ETH zürich

ETH Library

Enabling Large Scale DFT Simulation with GPU Acceleration and Machine Learning

Doctoral Thesis

Author(s):
Schütt, Ole

Publication date:
2016

Permanent link:
https://doi.org/https://doi.org/10.3929/ethz-a-010819495

Rights / license:
In Copyright - Non-Commercial Use Permitted

This page was generated automatically upon download from the ETH Zurich Research Collection.
For more information, please consult the Terms of use.

DISS. ETH NO. 23879

Enabling Large Scale DFT Simulation with GPU Acceleration and Machine Learning

A thesis submitted to attain the degree of
DOCTOR OF SCIENCES of ETH ZURICH
(Dr. sc. ETH Zurich)

presented by

Ole Schütt

Dipl.-Phys. Dipl.-Inform., FU Berlin

born on: 09.04.1985

citizen of Germany

accepted on the recommendation of
Prof. Dr. Joost VandeVondele
Prof. Dr. Thomas Schulthess
Prof. Dr. Jürg Hutter

2016

Abstract

The rapid increase in computational power and the development of linear scaling methods now allow for single-energy DFT calculations of systems with 10.000 - 1.000.000 atoms. However, the approach is computationally demanding for routine application, especially if first principles molecular dynamics or relaxation is required. In this work, two complementary approaches were developed to boost the performance of large scale DFT calculations. First, the predominant operation, sparse matrix-matrix multiplication, was ported to GPU accelerators. Afterwards, the performance was further increased employing geometry adapted basis sets obtained with machine learning.

For an increasing number of supercomputers, the majority of computing power is provided by accelerator devices, including CSCS's current flagship *Piz Daint*. To take full advantage of the GPU's high throughput parallelism within the established and highly MPI+OpenMP-parallel CP2K code base required significant algorithmic development. In particular, to accommodate for the GPU's asynchronous task based programming model, the matrix multiplication code in CP2K's sparse matrix library DBCSR was redesigned from the ground up. It was transformed from a lock-step scheme into a barrier free algorithm that relies solely on Cuda streams and events, as wells as asynchronous MPI calls. This software architecture allows for fully overlapping the double-buffered Cannon's algorithm with host-to-device copies, CPU book-keeping work, and Cuda kernel execution. Additionally, Cuda kernels for representative matrix sizes were developed together with Nvidia engineers as part of a co-design effort. These kernels served as templates for the auto-tuning of over 2350 kernels for domain specific matrix-sizes. For example, the single-node performance for the important 23 matrix size, comparing GPU+CPU vs CPU-only, was improved by 450%. As a consequence, sparse matrix-matrix multiplication has become an important primitive also for other methods in CP2K.

The second part of this work exploits the observation that the computational cost of a DFT calculation critically depends on the size and condition number of the employed basis set. Here, a small but optimal basis set is obtained by adapting an atom centered basis set to its chemical environment. To avoid a costly and cumbersome optimization for each molecular geometry, a machine learning approach was employed to infer the optimal basis from a limited set of training points. To make this possible, the basis is derived from the molecular geometry in a way that is invariant to rotations of the system and permutations of the atoms. These basis functions are obtained as the lowest energy states of an isolated atom perturbed by a molecular geometry dependent, and machine learned potential. The machine learning maps a fingerprint of the local geometry to optimal parameters for the potential parametrization. A key solved problem was obtaining consistent potentials, thus suitable for learning, which required regularization and effective optimization. The method has been demonstrated using MD simulations of liquid water, where it is shown that minimal basis sets yield structural and dynamical properties that outperform those of semi-empirical methods at similar cost. Compared to the standard approach, the number of flops needed in matrix-matrix multiplications decreased by over four orders of magnitude, leading to an effective 60-fold run-time speedup.

