

Hoofdstuk 1

Samenvatting

Ontleding en Voorspelling van de Prestaties van Graafalgoritmes op Beeldverwerkingseenheden

Versnellingsapparatuur in het algemeen, en [Beeldverwerkingseenheden \(BVE's\)](#) in het bijzonder, zijn een standaard onderdeel van hoogprestatieberekeningssystemen geworden. Dit betekent dat het ontleden, begrijpen, en voorspellen van de prestaties van generiekgebruiksbereningen op [BVE's](#) van het hoogste belang is om dergelijke hoogprestatieberekeningssystemen naar behoren te kunnen gebruiken

Voor regelmatige algoritmes — dat wil zeggen, algoritmes met onveranderlijke, van tevoren bekende geheugenhandelingspatronen — is er een veelvoud aan onderzoek dat toont hoe deze algoritmes op een [BVE](#) uitgevoerd kunnen worden en hoe hun prestaties geanalyseerd en voorspeld kunnen worden.

Dit is niet het geval voor onregelmatige algoritmes — dat wil zeggen, algoritmes wiens geheugenhandelingspatronen niet van tevoren bekend zijn omdat zij, bijvoorbeeld, afhangen van de invoergegevens van een berekening. Moderne [BVE's](#) maken gebruik van een diepe

geheugenhiërarchie en regelmatige verwerkingseenheidsstructuur om grootschalig parallelisme te bewerkstelligen. De veranderlijkheid van onregelmatige algoritmes maakt het uitdagend om deze doeltreffend uit te voeren op **BVE's**. Het verbeteren en voorspellen van de prestaties van onregelmatige algoritmes blijft dan ook ingewikkeld.

In dit proefschrift leggen wij ons toe op een deelverzameling van de onregelmatige algoritmes: graafverwerkingsalgoritmes. Deze zijn veelvoorkomend in verschillende wetenschappen, door de toepasbaarheid van grafen als model voor sterk onderlingverbonden gegevens. Echter, vormen grafen ook *het* klassieke voorbeeld voor onregelmatige algoritmes, doordat de verwerking van grafen volkomen afhankelijk is van de samenstelling van de invoergraaf.

Er zijn verschillende moderne voorbeelden van **BVE** graafverwerkingsprogrammatuur [1–4, 6–9] die ontwikkeld zijn om de uitdaging van hoogprestatiegraafverwerking te overkomen. Elk van deze programmaturen gebruikt andere grondbeginselen, technieken, en verbeteringen om om te gaan met de onregelmatige structuur van invoergrafen. De schrijvers doen hun uiterste best om aan te tonen dat hun grondbeginselen doeltreffend op een **BVE** verwezenlijkt kunnen worden en dat dit leidt goede prestaties.

Er is echter weinig informatie die ons helpt begrijpen waar de prestaties geleverd door deze technieken en grondbeginselen vandaan komt. Noch is het duidelijk hoe deze prestaties in verband staan tot de gebruikte invoergegevens en onderliggende apparatuur.

Om de prestaties van graafverwerking op **BVE's** te beginnen te begrijpen hebben wij een holistische aanpak nodig die ons de ingewikkelde wisselwerking tussen invoergegevens, datastructuren, en onderliggende apparatuur laat ontleden. Dit vereist een stelselmatig overzicht van alle informatie langs deze assen. De hulpmiddelen, technieken, theorieën, en werkwijzen die nodig zijn om zo'n overzicht te verkrijgen ontbreken op dit moment.

Het onderzoeksgebied van graafverwerking op **BVE's** is te breed,

te verweve met elkaar, en te verscheiden om in een proefschrift af te handelen. Als gevolg is het hoofddoel van dit proefschrift drievoudig:

1. Hoe de samenstelling van invoergrafen de prestaties van graafverwerkingsalgoritmes beïnvloed ten opzichte van datastructuur keuzes en de onderliggende apparatuur.
2. De ontwikkeling van hulpmiddelen en werkwijzen die een stelselmatig overzicht kunnen geven van de wisselwerking tussen invoergegevens, algoritme, datastructuur, en apparatuur.
3. Onderzoeken of het haalbaar is om deze resultaten te gebruiken om de prestaties van graafverwerking op [BVE's](#) te verbeteren.

