Verifizierte globale Optimierung auf Parallelrechnern

Zur Erlangung des akademischen Grades eines

DOKTORS DER NATURWISSENSCHAFTEN

von der Fakultät für Mathematik der
Universität Karlsruhe (TH)
genehmigte

DISSERTATION

von

Dipl.-Math. Andreas Wiethoff

aus Münster (Westf.)

Tag der mündlichen Prüfung: ..... 17. Dezember 1997
Referent: ..... Prof. Dr. U. Kulisch
Korreferent: ..... H.-Doz. Dr. W. Krämer

Inhaltsverzeichnis

Einleitung

Problemstellungen, die sich in Form eines globalen Optimierungsproblems repräsentieren lassen, treten häufig bei der mathematischen Modellierung von Anwendungen aus Wirtschaft und Wissenschaft auf. Derartige Anwendungen sind in so unterschiedlichen Bereichen wie Finanzwesen, Steuerung von industriellen Produktionsanlagen, Chip-Entwurf, Strukturoptimierung, *operations research*, Molekularbiologie uvm. zu finden. Globale Optimierung ist aus mathematischer Sicht die Bestimmung und Charakterisierung des globalen Minimums (oder Maximums) einer in der Regel nichtlinearen Funktion. Häufig ist man auch an den zugehörigen Minimalstellen interessiert, an denen das globale Minimum angenommen wird. Globale Optimierung ist wesentlich „schwieriger“ als lokale Optimierung, bei der nur ein beliebiges lokales Minimum in der Nähe eines Startpunktes gesucht wird. Viele Anwendungsprobleme sind durch eine sehr große Zahl an lokalen Minima gekennzeichnet. Es ist daher nicht praktikabel, alle lokalen Minima zu bestimmen und unter diesen dann das kleinste herauszufiltern. Stattdessen müssen globale Informationen verwendet werden, um den Suchaufwand zu beschränken.

Intervallmethoden sind gut geeignet, um mit vergleichsweise wenig Aufwand derartige globale Informationen zu erhalten. Intervalle werden auf dem Rechner durch zwei Maschinenzahlen dargestellt. Dennoch bezeichnet ein Intervall eine in der Regel unendliche Teilmenge der reellen Zahlen, nicht nur die endliche Menge der zwischen den beiden Schranken liegenden Maschinenzahlen. Intervalle sind somit Träger globaler Information. Verwendet man Intervalle in einem strikten mathematischen Kalkül, der in Abschnitt 1.4 eingeführten Intervallarithmetik, so lassen sich mathematisch gesicherte Aussagen auf dem Rechner beweisen. Beispielsweise kann durch eine einzige Maschinenintervallauswertung (sie ist nur etwa doppelt so auf-

wendig wie eine entsprechende reelle Auswertung) einer Zielfunktion bei der globalen Optimierung eine gesicherte Aussage über den Wertebereich über dem gesamten Intervall getroffen werden. Alle auf dem Rechner auftretenden Rundungsfehler werden dabei durch die Maschinenintervallarithmetik automatisch miterfaßt.

Natürlich lassen sich „klassische“ numerische Algorithmen zur globalen Optimierung nicht ohne weiteres mit der Intervallarithmetik kombinieren. Um effizient mathematisch gesicherte Ergebnisse zu bestimmen, sind andere, neuartige Algorithmen erforderlich. Für die globale Optimierung ohne Nebenbedingungen wird in dieser Arbeit ein Algorithmus eingesetzt, der in seiner ursprünglichen Form zuerst von Hansen [23] sowie in einer sehr ähnlichen Form von Moore [57] und Skelboe [74] angegeben wurde. Viele Autoren haben in den vergangenen Jahren zahlreiche effizienzsteigernde Modifikationen des ursprünglichen Algorithmus betrachtet, unter anderem verschiedene Varianten des erweiterten Intervall-Newton-Schritts. Im ersten Teil dieser Arbeit wird ein neuartiger Algorithmus angegeben, der den durch die notwendige Hessematrixauswertung aufwendigen Intervall-Newton-Schritt adaptiv gesteuert selektiv einsetzt. Dadurch wird der Intervall-Newton-Schritt nur dann ausgeführt, wenn durch ihn tatsächlich ein Fortschritt zu erwarten ist. Außerdem wird bei dem hier vorgestellten Algorithmus zur globalen Optimierung in bestimmten Phasen der Berechnung eine andere Aufteilung der aktuell zu untersuchenden Teilbox vorgenommen. Dies führt ebenfalls zu Effizienzsteigerungen im Vergleich zu bisher in der Literatur behandelten Algorithmen. Insgesamt liefert der Algorithmus gesicherte Einschließungen für das globale Minimum und alle globalen Minimalstellen. Zusätzlich wird anschließend versucht, die lokale Eindeutigkeit der Minimalstellen auf dem Rechner automatisch zu verifizieren. Der hier behandelte Algorithmus zur globalen Optimierung fällt also in die Klasse der sogenannten Verifikationsalgorithmen.

