zoeken voor deze kaarten. Snel denk je al aan de ontdubbeling van de CPU, omdat die vaak onder grotere stress staat dan de GPU, maar dit is niet altijd zo haalbaar, omdat grafische kaarten niet vast op het moederbord zitten, maar in bijvoorbeeld een PCI-poort gestopt kunnen worden. CPU's echter moeten in een speciaal daarvoor gemaakte socket gestopt worden. Het is ook moeilijker voor de gemiddelde end-user om een CPU te vervangen. Daardoor zijn GPU's makkelijker up te graden. Het probleem is echter dat normaal gezien enkel videobewerkingen uitgevoerd worden in een GPU. GPGPU (het uitvoeren van standaard-taken op een GPU of General Processing on Graphics Processing Units) maakt gebruik van het grote aantal processen die tegelijk (maar op een lagere snelheid) in een GPU kunnen uitgevoerd worden. Een interessante manier om dit te vergelijken is om de CPU te zien als een team van 10 wetenschappers die snel heel moeilijke operaties kunnen uitvoeren. De GPU kan je dan vergelijken met een leger van 1000 man. Een individuele soldaat zal niet goed zijn in bijvoorbeeld het uitrekenen van een integraal. Maar als je de taak opdeelt in 1000 kleine optellingen, zullen ze waarschijnlijk zelfs sneller zijn dan de wetenschappers omdat ze met zo veel zijn. Bij GPGPU heb je dus nood aan heel specifiek opgedeelde taken, maar als je dat zo kan doen en het efficiënt is om die berekening op te delen, zal dit een zeer grote versnelling opleveren, die makkelijk te vermeerderen is door het aantal GPU's te verhogen.[1]

2 Wat is GPGPU?

GPGPU, ofwel General-Purpose Computing on Graphics Processing Units is het uitvoeren van gewone rekenopdrachten op de grafische kaart. Als je aan GPGPU-computing wil doen, zal je dit waarschijnlijk in OpenCL doen (of Nvidia's CUDA). GPGPU is nog niet zo heel lang mogelijk omdat tot vanaf 2001 grafische kaarten nog niet zo heel geavanceerd waren. Dankzij het geavanceerder worden van videospelletjes, videobewerkingen, gecombineerd met het daardoor performanter worden van grafische kaarten maakte het haalbaar om deze ook te gebruiken voor andere taken.[2]

3 Opkomst van GPGPU

In 2001 werd het mogelijk om floating-point- en shader-berekeningen uit te voeren op een GPU. Dit betekende de opkomst van GPGPU, omdat nu ook toepassingen die niet enkel gebruikt worden voor het tonen van beelden op een scherm, maar ook echte berekeningen mogelijk werden. Dit werd bewezen door het berekenen van matrixvermenigvuldigingen in 2001, dat voor het eerst sneller op een GPU uit te voeren was dan op een CPU. Na deze eerste toepassingen van GPGPU werden er frameworks zoals OpenGLen Microsoft's DirectX geschreven om gebruik van de vooruitgangen te kunnen maken in ander programma's. Later kwam NVidia met CUDA op de proppen, gevolgd door de meest recente Apple/Khronos OpenCL en Microsoft's DirectCompute.

4 Voordelen van GPGPU

Omdat grafische kaarten van nature uit zich al parallel gedragen en grafische kaarten makkelijk verdubbelbaar zijn (door bijvoorbeeld twee grafische kaarten te installeren), zijn grafische kaarten een stuk flexibeler dan CPU's. Dit wil zeggen dat vooral bij lange berekeningen en dingen waar veel berekeningen tegelijk moeten gebeuren het zeer handig is om aan GPGPU te doen. In andere secties wordt er specifieker toegespitst op een aantal voorbeelden waarbij het handig is om die rekenopdrachten uit te voeren op grafische kaarten. Typische voorbeelden waarbij GPGPU handig is, zijn wiskundige berekeningen, cryptografie en ander wetenschappelijk onderzoek.

