

JÄGER, C. UND RATZ, D.

## Ein kombiniertes Verfahren zur Einschließung aller Lösungen eines nichtlinearen Systems von algebraischen Gleichungen

Ein Verfahren zur Berechnung von Einschließungen *aller* Lösungen eines nichtlinearen polynomialen Gleichungssystems wird vorgestellt. Das Verfahren kombiniert eine Intervallform des Buchberger-Algorithmus mit einer Variante des eindimensionalen Hansen-Algorithmus. Testergebnisse einer portablen PASCAL-XSC Implementierung unseres Verfahrens demonstrieren, daß die Methode erheblich schneller arbeitet als das Hansen-Verfahren oder ähnliche Methoden für nichtlineare Gleichungssysteme.

### 1. Das kombinierte Verfahren

**Input:** System von multivariaten Polynomen

**Intervallversion des Buchberger-Algorithmus**

Zwischenergebnis: Dreieckssystem von Polynomen

**Variante des Hansen-Algorithmus für Funktionen in einer Variable**

**Output:** Liste alle Nullstellen

Das gegebene algebraische Gleichungssystem wird zunächst mit Hilfe einer Intervallversion des Buchberger-Algorithmus auf Dreiecksgestalt transformiert, ohne dabei die algebraischen Eigenschaften des gegebenen Systems, wie z. B. Anzahl oder Vielfachheit der Nullstellen zu verändern. Für dieses gestaffelte nichtlineare Intervallgleichungssystem berechnet eine Variante des Hansen-Algorithmus dann die Liste der Einschließungen aller Nullstellen und damit auch der Lösungen des Ausgangssystems.

### 2. Der Buchberger-Algorithmus und seine Intervallversion

Ausgehend von einer Polynomdarstellung in der Form  $f = a_0 \cdot \text{MON}_0 + \dots + a_l \cdot \text{MON}_l$  mit  $a_0, \dots, a_l \in \mathbb{R}$  und Monomen  $\text{MON}_i = x_1^{p_1} \cdots x_n^{p_n}$  ( $p_j \in \{0, 1, 2, \dots\}$  für  $j = 1, \dots, n$ ), wird aufbauend auf eine lexicographische Ordnung für die Variablen  $x_j$  und die Monome  $\text{MON}_i$  sowie auf eine *Normalisierung*  $a_0 = 1$  eine eindeutige Darstellung festgelegt, in der  $\text{MON}_0$  auch GPP (*größtes Potenzprodukt*) genannt wird (vgl. [2]). Führt man weiterhin das *kleinste gemeinsame Vielfache zweier Monome* (KGV) ein, so läßt sich für zwei von Null verschiedene normalisierte Polynome  $f_i$  und  $f_j$  das sogenannte *S-Polynom* (vgl. [1]) definieren als

$$S(f_i, f_j) = \frac{\text{KGV}(\text{GPP}_i, \text{GPP}_j)}{\text{GPP}_i} \cdot f_i - \frac{\text{KGV}(\text{GPP}_i, \text{GPP}_j)}{\text{GPP}_j} \cdot f_j.$$

S-Polynome, gebildet aus zwei Polynomen des Gleichungssystems, können zusätzlich in das System aufgenommen werden, ohne daß seine algebraischen Eigenschaften (Nullstellen) verändert werden. Sie bilden die Grundlage des *Buchberger-Algorithmus*, der in einfacher Form wie folgt formuliert werden kann:

**Input** System von multivariaten Polynomen  $G$   
 $F := G$ ;  $L_F := L_G$ ;  $B := \{(i, j) \mid 1 \leq i < j \leq L_G\}$ ;  
**while exists**  $(i, j) \in B$  **do**  
 $h := S(f_i, f_j)$ ;  
**if**  $h \neq 0$  **then**  $F := F + h$ ;  $L_F := L_F + 1$ ;  
 $B := B \cup \{(k, L_F) \mid 1 \leq k < L_F\}$ ;  
 $B := B - (i, j)$ ;  
**Output** Dreieckssystem enthalten im System  $F$

Zunächst werden in  $B$  alle möglichen Polynompaarungen aus  $G$  eingetragen, die zur S-Polynombildung herangezogen werden können. Für jedes Paar aus  $B$  wird die S-Polynombildung ausgeführt. Falls das erzeugte Polynom von Null verschieden ist, wird es in  $F$  aufgenommen und alle sich dadurch ergebenden neuen Paarungen mit Polynomen aus  $F$  werden in  $B$  eingetragen. Dies wird solange wiederholt, bis die Menge  $B$  leer ist. Dabei bezeichnen  $f_i$  bzw.  $h$  Polynome aus  $F$  bzw. neu generierte Polynome und  $L_F$  bzw.  $L_G$  den Index des letzten Polynoms in  $F$  bzw.  $G$ .

In der Praxis wird eine vereinfachte S-Polynombildung (genannt *M-Reduktion*) nach jeder S-Polynombildung ausgeführt, was zu einer erheblichen Zeitersparnis führt. Außerdem werden weitere Kriterien für überflüssige Paarungen in  $B$  und für mögliche Eliminationen von Polynomen aus  $F$  angewandt [1,2], so daß  $F$  Dreiecksgestalt annimmt. Die *Theorie der Gröbnerbasen* [2] bildet die theoretische Grundlage und garantiert die Terminierung des Verfahrens.

Das exakte Buchberger-Verfahren in Computeralgebrasystemen ist oft zu langsam. Um kürzere Laufzeiten zu erzielen und außerdem die Kommunikation mit numerischen Verfahren zu ermöglichen ist eine Gleitkomma-Implementierung wünschenswert, wirft jedoch sofort die bekannten Probleme durch Darstellungs- und Rundungsfehler auf. Unter Verwendung der vom Intervall-Gauß-Algorithmus (IGA) bekannten Techniken, konnten wir einen Intervall-Buchberger-Algorithmus (IBA) entwickeln, der eine garantierte Einschließung des exakten Dreieckssystems berechnet [5].