Parallelrechner gehören zur Klasse der Höchstleistungsrechner. Sie werden eingesetzt, um bislang aus Laufzeitgründen seriell unlösbare Probleme zu bewältigen. Bei einem Parallelrechner bearbeiten mehrere Prozessoren gemeinsam ein Problem. Um die durch einen Parallelrechner zur Verfügung gestellte Rechnerleistung effizient nutzen zu können, sind speziell angepaßte Programme notwendig. Auch für die Problemklasse der verifizierten globalen Optimierung sind Parallelrechner einsetzbar. Im zweiten Teil dieser Arbeit wird eine Adaption des oben skizzierten neuartigen Algorithmus zur globalen Optimierung für Parallelrechner vorgenommen. Der dabei entste-

hende parallele Algorithmus ist vollständig verteilt. Alle Prozessoren des Parallelrechners bearbeiten gleichberechtigt Teilprobleme und tauschen Informationen über den Fortschritt während des Berechnungsprozesses aus. Der Algorithmus ist somit gut skalierbar. Implementiert wurde der parallele Algorithmus in C–XSC, einer C++ Klassenbibliothek für wissenschaftliches Rechnen und MPI, der Standardbibliothek für *message passing* Programme in der Programmiersprache C. Konkret wurden in dieser Arbeit Testprobleme auf bis zu 96 Prozessoren gerechnet. Dabei konnte für viele Testprobleme eine hohe Beschleunigung erzielt werden.

Die vorliegende Arbeit ist wie folgt gegliedert: Im ersten Kapitel erfolgt eine kurze Einführung in die Grundlagen des wissenschaftlichen Rechnens, soweit sie in dieser Arbeit benötigt werden. Nach der Vorstellung der verwendeten Bezeichnungen sowie einiger grundlegender Definitionen und Sätze geben wir einen kurzen Überblick über die mathematisch fundierte Rechner- und Intervallarithmetik sowie die erweiterte Intervallarithmetik, die im Intervall-Newton-Schritt eingesetzt wird. Anschließend wird die automatische Differentiation vorgestellt, die die effiziente Berechnung von Einschließungen von Gradienten und Hessematrizen ermöglicht, ohne formal zu differenzieren oder die numerischen Schwierigkeiten herkömmlicher dividierter Differenzen in Kauf nehmen zu müssen.

Kapitel 2 gibt dann einen Überblick über die in der vorliegenden Arbeit verwendeten Grundlagen des parallelen Rechnens. Nach einer Klassifizierung von Parallelrechnerarchitekturen und der Bereitstellung der benötigten Definitionen und Begriffe zur Leistungsmessung und Beurteilung von parallelen Algorithmen wird der als Grundlage für den parallelen Algorithmus zur verifizierten globalen Optimierung dienende Lastverteilungsalgorithmus für parallele *Branch and Bound*-Algorithmen vorgestellt. Anschließend werden noch einige Details zur verwendeten Hard- und Softwareumgebung angegeben.

