Lehrstuhl für Angewandte Informatik II

Prof. Dr. T. Rauber Tel.: 0921/55 7700 Büro: AI 2.08
Internet: http://ai2.inf.uni-bayreuth.de
E-Mail: rauber@uni-bayreuth.de

UNIVERSITÄT BAYREUTH

Implementierung paralleler Algorithmen mit modernen parallelen Programmiersprachen und Programmierumgebungen

mehrere Bachelor-, Master- oder Zulassungsarbeiten

Hintergrund

Aktuell findet eine aktive Forschung im Bereich paralleler Programmiermodelle statt. Die im Rahmen von Forschungsarbeiten vorgeschlagenen Programmiersprachen, Programmierumgebungen und Programmbibliotheken haben jeweils bestimmte Vorzüge, meist aber auch Nachteile gegenüber konkurrierenden Vorschlägen. Etabliert und zu Standards entwickelt haben sich im Laufe der Zeit MPI, POSIX Threads (Pthreads) und OpenMP. Beispiele für neuere Projekte sind UPC, Chapel, X10, Fortress, Cilk/Cilk++, Intel Threading Building Blocks, Charm++ und MCSTL.

Aufgabenstellung

Jeweils eine Programmiersprache, Programmierumgebung oder Programmbibliothek soll evaluiert werden, indem ein oder mehrere parallele Algorithmen (z. B. aus dem Bereich des wissenschaftlichen Rechnens oder der Computergrafik) mit dieser implementiert und untersucht werden. Ausgangspunkt für die Implementierung ist entweder eine textuelle Beschreibung des jeweiligen parallelen Algorithmus oder eine vorhandene Implementierung in einer klassischen Programmiersprache (meist C), die z. B. einer Benchmarksammlung entnommen werden kann. Folgende Arbeitsschritte sind durchzuführen:

- Analyse des Parallelitätspotentials: Welche Parallelisierungsmöglichkeiten gibt es für die betrachtete Anwendung oder Problemstellung? Wieviele Prozessorkerne sind theoretisch nutzbar? Wo könnten Engpässe für die Skalierbarkeit entstehen?
- Erarbeitung einer Übersicht über Eigenschaften und Besonderheiten der betrachteten Sprache, Umgebung oder Bibliothek: Welche Möglichkeiten zur Beschreibung paralleler Algorithmen werden zur Verfügung gestellt? Welche davon sind für die betrachtete Anwendung oder Problemstellung geeignet? Welche sollen für die Implementierung in der Arbeit verwendet werden?
- Implementierung des parallelen Algorithmus mit der betrachteten Sprache, Umgebung oder Bibliothek.
- Bei Bedarf Erstellung einer Vergleichsimplementierung in einem etablierten parallelen Programmiermodell (MPI, POSIX Threads, OpenMP oder JavaThreads).
- Ausführliche Analyse des parallelen Laufzeitverhaltens der Implementierung auf verschiedenen parallelen Rechnersystemen. Soweit möglich, Vergleich mit etabliertem Programmiermodell.
- Auswertung: Wie gut ist die betrachtete Sprache, Umgebung oder Bibliothek für die Implementierung des betrachteten Algorithmus geeignet? Welche Vor- und Nachteile hat sie gegenüber etablierten Programmiermodellen? Welche parallele Performance wurde erreicht?

Erforderliche Kenntnisse und Fähigkeiten

- Prozessor- und Rechnerarchitektur
- parallele Programmierung

Betreuer

Prof. Dr. Thomas Rauber, Dr. Matthias Korch[1], Andreas Prell[2]

[1] Büro: AI 2.11, Tel.: 7705, E-Mail: korch@uni-bayreuth.de
[2] Büro: AI 2.14, Tel.: 7709, E-Mail: andreas.prell@uni-bayreuth.de