5 Toepassingen van GPGPU

5.1 Wiskunde

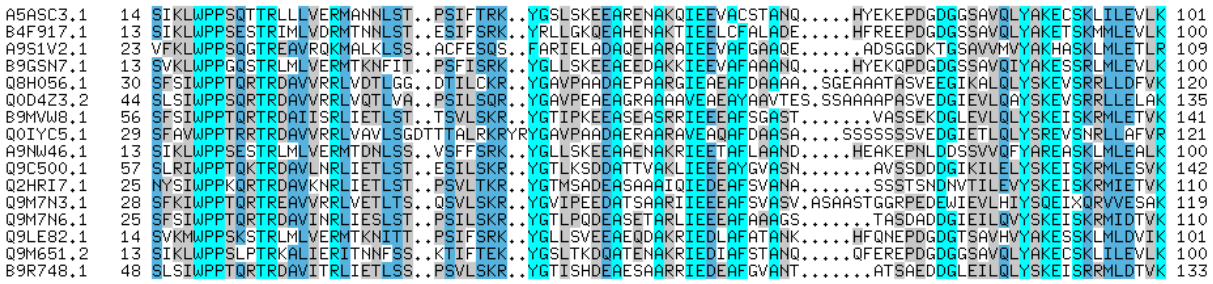
De eerste toepassingen van GPGPU waren matrixvermenigvuldigingen in 2001 en LU decompositie in 2005. Beide zijn bewerkingen op matrices. Matrices zijn van nature uit zeer ontdubbelbaar, omdat je altijd een hele hoop relatief simpele bewerkingen moet doen (optelling en vermenigvuldiging). Later zijn ook andere wiskundige toepassingen in de wiskunde gekomen. Ook AES- en RSA-encryptie lenen zich zeer goed tot het uitvoeren op grafische kaarten.

5.2 Fysica

Een aantal veelvoorkomende, maar toch zeer ingewikkelde problemen zijn onder andere het Ising-model[3] in de statistische fysica (waarmee ferromagnetisme gemodelleerd wordt), kwantumfysica, het berekenen van de beweging van onsamendrukbare vloeistoffen[4] met de Navier-Stokes-vergelijkingen, stroom van Newtoniaanse vloeistoffen (met de Lattice Boltzmann methode, die in plaats van de Navier-Stokes-vergelijkingen andere uitvoert, die gebruik maakt van een collisiestap en een vloeistap), de Monte Carlo methode[5] (een methode die gebruikt wordt om dingen precies te schatten) etc.

5.3 Biologie

Biologische berekeningen die gebruik maken van GPGPU zijn meestal van bio-informatica-aard. Bio-informatica is de wetenschap van zowel het opmaken van programma's en methodes om biologische data te analyseren, als de wetenschap die deze data effectief analyseert. Qua GPGPU gaat het hier vooral over visualisaties van genomen, chromosomen etc. Bij sequentie-analyse van DNA is het ook nuttig om de berekeningen op te splitsen en te laten uitvoeren door GPU's of GPU-clusters.



Figuur 1: *Het vergelijken van het genoom op het WPP-domein*

5.4 Chemie

Ook in de chemie is het haalbaar om zeer ingewikkelde berekeningen om te zetten in kleinere, meer haalbare berekeningen, die op grafische kaarten kunnen uitgevoerd worden. Dit gaat dan bijvoorbeeld over kwantumchemie. Ook het visualiseren van moleculen is een taak die efficiënt is om op te delen in kleine subtaken die door GPU's kunnen worden uitgevoerd.

5.5 Informatica

Natuurlijk kunnen we binnen de informatica alles onderbrengen wat in de andere secties besproken wordt, maar specifiek voor de toepassingen van GPGPU in informatica wordt gedacht aan het opbouwen van high performance computerclusters (HPC clusters[6]), deze maken vaak gebruik van grafische kaarten omdat deze makkelijker ontdubbelbaar zijn. Bij MATLAB-servers wordt er vaak met verschillende GPU's gewerkt, omdat wiskundige taken als deze makkelijk opdeelbaar zijn.

5.5.1 Cryptografie

Cryptografie is het proces dat gebeurt bij het encrypteren van data. Dit gebeurt meestal met AES of RSA. Aan het einde van het vorige decennium werd het haalbaar om deze berekeningen op grafische kaarten uit te voeren. Bij cryptografie worden zeer veel ingewikkelde berekeningen tegelijk uitgevoerd. Dat maakt hiervan een goed voorbeeld om uit te voeren op grafische kaarten.

Niet zo vaak geweten is dat de berekeningen die door cryptografische programma's uitgevoerd worden zo geproduceerd zijn dat ze te moeilijk zijn om met de middelen die op dat moment beschikbaar de encryptie te kraken.[7] Dit kraken houdt in dat je dezelfde berekeningen uitvoert op de uitgekomen getallen. Wat het nu echter zo moeilijk maakt om die berekeningen achterstevoor uit te voeren is dat wanneer je een priemfactorisatie moet maken van zeer grote getallen (dit is nodig om van een uitkomst aan de ingedutte waarden te raken), dat dit dagen, tot zelfs weken kan duren (afhankelijk van de graad van encryptie gebruikt en de hardware natuurlijk).