In [Hoofdstuk 3](#) op pagina [21](#) tonen wij een ontwerpoverzicht van de door ons ontwikkelde programmatuur voor het opzetten, verzamelen, samenvoegen, en ontleden van prestatiegegevens voor een alomvattende evaluatie van [BVE](#)-graafverwerking. Het gebruik van een enkele gegevensbank voor het beheer en opslag van alle gegevens en metadata verbeterd de gebruikersvriendelijkheid, helpt de herkomst van gegevens te volgen, en versimpeld de reproductie van onze resultaten.

Het gebruik van een enkel bestand, in het breedondersteunde SQLite formaat, maakt het delen van resultaten met andere onderzoekers makkelijker. Dit stelt andere onderzoekers in staat om eenvoudig voort te borduren op onze resultaten, zonder deze tijdrovende prestatiebepalingen te herhalen.

In [Hoofdstuk 4](#) op pagina [53](#) onderzoeken wij hoe de prestaties van verschillende parallelisatiewijzes voor burenbzoek beïnvloed worden door de samenstelling van invoergrafen. Burenbzoek is een basis-handeling die in vele graafverwerkingsalgoritmes gebruikt word. Onze resultaten zijn dus ook bruikbaar buiten de PageRank en [Breedte-eerst Zoekopdracht \(BEZ\)](#) algoritmes die wij voor dit onderzoek gebruiken.

Wij laten zien dat er betekenisvolle verschillen zijn — voor zowel [BEZ](#) en PageRank — in de prestaties van elke parallelisatiewijze. Afhankelijk van de invoergraaf kunnen deze verschillen oplopen tot meerdere ordes van grootte. Tevens tonen wij aan dat de prestaties van [BEZ](#) niet alleen verschillen tussen invoergrafen, maar ook tussen stappen van het [BEZ](#)-verloop. We laten zien dat het juist voorspellen van de beste parallelisatiewijze voor een [BEZ](#)-stap betekenisvolle verbeteringen en prestaties oplevert.

Om de samenhang tussen graafeigenschappen en parallelisatieprestaties te verduidelijken, hebben wij de gestuurde vervaardiging van grafen onderzocht — dat wil zeggen, de vervaardiging van grafen met specifieke eigenschappen. In [Hoofdstuk 5](#) op pagina [89](#) tonen wij onze graafvervaardiger, met focus op het ontwerp van onze *evolutionair algoritme* aanpak. Hoewel onze graafvervaardiger succesvol kleinschalige grafen kan maken, was hij niet in staat dit te bewerkstelligen op de grotere schaal die voor ons nodig is om conclusies te trekken over het verband tussen graafsamenstellingen en parallelisatiewijzes.

In [Hoofdstuk 6](#) op pagina [115](#) vertonen wij werklustmodellen voor elk van onze PageRank parallelisatiewijzes. Deze analytische werklustmodellen zijn gebaseerd op geheugenhandelingen, omdat PageRank grotendeels geheugengebonden is. We toetsen onze werklustmodellen door ze te vergelijken met het gedrag waargenomen door NVIDIA's waarnemingsprogrammatuur. We zien een sterke overeenkomst tussen onze modellen en het waargenomen gedrag.

We laten zien dat, ondanks de overeenkomst tussen modellen en waarnemingen, onze werklustmodellen niet voldoende zijn om best presterende parallelisatiewijze voor een gegeven invoergraaf te voorspellen. Verdere experimenten — met herordende versies van de grafen — tonen aan dat het niet mogelijk is om de parallelle uitvoer van onze werklustmodellen te benaderen. Wij leiden hier uit af dat het nauwkeurig voorspellen van de prestaties van verschillende parallelisatiewijzes niet mogelijk is zonder een gedetailleerd model van het

uitvoergedrag van de [BVE](#) en daarom niet redelijkerwijs haalbaar is.

