

Implementation of CGS on small devices

– software demonstration –

Akira Suzuki, Kobe University

ACA2008@RISC, Linz
2008, July 29

Introduction

CGS is a kind of parametric Gröbner basis, which is introduced by V. Weispfenning. We introduced a simple and compact algorithm to compute it.

In this demo, I show implementations of it to small devices, e.g., PDA, smartphone. I also showcase several platforms to implement CGS with tiny comparison.



Contents

- **What is CGS**
 - **Gröbner Bases**
 - **Parametric Gröbner Bases**
 - **the Simple Algorithm for CGS**
- **Several Languages to Implement CGS**
 - **Risa/Asir**
 - **Java**
 - **Python**
- **Several Platforms to Implement CGS**
 - **Linux Zaurus**
 - **Symbian OS**
 - **Windows Mobile**
- **Demonstration**

What is CGS? – Gröbner Basis –

The concept of Gröbner basis is the powerful one to treat ideals on polynomial rings. It is a generalization of both

- (1) GCD of univariate polynomial, and
- (2) Gaussian elimination for linear systems.

In 1965, Buchberger introduced it along with the algorithm to compute it.

Gröbner basis has many applications in Science and Engineering. And many of computer algebra systems have the Buchberger algorithm. Please refer the “Session on Gröbner Bases and their Applications” for the detail.

In this talk, we concentrate on the polynomial ring $\mathbb{Q}[X_1, \dots, X_n]$ where X_1, \dots, X_n are distinct variables.

What is CGS – Parametric Gröbner Basis –

From the viewpoint of computer algebra, it is useful and important to introduce parameters to equations. For example, we can see **when** a given polynomial system have a root using parametric polynomial systems.

In this talk, we set some k with $1 \leq k \leq n$ and consider X_1, \dots, X_{k-1} are (standard) variable and X_k, \dots, X_n are parameters. We concentrate on **CGS** (comprehensive Gröbner system) as the most commonly used parametric Gröbner bases. CGS was introduced by Weispfenning.

What is CGS – Example –

Example. The CGS of $[x_1^2 - x_4, x_2^3 - x_5, x_1 + x_2 - x_3]$ wrt lex. order with $x_1 > x_2 > x_3$ where x_1, x_2, x_3 are variables and x_4, x_5 are parameters is:

$$(x_5 * x_4 * (64 * x_4^3 + 27 * x_5^2) \neq 0)$$

$$[-x_3^6 + 3 * x_4 * x_3^4 + 2 * x_5 * x_3^3 - 3 * x_4^2 * x_3^2 + 6 * x_5 * x_4 * x_3 + x_4^3 - x_5^2, (-64 * x_4^3 -$$

$$(x_5 * x_4 \neq 0, 64 * x_4^3 + 27 * x_5^2 = 0)$$

$$[-192 * x_4 * x_3^5 - 72 * x_5 * x_3^4 + 640 * x_4^2 * x_3^3 + 624 * x_5 * x_4 * x_3^2 + 477 * x_5^2 * x_3 + 728 * x_4^3 -$$

$$(x_5 \neq 0, x_4 = 0)$$

$$[-x_3^6 + 2 * x_5 * x_3^3 - x_5^2, 3 * x_5 * x_2 + x_3^4 - 4 * x_5 * x_3, 3 * x_5 * x_1 - x_3^4 + x_5 * x_3]$$

$$(x_5 = 0, x_4 = 0)$$

$$[x_3^4, 3 * x_3^2 * x_2 - 2 * x_3^3, x_2^2 - 2 * x_3 * x_2 + x_3^2, x_1 + x_2 - x_3]$$

$$(x_4 \neq 0, x_5 = 0)$$

$$[-x_3^6 + 3 * x_4 * x_3^4 - 3 * x_4^2 * x_3^2 + x_4^3, -8 * x_4^2 * x_2 - 3 * x_3^5 + 10 * x_4 * x_3^3 - 7 * x_4^2 * x_3^2 +$$

What is CGS – Existing Implementations –

- Redlog [REDUCE] by Dolzmann-Sturm (1997)
- DPGB [Maple] by Montes (2002)
- ACGB [Risa/Asir] by Suzuki-Sato (2002)
- **SimpleCGS** by Suzuki-Sato (2006)
- CRCGS [Maple] by Montes-Wibmer (2008)


the simple algorithm for CGS

- introduced in ACA 2005, ISSAC 2006 by Suzuki-Sato.
- implemented mainly by Suzuki to **Risa/Asir**, Singular, Maple, Mathematica, **Java**, and **Python**.
- simple theory, easy implementation, and relatively fast speed.