Aufbauend auf diesen Grundlagen wird in Kapitel 3 zunächst der neuartige serielle Algorithmus zur verifizierten globalen Optimierung hergeleitet. Dabei wird das behandelte Problem genauer spezifiziert sowie ein einfacher Grundalgorithmus zur Lösung angegeben. Anschließend werden zahlreiche Modifikationen und Erweiterungen zur Beschleunigung des Grundalgorithmus vorgestellt. Diese Erweiterungen werden in einem ersten Algorithmus, der aufgrund seiner Ähnlichkeit mit dem in [21] angegebenen Algorithmus als „*Toolbox*-ähnlich“ bezeichnet wird, zusammengeführt. Dieser *Toolbox*-

ähnliche Algorithmus dient als Vergleichsgrundlage für den neuartigen Algorithmus, der anschließend dargelegt wird. Eine große Anzahl an Testproblemen aus der Literatur zur globalen Optimierung wird dann für numerische Tests zum Vergleich verschiedener Varianten beider Algorithmen verwendet. Dabei wird in einer ausführlichen Studie die Laufzeit sowie der Aufwand für Funktions-, Gradienten- und Hessematrixauswertungen und der benötigte Speicherplatz miteinander verglichen. Hier zeigen sich deutlich die Effizienzvorteile des neuartigen Algorithmus.

Im zweiten Teil des dritten Kapitels wird dann der parallele Algorithmus zur verifizierten globalen Optimierung entwickelt. Dabei werden zunächst andere Ansätze aus der Literatur vorgestellt, danach der neue, vollständig verteilte parallele Algorithmus hergeleitet. Auch hier folgt ein Abschnitt über die Ergebnisse der numerischen Tests sowie Effizienzbetrachtungen.

Kapitel 4 gibt eine kurze Zusammenfassung der Ergebnisse dieser Arbeit sowie einen Ausblick über denkbare Erweiterungsmöglichkeiten und Ergänzungen.

Im Anhang findet sich eine detaillierte Auflistung der untersuchten Testprobleme. Zu jedem Testproblem wird die Zielfunktion, der Startbereich sowie die berechneten Ergebnisse angegeben. Außerdem enthält der Anhang eine wichtige Verbesserung der Hessematrixarithmetik aus der *Toolbox* sowie eine Auflistung aller Funktionen, die zur Verwendung von C–XSC und MPI auf Parallelrechnern notwendig sind.

Die Programme, die für die numerischen Tests verwendet wurden, finden sich im Internet unter:

`http://www.uni-karlsruhe.de/~Andreas.Wiethoff/globopt.html`

Kapitel 4

Zusammenfassung und Ausblick

Zur effizienten Lösung globaler Optimierungsprobleme wurde in dieser Arbeit ein neuartiger Algorithmus zur verifizierten globalen Optimierung auf herkömmlichen seriellen Rechnern und auf diesen aufbauend ein vollständig verteilter und somit skalierbarer paralleler Algorithmus entwickelt. Die numerischen Ergebnisse in Abschnitt 3.2.10 belegen, daß durch den neuartigen seriellen Algorithmus deutliche Einsparungen bei Laufzeit und Auswertungsaufwand (fast doppelt so schnelle Laufzeit und mehr als halbierter Auswertungsaufwand) im Vergleich zu bisherigen verifizierenden Algorithmen für die globale Optimierung erzielt werden konnten. Entscheidende Merkmale für die Effizienzvorteile des neuartigen seriellen Algorithmus sind der adaptiv gesteuerte selektive Einsatz des Intervall-Newton-Schritts zur Reduzierung der aufwendigen Hessematrixauswertungen der Zielfunktion, die Verwendung eines Sortierungsvektors bei jeder Aufteilung der aktuell zu untersuchenden Teilbox, die geänderte Aufteilungsstrategie (das *Boxing*-Verfahren) der aktuellen Teilbox in bestimmten Situationen während der Berechnung sowie der Einsatz von Multisektion statt der bislang häufig verwendeten Bisektion.