Zusammenfassung

Durch den rapiden Anstieg der verfügbaren Rechenleistung sowie der Entwicklung von linear skalierenden Methoden können Einzelenergien von 10.000 - 1.000.000 Atomen mit DFT berechnet werden. Allerdings ist auch dieser Ansatz für die alltägliche Anwendung sehr rechenintensiv, insbesondere dann, wenn ab-initio Molekulardynamik und Äquilibrierungen benötigt werden. In dieser Arbeit wurden zwei sich ergänzende Ansätze entwickelt, um die Leistungsfähigkeit von grossskaligen DFT-Berechnungen zu verbessern. Zuerst wurde die dominierende Operation, nämlich die Multiplikation von dünnbesetzten Matrizen, auf GPU-Beschleuniger portiert. Anschliessend wurde die benötigte Rechenzeit durch einen an die Geometrie angepassten und auf maschinellem Lernen basierenden Basissatz weiter verkürzt.

Für eine zunehmende Zahl an Supercomputern wird die Rechenleistung vorwiegend durch Beschleunigerkarten bereitgestellt, einschliesslich CSCS's aktuellem Flaggschiff *Piz Daint*. Es waren erhebliche algorithmische Entwicklungen nötig, um den Hochdurchsatz-Parallelismus der GPUs zusammen mit der etablierten und MPI + OpenMP-parallelisierten CP2K Codebasis vollständig nutzen zu können. Insbesondere für das GPU-typische asynchrone und aufgabenorientierte Programmiermodell, musste der Matrixmultiplikationscode in CP2K's Bibliothek für dünnbesetzte Matrizen (DBCSR), von Grund auf neu entworfen werden. Er wurde von einem Gleichschritt-Schema in einen nicht-blockierenden Algorithmus transformiert, der einzig auf Cuda-Streams und Events sowie auf asynchronen MPI-Aufrufen basiert. Diese Softwarearchitektur erlaubt den doppelt gepufferten Cannon's Algorithmus mit CPU-zu-GPU-Kopien, CPU-Buchhaltungsarbeiten sowie Cuda Kernelausführungen komplett zu überlappen. Zusätzlich wurden, in Zusammenarbeit mit Nvidia-Ingenieuren und als Teil eines co-design Bestrebens, Cuda-Kernel für repräsentative Matrizengrössen entwickelt. Auf der Grundlage dieser Kernel wurden später über 2350 Kernel für weitere domänenspezifische Matrizengrössen auto-optimiert. Zum Beispiel wurde

die Einzelknotenleistung der wichtigen Matrizengrösse 23, im Vergleich von einer alleinigen CPU zu GPU+CPU, um 450% verbessert. Als Folge davon ist die Multiplikation von dünnbesetzten Matrizen eine wichtige Primitive auch für andere Methoden in CP2K geworden.

Der zweite Teil dieser Arbeit beruht auf der Beobachtung, dass die Rechenintensität einer DFT-Berechnung kritisch von der Grösse und Konditionierung des benutzten Basissatzes abhängt. Durch die Adaption eines atomzentrischen Basissatzes an seine chemische Umgebung wurde ein kleiner, aber optimaler Basissatz erzeugt. Um die teure und umständliche Optimierung für jede Molekülgeometrie zu vermeiden, wurde ein Ansatz des maschinellen Lernens gewählt, der den optimalen Basissatz aus einem begrenzten Trainingssatz ableitet. Um dies zu ermöglichen, wurde die Basis von der Molekülgeometrie in einer Weise abgeleitet, die invariant unter Rotation und Vertauschung von Atomen ist. Die Basisfunktionen ergeben sich als die energetisch niedrigsten Zustände eines isolierten Atoms, das durch ein molekülgeometrieabhängiges und maschinengelerntes Potential gestört wird. Das maschinelle Lernen bildet einen Fingerabdruck der lokalen Geometrie auf die optimalen Parameter der Potentialparametrisierung ab. Ein Schlüsselproblem war die Erzeugung von konsistenten, und daher für maschinelles Lernen geeigneten Potentialen. Diese benötigten zudem Regularisierung und eine effiziente Optimierung. Anhand von MD-Simulationen von flüssigem Wasser wurde demonstriert, dass diese Methode zu ähnlichen Kosten bessere Resultate für strukturelle und dynamische Eigenschaften liefert als semi-empirische Methoden. Im Vergleich zum Standardansatz konnten die benötigten Gleitkommaoperationen für Matrix-Matrix-Multiplikationen um vier Grössenordnungen gesenkt werden, welches effektiv zu einer 60-fachen Laufzeitverkürzung führte.