Statusbericht des SeASiTe -Projekts

GEFÖRDERT VOM

Bundesministerium
für Bildung
und Forschung

Matthias Korch

Universität Bayreuth
Institut für Informatik

UNIVERSITÄT
BAYREUTH

8. HPC-Status-Konferenz der Gauß-Allianz
Regionales Rechenzentrum Erlangen (RRZE)

8. und 9. Oktober 2018

Inhalt

GEFÖRDERT VOM
Bundesministerium für Bildung und Forschung

Förderung, Partner, Ziel

GEFÖRDERT VOM
Bundesministerium für Bildung und Forschung

Selbstadaption für zeitschrittbasierte Simulationstechniken auf heterogenen HPC-Systemen

- Gefördert vom BMBF seit 1. 1. 2017
- Programm „IKT 2020 – Forschung für Innovationen“
- Bekanntmachung „Grundlagenorientierte Forschung für HPC-Software im Hoch- und Höchstleistungsrechnen“

Förderung, Partner, Ziel

GEFÖRDERT VOM
Bundesministerium für Bildung und Forschung

Selbstadaption für zeitschrittbasierte Simulationstechniken auf heterogenen HPC-Systemen

- Gefördert vom BMBF seit 1. 1. 2017
- Programm „IKT 2020 – Forschung für Innovationen“
- Bekanntmachung „Grundlagenorientierte Forschung für HPC-Software im Hoch- und Höchstleistungsrechnen“
- 3 Universitäten + 1 assoziiertes Unternehmen, 6 Doktoranden, 3 Jahre

TECHNISCHE UNIVERSITÄT CHEMNITZ
Praktische Informatik, Gudula Rünger

FAU FRIEDRICH-ALEXANDER UNIVERSITÄT ERLANGEN-NÜRNBERG
Höchstleistungsrechnen, Gerhard Wellein

UNIVERSITÄT BAYREUTH
Parallele und verteilte Systeme,
Thomas Rauber und Matthias Korch
(Koordination)

MEGWARE
(assoziierter Partner)
Jürgen Gretzschel

SeASiTe

Förderung, Partner, Ziel

GEFÖRDERT VOM
Bundesministerium für Bildung und Forschung

Selbstadaption für zeitschrittbasierte Simulationstechniken auf heterogenen HPC-Systemen

- Gefördert vom BMBF seit 1. 1. 2017
- Programm „IKT 2020 – Forschung für Innovationen“
- Bekanntmachung „Grundlagenorientierte Forschung für HPC-Software im Hoch- und Höchstleistungsrechnen“
- 3 Universitäten + 1 assoziiertes Unternehmen, 6 Doktoranden, 3 Jahre

☞ Simulationssoftware sollte sich soweit möglich selbst an sich verändernde heterogene HPC-Hardware und neue Eingaben (Anfangswerte, Modell) anpassen.

TECHNISCHE UNIVERSITÄT CHEMNITZ
Praktische Informatik, Gudula Rünger

FAU FRIEDRICH-ALEXANDER UNIVERSITÄT ERLANGEN-NÜRNBERG
Höchstleistungsrechnen, Gerhard Wellein