Der in dieser Arbeit vorgestellte parallele Algorithmus zur verifizierten globalen Optimierung verwendet den neuartigen seriellen Algorithmus zur Bearbeitung von Teilboxen auf jedem Prozessor. Er ist vollständig verteilt und daher gut skalierbar. Die Kommunikation mit einem zentralen I/O-Prozessor findet nur in der Start- und in der Endphase des parallelen Algorithmus statt. Durch den Einsatz eines Arbeits- und eines Kontrollpro-

zesses sowie einer damit verbundenen *Feedback*-Strategie auf jedem Prozessor werden *Trashing*-Effekte vermieden und so eine hohe Effizienz bei der Parallelisierung erreicht. Für hinreichend schwere Optimierungsprobleme wurde eine hohe Beschleunigung bei Einsatz von bis zu 96 Prozessoren erzielt. Durch die Parallelisierung wird die Lösung von Optimierungsproblemen ermöglicht, deren Lösung auf herkömmlichen Rechnern entweder an zu hohem Speicherplatzbedarf oder an zu großer Laufzeit scheitert (vgl. Testprobleme WK (Abschnitt 3.3.4), KOW und HM4). Es ist also durch die Parallelisierung möglich, wesentlich schwierigere bzw. größere Probleme anzugehen als bisher.

Für zukünftige Arbeiten ist sicher eine Erweiterung des seriellen und parallelen Algorithmus für globale Optimierungsprobleme mit Nebenbedingungen denkbar. Zu dieser Problemstellung gibt es in der Literatur einige Arbeiten (z.B. in [24] oder [67]), die dabei verwendet werden können. Daneben sind weitere Verbesserungen an einzelnen Details des seriellen Algorithmus möglich. Es kann noch der Einfluß von verschiedenen Werten für ϵ_{Box} im *Boxing*-Schritt (Algorithmus 3.10) und die Unter- und Oberschranken für ϵ_{Newton} systematisch in einer Studie untersucht werden. Für die Steuerung des selektiven Einsatzes des Intervall-Newton-Schritts ist es auch denkbar, das jeweilige aktuelle ϵ_{Newton} und einen Indikator, ob der letzte Intervall-Newton-Schritt erfolgreich war oder nicht, zusammen mit der aktuellen Teilbox Y und der Unterschranke der Intervallauswertung $\underline{F(Y)}$ in der Arbeitsliste abzuspeichern. Dadurch kann ein besserer Bezug zwischen dem verwendeten ϵ_{Newton} und der untersuchten Teilbox hergestellt werden. Dieses Vorgehen ist jedoch recht aufwendig zu implementieren.

Globale Optimierungsprobleme mit einer nichtdifferenzierbaren Zielfunktion f (f kann etwa $|\cdot|, \min, \max,$ **if**...**then**...**else** enthalten) können mit den in dieser Arbeit beschriebenen Algorithmen nicht behandelt werden, da für die in Abschnitt 3.2.6 beschriebenen beschleunigenden Methoden die zweimalige Differenzierbarkeit von f vorausgesetzt wird. Es ist zwar möglich, auf diese Methoden zu verzichten, jedoch werden dann die Berechnungszeiten wesentlich größer.

Ein kürzlich entwickelter, vielversprechender Ansatz ist die Verwendung eines Algorithmus, der auf einer Steigungsarithmetik (siehe etwa [1], [43], [64]) sowie eines anderen *Pruning*-Schritts zur Verkleinerung der aktuellen Box beruht. Dabei ist im Prinzip nur die Funktion BearbeiteBox() im seriellen Algorithmus komplett auszutauschen. Der für BearbeiteBox() notwendige andere *Pruning*-Schritt ist für den eindimensionalen Fall in [72] dar-

gestellt. Eine Grundidee für den *Pruning*-Schritt findet sich bereits in [25]. Dort wird jedoch die dreimalige Differenzierbarkeit der Zielfunktion vorausgesetzt. Der kompliziertere mehrdimensionale Fall wird in [73] behandelt. Prinzipiell läßt sich also dieser andere serielle Algorithmus mit dem in dieser Arbeit beschriebenen parallelen Lastverteilungsalgorithmus zu einem neuen parallelen Algorithmus zur verifizierten globalen Optimierung für nichtdifferenzierbare Zielfunktionen f kombinieren. Ein solches Vorgehen erscheint recht erfolgversprechend.