### 3. Ein Variante des Hansen-Algorithmus

Den zweiten Teil unseres Verfahrens bildet der *NLTSS*, der Nonlinear Triangular System Solver.

**Input** Gleichungssystem  $F = (f_1, \dots, f_n)$  in Dreiecksgestalt  
Setze  $k := n$  und starte **RecursiveHansen**(k);

#### Procedure RecursiveHansen(k)

Berechnung des Startintervalls für  $f_k$  mit Gershgorin;  
Start des *Hansen-Algorithmus* für  $f_k(x_k)$  zur Berechnung der  
Einschließungen aller Nullstellen  $[y_k]_1, [y_k]_2, \dots, [y_k]_{N_k}$ ;

**if**  $k = 1$  **then return**

**else for**  $i := 1$  **to**  $N_k$  **do**

Ersetze  $x_k$  durch  $[y_k]_i$  in den Polynomen  $f_1, \dots, f_{k-1}$ ;

Starte **RecursiveHansen**( $k - 1$ );

**Output** Liste der Einschließungen aller Nullstellen des Systems

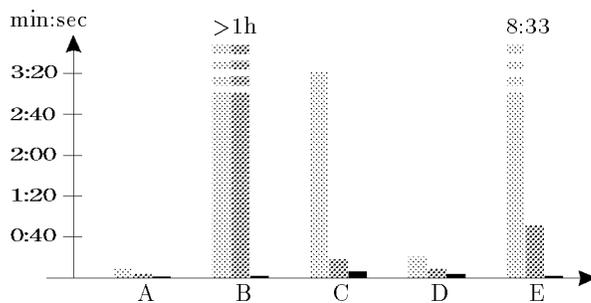
Ausgehend von der letzten Gleichung  $f_n$  im Dreieckssystem wird der eindimensionale Hansen-Algorithmus [4] angewandt, um alle Nullstellen dieses Polynoms zu berechnen. Hierbei dient der Satz von Gershgorin zur Berechnung des benötigten Startintervalls. Die so erhaltenen Einschließungen für die Nullstellen werden nacheinander in die weiteren Polynome  $f_{n-1}, \dots, f_1$  eingesetzt. Hierbei entsteht jedesmal wiederum ein Intervall-Polynom in nur einer Variable, das seinerseits wieder auf dieselbe Art bearbeitet werden kann.

### 4. Bewertung und Beispiele

Nach der Reduktion des Gleichungssystems auf Dreiecksgestalt, können die benötigten Startintervalle, die alle Nullstellen enthalten, automatisch durch den Algorithmus berechnet werden (Satz von Gershgorin). Vorabinformationen über die Lage der Nullstellen des Systems werden so überflüssig. Desweiteren kann noch vor dem Start des NLTSS über die Lösbarkeit bzw. die Gestalt der Lösungsmenge des Systems entschieden werden.

Zukünftige Forschungsarbeit ist zu investieren, um die, wie beim IGA, auch beim IBA für ungünstige Systeme auftretenden Probleme (wie Overflow oder Aufblähungen in den führenden Intervallkoeffizienten) zu lösen.

Wir testeten unsere PASCAL-XSC Implementierung unter Verwendung einiger wesentlicher Verbesserungen im Buchberger-Algorithmus (Suche nach optimaler Ordnung, Pivotisierungen bei den S-Polynombildungen) an den nachfolgenden Beispielen. Details zu den genannten Systemen finden sich in [5].



- A. Schnitt dreier Kugeln (Dim 3)
- B. Gleichungssystem von Brown (Dim 5)
- C. Gleichungssystem höheren Grades (Dim 3/Grad 9)
- D. Feigenbaum (Dim 3)
- E. Gleichungssystem von Powell (Dim 3)

Die Graphik zeigt die Laufzeit unseres kombinierten Verfahrens (Zeit für IBA plus Zeit für NLTSS) ■ im direkten Vergleich mit den Laufzeiten der nichtlinearen Gleichungssysteml6ser aus dem PASCAL-XSC Demoprogramm [6] ☼ und der Toolbox-Software [3] ☼.

### 5. Literatur

- 1 BUCHBERGER, B.: *Ein algorithmisches Kriterium f6r die L6sbarkeit eines algebraischen Gleichungssystems*. Aequationes Mathematicae 4, 374–383, 1970.
- 2 DAVENPORT; J. H.; SIRET, Y.; TOURNIER, E.: *Computer Algebra, Systems and Algorithms for Algebraic Computation*. Academic Press, New York, 1988.
- 3 HAMMER, R.; HOCKS, M.; KULISCH, U.; RATZ, D.: *Numerical Toolbox for Verified Computing I – Basic Numerical Problems*. Springer-Verlag, Heidelberg, 1993.
- 4 HANSEN, E.: *A Globally Convergent Interval Method for Computing and Bounding Real Roots*. BIT 18, 415–424, 1978.
- 5 JÄGER, C.; RATZ, D.: *A Combined Method for Enclosing All Solutions of Nonlinear Systems of Polynomial Equations*. To appear in: Interval Computations, 1994.
- 6 KLATTE, R.; KULISCH, U.; NEAGA, M.; RATZ, D.; ULLRICH, CH.: *PASCAL-XSC – Language Reference with Examples*. Springer-Verlag, New York, 1992.

*Anschrift:* JÄGER, C.; RATZ, D., Universität Karlsruhe, Institut für Angewandte Mathematik, D-76128 Karlsruhe