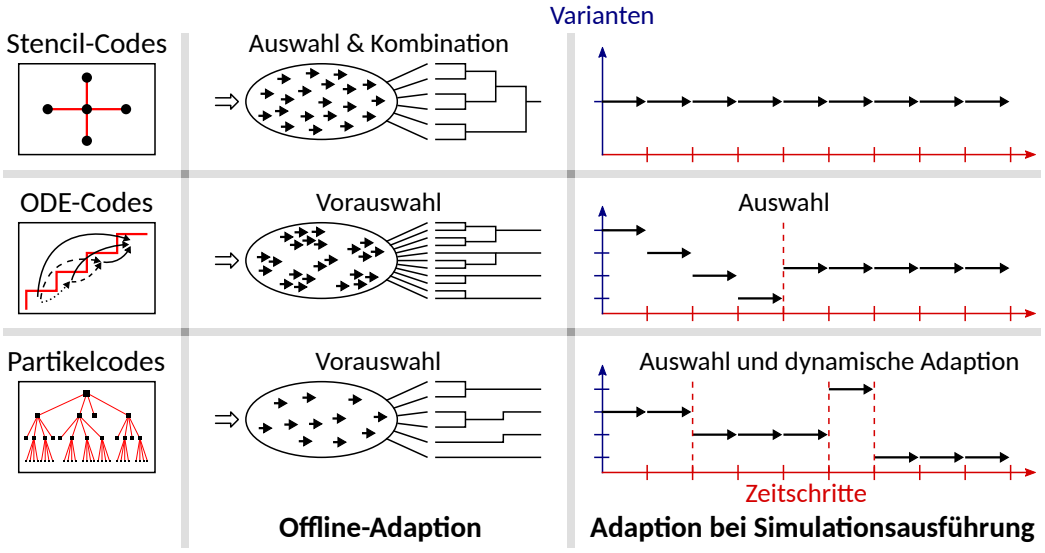
Parallele und verteilte Systeme, Thomas Rauber und Matthias Korch (Koordination)

MEGWARE
(assoziierter Partner)
Jürgen Gretzschel

SeASiTe

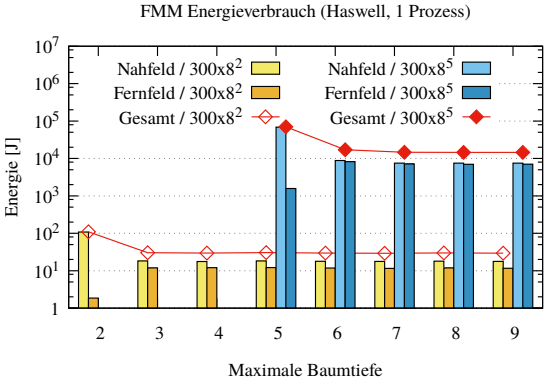
GEFÖRDERT VOM
Bundesministerium für Bildung und Forschung

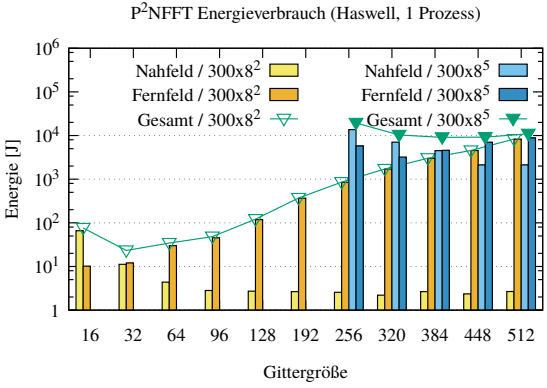
Entwicklung und Integration selbstadaptierender Simulationsverfahren



Partikelcodes – Analyse von Programmparametern

GEFÖRDERT VOM
Bundesministerium für Bildung und Forschung

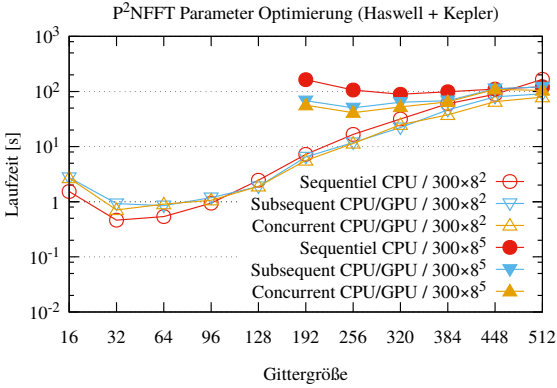


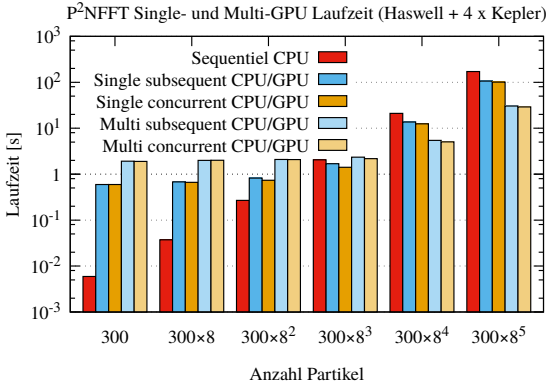


- Verschiedene Partikellöser der ScaFaCoS-Bibliothek wurden untersucht
- FMM ist baumbasiert, P^{2}NFFT gitterbasiert
- Energieverbrauch mit angepassten Programmparametern untersucht
- Nahfeld- und Fernfeldberechnungen gemessen
- Optimale Parameter hängen vom Partikelsystem ab