5.5.2 Cryptocurrencies

Cryptocurrencies zoals Bitcoin, Litecoin en Dogecoin maken gebruik van die cryptografie om een monetair systeem op te stellen. Virtuele munten zoals deze werken door middel van een zogenaamde blockchain. Alle geconfirmeerde transacties komen hierin terecht. Het stuk waarin cryptografie gebruikt wordt is het controleren van die transacties. In de beginperiode van die virtuele munten werden die op gewone GPU's uitgevoerd. GPU's zijn zeer handig voor het controleren van transacties (het zogenaamde *minen*) omdat er daarvoor veel verschillende berekeningen tegelijk moeten gebeuren. In tegenstelling tot CPU's zijn GPU's zeer goed in parallelle berekeningen[9]. Dit is het grote voordeel van GPGPU. Tegenwoordig worden die beperkingen echter op gespecialiseerde apparatuur gedaan zoals ASICMiner[10], AntMiner[11] en andere. Deze zijn zogenaamde ASIC's (circuits die voor een specifieke doeleinde gemaakt zijn)[8]. [12]

5.6 Medische sector

Bij bijvoorbeeld CT-scans en andere methodes van beeldverwerving is er nood aan snelle erekening van anomaliën en een duidelijk beeld zodat er snel een diagnose kan gesteld worden. Hierbij is GPGPU handig, omdat die taken makkelijk onderverdeelbaar zijn in kleinere subtaken, die door een GPU kunnen worden uitgevoerd. Analyse van de uitspreiding van kanker is ook een proces waarop GPGPU toepasbaar is.

6 conclusie

Het kan dus zeer handig zijn om berekeningen uit te voeren op een grafische kaart, maar dit geldt enkel als de berekeningen of methodes opdeelbaar zijn, zodat je gebruik kan maken van de vele threads die een GPU heeft.

Referenties

[1] **GPGPU**, About GPGPU.org, [geraadpleegd op 22 oktober 2014]. http://gpgpu.org/about

[2] **Wikipedia**, General-purpose computing on graphics processing units, [geraadpleegd op 22 oktober 2014]. http://en.wikipedia.org/wiki/General-purpose_computing_on_graphics_processing_units.

[3] **Martin Weigel**, Mini-Workshop Simulations on GPU, Theoretische Physik, Universität des Saarlandes, Saarbrücken, Germany and Institut für Physik, Johannes-Gutenberg-Universität Mainz, Germany

[4] **Douglas C. Giancoli**, Physics for Scientists & Engineers with Modern Physics (6th Edition), Pearson.

[5] **Benjamin Block, Peter Virnau, Tobias Preis**, Multi-GPU Accelerated Multi-Spin Monte Carlo Simulations of the 2D Ising Model, Department of Physics, Mathematics and Computer Science, Johannes Gutenberg University Mainz

[6] **Volodymyr V. Kindratenko, Jeremy J. Enos, Guochun Shi, Michael T. Showerman, Galen W. Arnold, John E. Stone, James C. Phillips, Wen-mei Hwu**, GPU Clusters for High-Performance Computing, University of Illinois at Urbana-Champaign.

[7] **Junior College 2012-2013**, van priemgetal tot digitale handtekening, KU Leuven faculteit Wetenschap & Technologie.

[8] **Wikipedia**, Application-specific integrated circuit. [geraadpleegd op 27 oktober 2014]. http://en.wikipedia.org/wiki/Application-specific_integrated_circuit

[9] **Bitcoin Wiki**, Why a GPU mines faster than a CPU. [geraadpleegd op 23 oktober 2014]. https://en.bitcoin.it/wiki/Why_a_GPU_mines_faster_than_a_CPU

[10] **Bitcointalk**, ASICMiner Publicly Looking for Potential Customers/Partners for New Chips, [geraadpleegd op 27 oktober 2014]. https://bitcointalk.org/index.php?topic=438359.0

[11] **Bitmain**, Products. [geraadpleegd op 27 oktober 2014]. https://www.bitmaintech.com/product.htm

[12] **Bitcoin**, How does Bitcoin work? [geraadpleegd op 27 oktober 2014]. https://bitcoin.org/en/how-it-works