Several Languages to Implement CGS – Risa/Asir –

Risa/Asir



- developed by Noro (Kobe Univ.)
- open source, included in OpenXM 
- it runs on many kind of OS including Linux, Windows, Mac OS X
- it runs also on Linux Zaurus.

Risa/Asir on Linux Zaurus

- Risa/Asir contains several procedure of reduced Gröbner basis
- implementation can be very simple
 - the initial version has only < 200 lines
- the code for Risa/Asir on PC works also on Linux Zaurus in many cases

Several Languages to Implement CGS – Java –

Java



- developed by Sun Microsystems
- object oriented, became open source, with some exceptions
- Java Platforms: **Java ME**/Java SE/Java EE
- can be share source/binary codes in JavaSE in almost cases

JavaME

- major configurations: **CLDC** (Connected Limited Device Configuration) and **CDC** (Connected Device Configuration)
- many mobilephone has VM for JavaME CLDC, but no BigInteger
- JavaME CDC has **java.math.BigInteger**
- **JavaME CDC PP** (Personal Profile) also includes **AWT** with some exceptions
- JavaME CDC PP has relatively high compatibility with JavaSE
- some smartpone has VM for JavaME CDC PP

Several Languages to Implement CGS – Python –



Python

- developed by Guido van Rossum
- open source, object-oriented
- it runs on many kind of OS including Mac OS, Windows, Linux, and **Windows Mobile**.

Python CE

- a version of the Python interpreter that runs on Microsoft Windows CE hand-held devices.
- so it runs on most of Windows Mobile smartphone.
- though it lacks some system related function (e.g., fork, process), many application written in Python works on Windows Mobile phone.

Several Platforms to Implement CGS – Linux Zaurus –

Linux Zaurus (SHARP Zaurus SL Series)

- manufactured by SHARP CORPORATION
- Linux OS 2.4 / Qtopia (Trolltech)
- SL-5500 / 6000 / A300 / B500 / C700 / C760 / C860 / C1000 / C3000 / C3100 / C3200
- released during 2002-2006 and production discontinued...
- Fujimoto ported Risa/Asir to Linux Zaurus



SL-C3000

- released at Nov. 2004
- Intel XScale (ARM) PXA270 416MHz
- SDRAM 64MB / HDD 4GB / LCD 640x480 (touchpanel)
- 124 x 87 x 25 mm / 298g

Several Platforms to Implement CGS – Symbian OS –

Symbian OS

- major interfaces: S60 (Nokia) and UIQ (Motrola, Sony Ericsson)
- a certain kind of devices have VM of **JavaME CDC PP**



SonyEricsson M600

- released at Jun. 2006
- Symbian OS 9.1 / UIQ 3.0
- ARM9 CPU 208MHz
- RAM 60MB / FLASH Storage 80MB / LCD 240 x 320 (touchpanel)
- 107 x 57 x 15 mm / 112g

Several Platforms to Implement CGS – Windows Mobile –

Windows Mobile

- developed by Microsoft
- based on Windows CE 5.0
- has **NO** software compatibility with either Windows XP or Vista
- Python CE enables to run Python (but it is slow...)
- Mysaifu JVM enables to run JavaSE (but it is slow...)

HTC Kaiser (HTC TyTn II)

- released at Sep. 2007
- developed by HTC (Taiwan)
- Windows Mobile 6 Professional Edition
- CPU: Qualcomm MSM7200 (ARM11) 400MHz
- RAM 128MB / FLASH Storage 256MB / LCD 240 x 320 (touch-panel)
- 59 x 112 x 19 mm / 190g



Demonstration



One More Thing

One More Thing



Several Platforms to Implement CGS – iPod touch –

- developed by Apple / released at Sep. 2007
- iPhone OS 2.0 (upgraded) / released at Jul. 2008
- using Darwin foundation (open source UNIX, derived from BSD)
- jailbroken (we can use Java, Ruby, Python, and so on)
- at the current time, Java have neither AWT nor Swing, but another GUI wrapper exists
- CPU: ARM11 412MHz
- RAM: 128MB / FLASH Storage 8GB or 16GB or 32GB / LCD 480 x 320 (touch panel)
- 110 x 61.8 x 8 mm / 120g