Partikelcodes – OpenCL-Umsetzung

GEFÖRDERT VOM
Bundesministerium für Bildung und Forschung



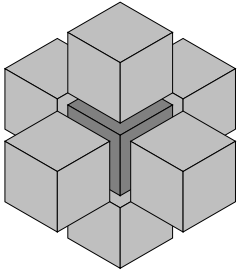


- Nahfeldlöser der ScaFaCoS-Bibliothek wurde in OpenCL umgesetzt
- Unterstützt nun heterogene Systeme
  ☞ Fernfeldlöser auf CPU und gleichzeitig Nahfeldlöser mit GPU
- Nutzung mehrerer GPUs möglich
- OpenCL-Nutzung bei großen Partikelsystemen mit Laufzeitvorteil

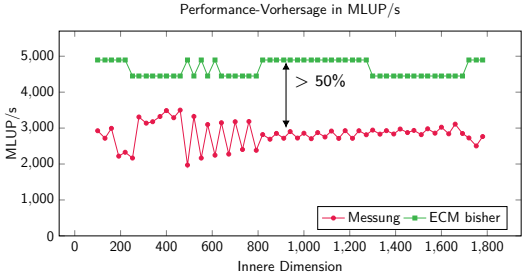
GEFÖRDERT VOM
Bundesministerium für Bildung und Forschung

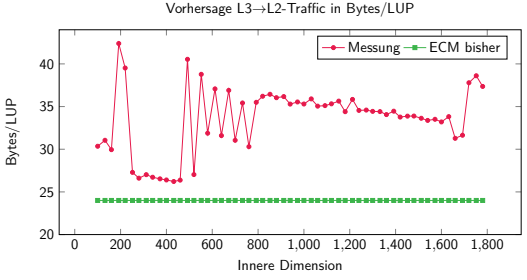
ECM-Modellierung zeitlich geblockter Stencils

3D 7-Punkt-Stencil, Intel Ivy Bridge E5-2660 v2 @ 2.20GHz



3D 7-Punkt-Stencil





- Zeitlich geblockt in L3
- Äußere „Layer Condition“ erfüllt im L2

☞ Bisheriges ECM-Modell zu ungenau

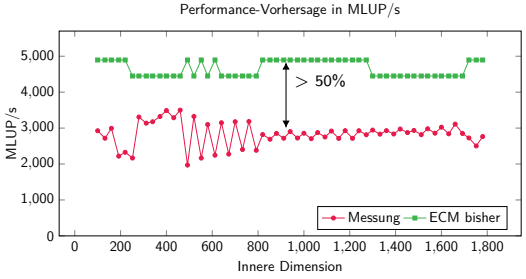
ECM-Modellierung zeitlich geblockter Stencils

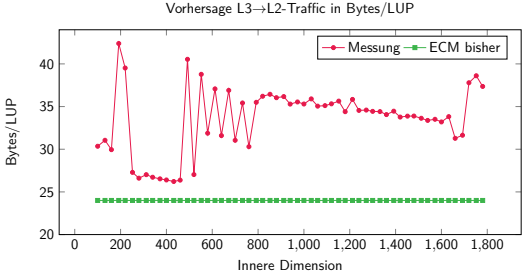
3D 7-Punkt-Stencil, Intel Ivy Bridge E5-2660 v2 @ 2.20GHz

Warum versagt das bisherige ECM-Modell?

Weitere Hardware-Details müssen einbezogen werden, insbesondere da die Blockgröße aufgrund der zeitlichen Blockung sehr klein wird.

