(Arbeitstitel: Geschwindigkeitsanalyse eines Hadoop-Systems in Bezug auf wissenschaftliche / Klima-Daten)

— Exposé —

Arbeitsbereich Wissenschaftliches Rechnen
Fachbereich Informatik
Fakultät für Mathematik, Informatik und Naturwissenschaften
Universität Hamburg

Vorgelegt von:	Arne Struck
E-Mail-Adresse:	1struck@informatik.uni-hamburg.de
Studiengang:	Bsc. Informatik
Betreuer:	Julian Kunkel

Hamburg, den 12. Mai 2015

Inhaltsverzeichnis

1 Motivation

Der Bereich des High Performance Computing ist seit Jahrzehnten ein wichtiges Forschungs- und Anwendungsgebiet der Informatik. Oftmals befasst sich das High Performance Computing mit Fragen der Meteorologie, Klimatologie, theoretischen Physik und Biologie. Die Lösungen dieser Fragen wird durch die Simulation von Modellen aus dem entsprechenden Bereich und die Analyse von gemessenen und berechneten Daten approximiert.
Dies sind meist sehr rechenintensive Tätigkeiten, daher sind extrem leistungsstarke Rechner von Nöten. Heutzutage werden zu diesem Zweck meistens Cluster, also verteilte Rechensysteme eingesetzt. Bei der Realisierung der oben erwähnten Tätigkeiten auf verteilten Systemen ist die Frage der Parallelisierung von Berechnungen eine entscheidende. Im High Performance Computing Bereich wird dies seit Jahrzehnten großteilig durch Message Passing Interface MPI (beispielsweise implementiert durch Open MPI) erreicht [Lud14].
Durch das Aufkommen des Internets als Massenmedium in den letzten beiden Jahrzehnten hat sich eine Ökonomie entwickelt, welche große Datenmengen managen und verarbeiten muss. Auch diese Ökonomie, deren Geschäftsfeld allgemein als Big Data bezeichnet wird, setzt stark auf verteilte Systeme. Allerdings wird hier nicht auf MPI für das Arbeiten auf verteilten Systemen gesetzt, sondern andere Lösungen für die Problematiken, welche verteilte Berechnungen mit sich bringen, bevorzugt. Als Beispiel für solche Lösungen wären spark und hadoop zu nennen, welche nicht auf dem MPI-Technologie Stack aufbauen, sondern neue Herangehensweisen darstellen.
Diese Herangehensweisen versprechen leichtere Nutzbarkeit und Umsetzbarkeit durch Reduktion des durch die Parallelisierung bedingten Overheads im Programm, sowie höhere Flexibilität des Codes. Analysen der Interessenlagen bezüglich der verschiedenen Herangehensweisen weisen auf einen Trend in Richtung der Lösungen der Big Data Unternehmen auf [Dur]. Die "Big Data Unternehmen" besitzen weiterhin das Potential High Performance Computing in Größenordnungen anzubieten, welche durchaus die der HPC Community übersteigen (erste Tendenzen in diese Richtung sind momentan zu beobachten [Lud14]).
Wegen der Verschiebung der Interessenslagen, der relativen Schwerfälligkeit MPIs und nicht zuletzt der Übermacht der Privatwirtschaft drängen sich für das klassische HPC die Frage auf, beziehungsweise ob die Lösungen aus der Big Data Ökonomie leistungstechnisch mit MPI konkurrieren können und somit eine zumindest partielle Alternative auch für die momentane High Performance Computing Community darstellen.

2 Problemstellung

3 Theoretischer Hintergrund

4 Methodik und Vorgehensweise

5 Zeitplan

Phase	**Gegenstand**	**veranschlagte Zeit**
Recherche	Literaturrecherche Theorieteil der Arbeit	2 Wochen
Design	Theoretische Umsetzung	3 Wochen
Implementation	Implementation, Datenerhebung, etc.	2 Wochen
Evaluation	Analyse, Interpretation	2 Wochen
Abschluss	Schluss schreiben und Überarbeitung	1 Woche
Restzeit	Pufferzone für Phasen, die länger dauern	2 Wochen

Tabelle 5.1: Zeitplan

Hierbei ist zu beachten, dass dies die Bearbeitungsdauer widerspiegelt und nicht den Bearbteitungszeitraum. Die Restzeit ist für die Arbeiten vorgesehen, bei denen sich herauskristalisiert, dass die veranschlagte Zeit nicht genügt. Da der Tag der Anmeldung noch nicht final steht, kann ich leider keine Angaben für den Bearbeitungszeitraum machen.

Literaturverzeichnis

[Dur] Jonathan Dursi. Hpc is dying, and mpi is killing it. `http://www.dursi.ca/hpc-is-dying-and-mpi-is-killing-it/`.

[Kun13] Julian Martin Kunkel. *Simulation of Parallel Programs on Application and System Level*. PhD thesis, Universität Hamburg, Von-Melle-Park 3, 20146 Hamburg, 2013.

[Lud14] Prof. Thomas Ludwig. Hochleistungsrechnen. Vorlesung, 2014.