



BSc/MSc Arbeit: Statische Auflösung von Überladenen Operatoren in Matlab

Automatisches Differenzieren^[1] (AD) ist eine Methode, numerische Programme automatisiert so zu modifizieren, dass die Ableitungen der ursprünglichen Berechnungen effizient bestimmt werden können. Das Werkzeug ADiMat (Automatisches Differenzieren für Matlab) wird am Fachgebiet SC entwickelt und ermöglicht AD von Matlab-Funktionen.

Insbesondere im Vorwärtsmodus ist es oft von Interesse zahlreiche Richtungsableitungen gleichzeitig zu berechnen. Die Laufzeit des mit AD erstellten Programms steigt linear mit der Anzahl der Richtungsableitungen. In ADiMat ist die Behandlung der Richtungsableitungen in spezielle Ableitungsklassen ausgelagert. Eine solche Klasse stellt sich von außen gesehen z.B. als Matrixobjekt dar, beinhaltet jedoch intern mehrere Matrizen, eine für jede Ableitungsrichtung. Das Auflösen der überladenen Operatoren der Klasse findet zur Laufzeit statt, da es sich bei Matlab um eine interpretierte, dynamisch getypte Sprache handelt. Andererseits ist es durch statische Analyse des Ableitungscode möglich den aufzurufenden Operator vorherzusagen. Bereits getätigte Forschungen haben gezeigt, dass ein einfacher Funktionsaufruf in Matlab im Vergleich zu einem überladenen Operator deutlich schneller ist.

Im Rahmen dieser Arbeit sollen das Werkzeug ADiMat um einen Nachbearbeitungsschritt erweitert werden, der alle Operationen, welche Ableitungsklassen involvieren, statisch auflöst und durch Aufrufe von Funktionen ersetzt, welche die gleiche Funktionalität erfüllen. Dieser Nachbearbeitungsschritt soll idealerweise sowohl auf Ableitungscode im Vorwärtsmodus aus auch im Rückwärtsmodus eingesetzt werden können, oder sogar von dritten Matlab-Nutzern für die effizientere Nutzung von beliebigen Klassen verwendet werden können. Daher sollen verschiedene Möglichkeiten zur Festlegung des Klassentyps von Variablen angeboten werden. Z.B. haben im AD-Code der von ADiMat generiert wird die Ableitungsvariablen einen bestimmten Namenspräfix. Der Nachbearbeitungsschritt kann wahlweise basierend auf einer AST-Repräsentation in C++ oder XML implementiert werden.

Voraussetzungen sind

- grundlegende Programmierkenntnisse, idealerweise auch in Matlab
- grundlegende Kenntnisse in Compiler-Techniken

Im Rahmen dieser Arbeit werden Sie

- sich Wissen erarbeiten über
 - Automatische Codegenerierung und Klassen in Matlab
 - Automatisches Differenzieren
- eine Abschlussarbeit im Umfang von ca. 30-40 Seiten verfassen.

Dieses Thema kann sowohl als Bachelor- als auch als Master-Arbeit bearbeitet werden.

[1] Richard D. Neidinger. 2010. *Introduction to Automatic Differentiation and MATLAB Object-Oriented Programming*, SIAM Review, Vol. 52, No. 3, pp. 545-563.

Fachgebiet Scientific
Computing



Dipl.-Inform.
Johannes Willkomm

Mornewegstr. 30
64293 Darmstadt

Tel. +49 6151 16 - 2563
Fax +49 6151 16 - 75577
johannes.willkomm@sc.informatik.tu-
darmstadt.de

Datum
09.08.2012