- Aufteilung der Schleife in Burst Read und Cache-Wiederverwendung
- Restschleife
- Overhead durch Randbedingungen
- Overhead durch Prefetching
- Datenlatenz
- Realistischeres k-Wege-Caching gegenüber vollassoziativem Caching





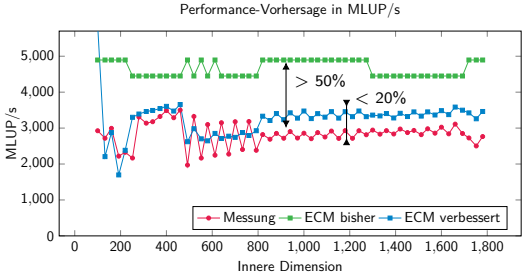
ECM-Modellierung zeitlich geblockter Stencils

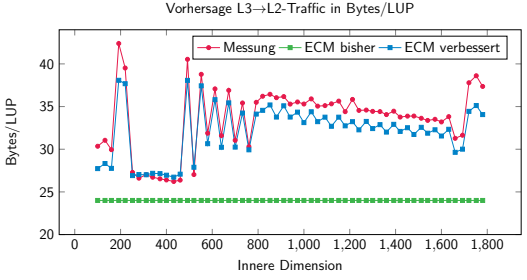
3D 7-Punkt-Stencil, Intel Ivy Bridge E5-2660 v2 @ 2.20GHz

Warum versagt das bisherige ECM-Modell?

Weitere Hardware-Details müssen einbezogen werden, insbesondere da die Blockgröße aufgrund der zeitlichen Blockung sehr klein wird.

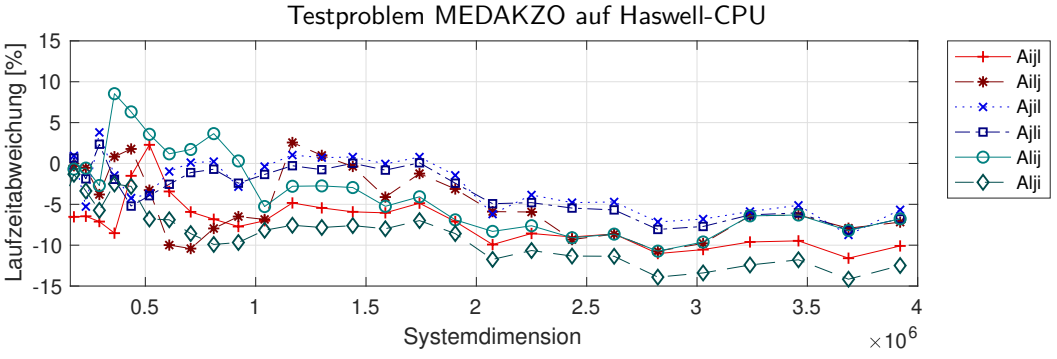
- Aufteilung der Schleife in Burst Read und Cache-Wiederverwendung
- Restschleife
- Overhead durch Randbedingungen
- Overhead durch Prefetching
- Datenlatenz
- Realistischeres k-Wege-Caching gegenüber vollassoziativem Caching