In [Hoofdstuk 7](#) op pagina [135](#) gebruiken we 247 grafen van de KONECT [\[5\]](#) dataset en verzamelen prestatiemetingen van meerdere [BVE](#)-systemen. We gebruiken deze metingen om een [Binaire Beslis-singsboom \(BBB\)](#)-model te trainen om de beste parallellisatiewijze voor een gegeven graaf te voorspellen. We laten zien dat de voorspellingen van ons [BBB](#)-model beter presteren dan de onze onveranderlijke parallellisatiewijzes.

In [Hoofdstuk 8](#) op pagina [151](#) laten wij zien dat onze [BBB](#)-modellen niet simpelweg de resultaten van onze metingen onthouden. Wij tonen dit aan de door verschillende groottes van trainingsinvoer te gebruiken voor het trainen van onze modellen. We laten zien dat modellen die slechts op een klein onderdeel van onze metingen getraind zijn nog steeds doeltreffend zijn.

We laten ook zien dat onze [BBB](#)-modellen niet beperkt zijn tot de dataset waarop zij getraind zijn. Dit ondersteunt het idee dat onze modellen een deel van het verband tussen graafsamenstelling en parallellisatiewijze vastleggen, en niet slechts de resultaten van een specifieke datasets onthouden.

In [Hoofdstuk 9](#) op pagina [169](#) komen wij tot de slotsom dat dit proefschrift eigenlijk een aantal verschillende beginpunten is. We hebben laten zien dat de samenstelling van een graaf een betekenisvolle invloed heeft op prestaties (zie [Hoofdstuk 4](#)), dat deze invloed bruikbaar is (zie [Hoofdstuk 7](#)), en, tenslotte, dat deze invloed consistent is tussen datasets en [BVE](#) ontwerpen (zie [Hoofdstuk 8](#)). Echter, geen van deze resultaten is, op zichzelf, direct bruikbaar in “de echte wereld”. De belangrijkste bijdrage van dit proefschrift zijn de programmatuur en, in mindere mate, dataset van resultaten die door ons ontwikkeld zijn.

De programmatuur en dataset hebben beide eindeloze mogelijkheden voor hergebruik en verder onderzoek. Een natuurlijke uitbreiding van het werk in dit proefschrift is om verder algoritmes, zoals

1. SAMENVATTING

het kortstepad-algoritme of het tussencentraliteit-algoritme. Een andere richting zou zijn om nog meer datasets van grafen te verkennen, en gedragsverschillen van onze algoritmeïmplementaties te vergelijken tussen de verschillende datasets.

Het zou ook interessant zijn om een meer alomvattend onderzoek te doen naar het gevolg van verschillende mogelijke geheugenordeningen van grafen. Voortbouwend op het verkennend onderzoek in [Hoofdstuk 6](#). Dit onderzoek kan zich richten op hoe de verschillende herordeningen de kwaliteit van de voorspellingen van onze [BBB](#)-modellen aantast. Een andere mogelijk toepassing van onze [BBB](#)-modellen is om deze te hergebruiken als eenvoudige graafindelingsmethodes.

Publications

Merijn Verstraaten. *Belewitte*. Version 1.0.0. Aug. 2022. DOI: [10.5281/zenodo.6959684](https://doi.org/10.5281/zenodo.6959684). URL: <https://doi.org/10.5281/zenodo.6959684>.

Merijn Verstraaten. *Belewitte GPU Experiment Results*. Version v1.0.0. Aug. 2022. DOI: [10.5281/zenodo.6925023](https://doi.org/10.5281/zenodo.6925023). URL: <https://doi.org/10.5281/zenodo.6925023>.

Merijn Verstraaten. *Mix-and-Match Dataset*. Zenodo, Oct. 2018. DOI: [10.5281/zenodo.4317449](https://doi.org/10.5281/zenodo.4317449). URL: <https://doi.org/10.5281/zenodo.4317449>.

