TU Clausthal

Kolloquium zur Masterarbeit

Kurt Böhm

„Hybrid-Parallelisierung eines algebraischen Mehrgitterlösers für lineare Gleichungssysteme"

Viele Vorgänge in Naturwissenschaft und Technik lassen sich als partielle Differentialgleichungen modellieren, die zumeist durch Diskretisierung in ein lineares Gleichungssystem überführt werden.
Eine effiziente und weit verbreitete Klasse von numerischen Lösern für so gewonnene Gleichungssysteme stellen die Mehrgitterverfahren dar, die durch Nutzung einer Hierarchie von Diskretisierungen besonders effizient sind.
Das PAAMG-Verfahren, auf dem diese Arbeit aufbaut, stellt ein generisches algebraisches Mehrgitterverfahren dar. Der wesentliche Bestandteil dieses Verfahrens ist ein Aggregationsverfahren, dass die Abbildung zwischen den Ebenen der Diskretisierungshierarchie bestimmt, indem die Matrix als gewichteter Graph interpretiert und darin Gruppen von Knoten identifiziert werden, die miteinander kombiniert werden können.
Eine existierende Implementierung des PAAMG-Verfahrens im ISTL-Modul des DUNE-Projekts weist jedoch einige Probleme und Einschränkungen in Hinblick auf Code-Qualität und Leistung auf.
Außerdem ist diese Implementierung rein prozessbasiert unter Nutzung von MPI, was durch regelmäßigen Datenaustausch auf großen Clustern zur Limitierung werden kann. Dieser Kommunikationsaufwand kann auf Mehrkernprozessoren durch eine Shared-Memory-Parallelisierung reduziert werden.
So war es zum einen Ziel der Arbeit, die Effizienz und Lesbarkeit der Implementierung zu verbessern.
Zum anderen wurden einige Varianten der Shared-Memory-Parallelisierung des PAAMG-Verfahrens mit verschiedenen Kompromissen zwischen Laufzeit und Qualität der Aggregation entwickelt, die im Verbund mit der MPI-Parallelisierung als hybride Parallelisierung auf Clustern aus Mehrkernprozessoren genutzt werden können. Besonderes Augenmerk lag auf der Aufteilung der Matrix durch einen Graphpartitionierer, wofür das neue Verfahren „Mustard" entwickelt wurde.
Innerhalb einer Evaluation mit zwei Gruppen von Testfällen hat sich ergeben, dass der sequentielle Aggregationsalgorithmus der verbesserten Implementierung im direkten Vergleich mit dem der ursprünglichen Implementierung eine Beschleunigung von 30% bis 72% aufweist.
Unter den in dieser eingeführten Shared-Memory-Parallelisierungsansätzen
sticht der Ansatz mit Partitionierung durch Mustard heraus, der bei großen Probleminstanzen zu den schnellsten Parallelisierungen gehört, wenn die Partitionierung nicht mitgezählt wird. Doch auch unter Beachtung der Dauer der Partitionierung ist Mustard in den meisten Fällen schneller als der sequentielle Algorithmus.

Montag, 12.04.2021, 13:00 Uhr

Videokonferenz: BBB https://webconf.tu-clausthal.de/b/kur-frb-ba0-nny

Technische Universität Clausthal • Institut für Informatik • Julius-Albert-Straße 4 • 38678 Clausthal-Zellerfeld
Tel +49 (0)53 23 72-7140• Fax +49 (0)5323 72-7199 • http://www.in.tu-clausthal.de • info@in.tu-clausthal.de