Anwendung des ECM-Modells auf ODE-Verfahren

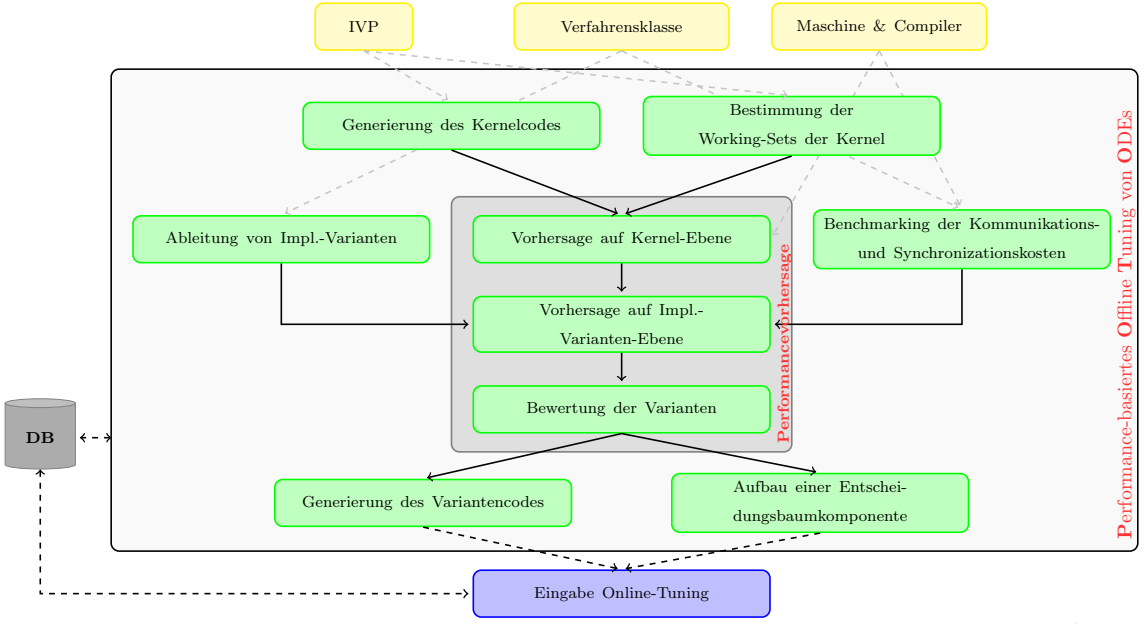
GEFÖRDERT VOM
Bundesministerium für Bildung und Forschung



- Außer für kleine $n \lesssim 10^6$, ist die Vorhersage i. A. zu optimistisch.
- Abweichungen bleiben innerhalb von 15 %.
- Einige Abweichungen durch Schwankungen der Speicherbandbreite bei den Messungen erklärbar. (Vorhersage nutzt feste Bandbreite!)
- Für $n \gtrsim 1{,}5 \cdot 10^6$ wird beste Variante recht zuverlässig gefunden.

Performance-basiertes Offline-Tuning von ODE-Codes

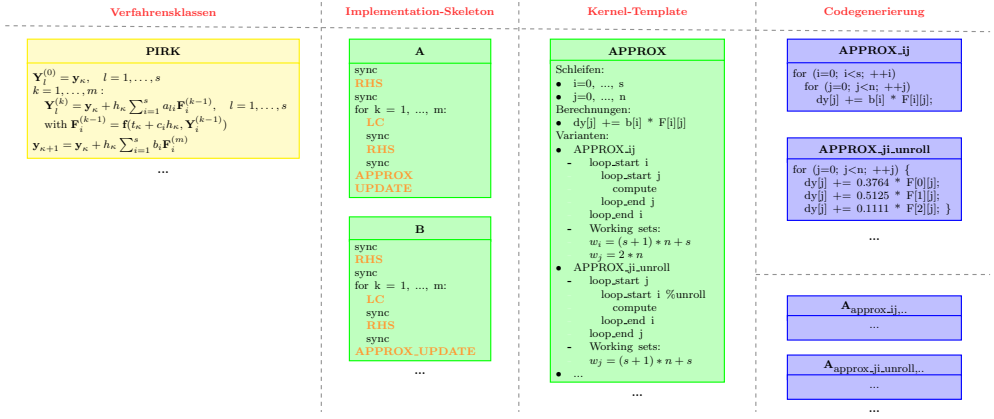
GEFÖRDERT VOM
Bundesministerium für Bildung und Forschung



