[Prof. Dr. Wolfgang Karl // **Rechnerarchitektur und Parallelverarbeitung**]

Wolfgang Karl studierte von 1979 bis 1986 Informatik an der Friedrich-Alexander-Universität Erlangen-Nürnberg. Er promovierte 1992 mit einer Arbeit über parallele Prozessorarchitekturen und ihren Codegenerierungstechniken an der Fakultät für Informatik der Technischen Universität München. Im Jahr 2002 habilitierte er sich dort mit einer Arbeit über die Architektur und effiziente Programmierung von Cluster-Systemen. Seit 2003 ist er Professor für Informatik am Karlsruher Institut für Technologie (KIT). Am Institut für Technische Informatik (ITEC) leitet er die Forschungsgruppe Rechnerarchitektur und Parallelverarbeitung.

Zu den Forschungsschwerpunkten von Wolfgang Karl gehören die Architektur und die effiziente Nutzung heterogener paralleler Rechnerstrukturen.

In der Gesellschaft für Informatik (GI) war er von 2010–2013 und von 2020–2021 Mitglied des erweiterten Vorstands und ist seit 2010 Mitglied des Präsidiums. Er ist Sprecher des GI / ITG Fachbereichs Technische Informatik. In der Informationstechnischen Gesellschaft (ITG) im VDE ist er Mitglied im wissenschaftlichen Beirat.

Seit 2009 ist Wolfgang Karl Vorsitzender der Konrad-Zuse-Gesellschaft e. V.

// Überblick und Allgemeines

Die Forschungsgruppe Rechnerarchitektur und Parallelverarbeitung befasst sich mit heterogenen parallelen Rechnerarchitekturen, die durch ein hohes Maß an Parallelverarbeitung auf den verschiedenen Systemebenen sowie durch Diversität beispielsweise auf Knotenebene durch Multi-core Prozessorarchitekturen, die durch Beschleuniger-Architekturen ergänzt werden, gekennzeichnet sind. Für den Anwendungsprogrammierer stellt sich die Aufgabe der effizienten Parallelisierung seiner Anwendung mit Hilfe (zum Teil verschiedenen) parallelen Programmiermodellen, zum anderen erfordern die unterschiedlichen Programmierschnittstellen der Zielressourcen umfangreiche und detaillierte Kenntnisse der zugrundeliegenden Zielplattform für deren effiziente Nutzung. Das Ziel ist, Methoden und Werkzeuge zu erforschen, mit denen die Komplexität der zugrundeliegenden Zielplattform vor dem Anwendungsprogrammierer verborgen werden kann und gleichzeitig eine effiziente Nutzung der verfügbaren Rechenressourcen ermöglicht wird.

Mit HALadapt ist ein Laufzeitsystem für heterogene parallele Rechnerarchitekturen entstanden und weiterentwickelt worden, das von der zugrundeliegenden Hardware abstrahiert und unabhängig vom Programmierer für eine Aufteilung und Abbildung der Arbeitslast auf die

zur Verfügung stehenden Zielressourcen sorgt. Gemäß den Prinzipien der Selbstorganisation beobachtet HALadapt das Laufzeitverhalten von Programmen und trifft auf der Basis der gesammelten Informationen Entscheidungen über die Anwendungsverteilung im Hinblick der aktuellen Situation des Systems und seiner Umgebung und unter Berücksichtigung mehrerer Optimierungsziele wie Laufzeit, Energie und Temperatur. Damit die Abbildungsentscheidung möglichst effizient und schnell getroffen wird, verwendet HALadapt ein Regelsystem zur Gewichtung der Systemoptimierungsziele. Diese Regeln beinhalten Vorhersagen über das zukünftige Systemverhalten und ermöglichen daher eine proaktive und nicht nur reaktive Abbildungsentscheidung. Um das Regelsystem einzulernen, verwendet HALadapt einen Reinforcement Learning Ansatz. Zusätzlich bietet HALadapt die Möglichkeit mehrere parallele Prozesse auf einem Rechenknoten mittels eines Co-Scheduling Mechanismus zu koordinieren. Dies bietet die Möglichkeit, Anwendungen, welche die vom System zur Verfügung gestellte Parallelität im Einzelnen nicht vollständig nutzen können, zu kombinieren und somit die Systemeffizienz zu verbessern.

Trotz einer möglichst guten Anpassung von Algorithmen an die zugrunde liegende heterogene parallele Hardware, existieren Anwendungsbereiche, in denen die Leistungsfähigkeit oder die Energieaufnahme des betrachteten Systems nicht zufriedenstellend ist. Einen weiteren Schwerpunkt bildet deshalb die Erforschung von Approximate Computing Ansätzen. Diese betrachten die gezielte Approximation in Systemen, um eine Abwägung zwischen Berechnungsgüte und benötigten Ressourcen gezielt steuern zu können. Hierbei wird die Genauigkeit der Ergebnisse einer Berechnung als Parameter in einem System berücksichtigt, so dass unter tolerierbaren Verlust der Genauigkeit Optimierungsziele wie Energieverbrauch, Rechenleitung oder Einhaltung von Echtzeitbedingungen verbessert werden können. Approximate Computing Ansätze können in Software- oder in Hardware integriert werden. Das sinnvolle Zusammenspiel verschiedener Verfahren in einem System zu erforschen ist ein wesentliches Ziel der Arbeiten in diesem Bereich. So werden neue genauigkeitsbewusste Ansätze im Bereich des wissenschaftlichen Rechnens erforscht. Für Anwendungen aus den Bereichen der Bildverarbeitung oder des maschinellen Lernens kann die gezielte Ausnutzung inhärenter Toleranzen hinsichtlich approximierter Berechnungen sinnvoll sein.

Ein Beispiel ist ein neuer Ansatz zur Simulation von Strömungen mit Hilfe von Neuronalen Netzen (NN). Dieser basiert auf einer Bild-zu-Bild Translation und ermöglicht eine schnelle und realitätsnahe visuelle Darstellung der Strömung, ohne eine aufwändige Berechnung der Strömungsparameter mittels numerischer Löser durchführen zu müssen. Weiterhin wird in diesem Bereich die effiziente Umsetzung von neuartigen NN-Architekturen erforscht. Als Kriterium wird die Unterbrechbarkeit dieser NN im Hinblick auf die Abwägung der Genauigkeit des Ergebnisses und die Einhaltung von Echtzeitbedingungen untersucht.

// Mitarbeiterinnen und Mitarbeiter

Verwaltungspersonal
Gull-Nida Amjad

Wissenschaftliches Personal
Thomas Becker
Markus Hoffmann
Roman Lehmann
Rebecca Seelos

// Website
capp.itec.kit.edu