Merijn Verstraaten, Ana Lucia Varbanescu, and Cees de Laat. “Mix-and-Match: A Model-driven Runtime Optimisation Strategy for BFS on GPUs”. In: *2018 IEEE/ACM 8th Workshop on Irregular Applications: Architectures and Algorithms (IA3)*. IEEE. 2018, pp. 53–60.

Merijn Verstraaten, Ana Lucia Varbanescu, and Cees de Laat. “Quantifying the Performance Impact of Graph Structure on Neighbour Iteration Strategies for PageRank”. In: *”Euro-Par 2015: Parallel Processing Workshops”*. Springer, Cham. ”Springer International Publishing”, 2015, pp. 528–540. ISBN: ”978-3-319-27308-2”.

PUBLICATIONS

Merijn Verstraaten, Ana Lucia Varbanescu, and Cees de Laat. “Synthetic Graph Generation for Systematic Exploration of Graph Structural Properties”. In: *”Euro-Par 2016: Parallel Processing Workshops”*. Springer, Cham. ”Springer International Publishing”, 2016, pp. 557–570. ISBN: ”978-3-319-58943-5”.

Merijn Verstraaten, Ana Lucia Varbanescu, and Cees de Laat. *Using Graph Properties to Speed-up GPU-based Graph Traversal: A Model-driven Approach*. 2017. eprint: [arXiv:1708.01159](https://arxiv.org/abs/1708.01159).

Bibliography

- [1] Martin Burtscher, Rupesh Nasre, and Keshav Pingali. “A quantitative study of irregular programs on GPUs”. In: *Workload Characterization (IISWC), 2012 IEEE International Symposium on*. IEEE. 2012, pp. 141–151.
- [2] Abdullah Gharaibeh, Lauro Beltrão Costa, Elizeu Santos-Neto, and Matei Ripeanu. “On Graphs, GPUs, and Blind Dating: A Workload to Processor Matchmaking Quest”. In: *IPDPS*. 2013, pp. 851–862.
- [3] Sungpack Hong, Sang Kyun Kim, Tayo Oguntebi, and Kunle Olukotun. “Accelerating CUDA Graph Algorithms at Maximum Warp”. In: *ACM SIGPLAN Notices*. Vol. 46. 8. ACM. 2011, pp. 267–276.
- [4] Farzad Khorasani, Keval Vora, Rajiv Gupta, and Laxmi N Bhuyan. “CuSha: vertex-centric graph processing on GPUs”. In: *HPCS*. ACM. 2014, pp. 239–252.
- [5] Jérôme Kunegis. “KONECT: The Koblenz Network Collection”. In: *Proceedings of the 22Nd International Conference on World Wide Web. WWW '13 Companion*. Rio de Janeiro, Brazil, 2013, pp. 1343–1350. ISBN: 978-1-4503-2038-2.

BIBLIOGRAPHY

- [6] Duane Merrill, Michael Garland, and Andrew S. Grimshaw. “Scalable GPU graph traversal”. In: *PPOPP 2012, New Orleans, LA, USA*. Feb. 2012, pp. 117–128.
- [7] Rupesh Nasre, Martin Burtscher, and Keshav Pingali. “Data-driven versus Topology-driven Irregular Computations on GPUs”. In: *Parallel & Distributed Processing (IPDPS), 2013 IEEE 27th International Symposium on*. IEEE. 2013, pp. 463–474.
- [8] Yangzihao Wang, Andrew Davidson, Yuechao Pan, Yuduo Wu, Andy Riffel, and John D Owens. “Gunrock: A high-performance graph processing library on the GPU”. In: *Proceedings of the 21st ACM SIGPLAN Symposium on Principles and Practice of Parallel Programming*. ACM. 2016, p. 11.
- [9] J. Zhong and B. He. “Medusa: Simplified Graph Processing on GPUs”. In: *17th ACM SIGPLAN symposium on Principles and Practice of Parallel Programming, PPOPP’12 (2012)*, pp. 283–284.

Glossary

BBB Binaire Beslissingsboom 5, 6

BEZ Breedte-eerst Zoekopdracht 3, 4

BVE Beeldverwerkingseenheid 1-3, 5