SeASiTe

DSL für Implementierungsvarianten von ODE-Verfahren

GEFÖRDERT VOM
Bundesministerium für Bildung und Forschung



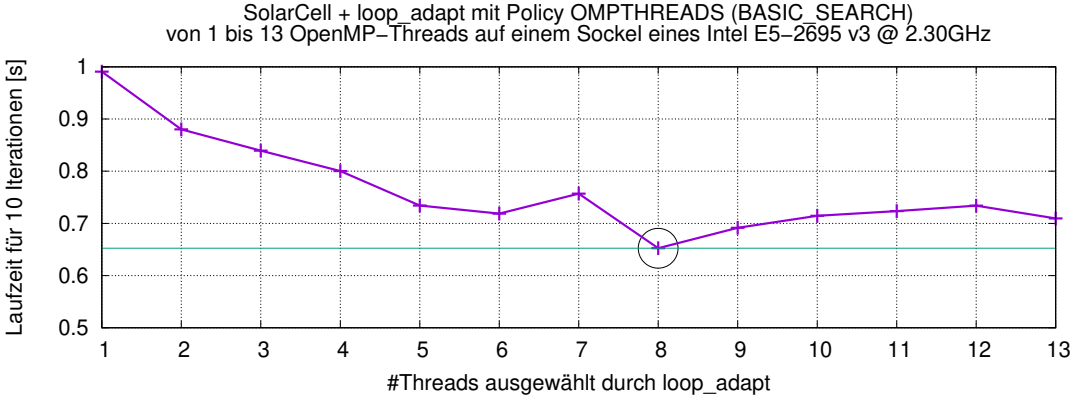
- Beschreibung einer möglichen Implementierungsvariante einer Verfahrensklasse durch *Implementation-Skeleton*.
- Abstrahierung der Basisbausteine des Verfahrens durch *Kernel*.
- Definition aller möglichen Varianten eines Kernels im *Kernel-Template*.
- Codegenerierung für Performancemodellierung und Online-Tuning
  ☞ spezialisierter Code möglich (fixe Verfahrenskoeffizienten und fixe RHS)

SeASiTe

Online-Adaption von Programmparametern – loop_adapt

GEFÖRDERT VOM
Bundesministerium für Bildung und Forschung

- Spezifikation von Schleifen, Bewertungsrichtlinien, Suchalgorithmen, Laufzeitparametern
- Adaption der Parameter durch Suchalgorithmen
- Integration mit LIKWID (☞ nutzerdefinierte Bewertungsrichtlinien)
- Erfassung und Berücksichtigung der heterogenen Hardware-Topologie
- Ausgabe der Laufzeitprofile und Parameterwerte zur Wiederverwendung im nächsten Lauf



GPU-Monitoring mit LIKWID

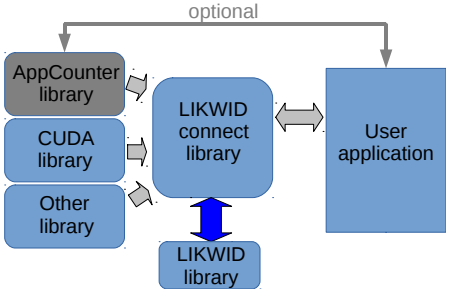
GEFÖRDERT VOM
Bundesministerium für Bildung und Forschung

Ziele:

- Keine Veränderungen am Sourcecode der Applikation notwendig
- Möglichkeit der Steuerung von außen (`likwid-perfctr`) und innen (`MarkerAPI`)
- Generisches Interface zum Anbinden mehrerer Messbibliotheken

Aktuelle Umsetzung:

- Laden der Bibliotheken zum Programmstart (`LD_PRELOAD`)
- Bibliothek mit Kommunikationsthread zur Anbindung an LIKWID
- Thread für Kommunikation und Overflow-Erkennung
- Spezifische Bibliotheken für Messungen (CUDA/CUPTI für NVIDIA GPUs fertig)



Zusammenfassung und Ausblick

GEFÖRDERT VOM
Bundesministerium für Bildung und Forschung

Status:

- Sensitivitätsanalysen durchgeführt
- Heterogene Programmcodes werden erstellt
- ECM-Modell wurde für Stencils verbessert und auf ODE-Verfahren übertragen
- Konzept für Offline-Tuning
- Erste Konzepte für Online-Tuning
- Weiterentwicklung und Integration von LIKWID

Ausblick:

- Weiterentwicklung Performance-Vorhersage (u. a. Energiemodell)
- Weiterentwicklung und Implementierung der Selbstadaptionskonzepte und -werkzeuge
- Umsetzung der Demonstratoren