# 基于标签的矩阵型 Gröbner 基算法研究

潘森杉* 胡予濮 王保仓
(西安电子科技大学综合业务网国家重点实验室 西安 710071)

**摘 要**：目前基于标签的Gröbner基算法大多是Buchberger型的，涉及矩阵型算法的文献往往是为了进行复杂度分析，而不考虑实际的效率。该文从实际应用出发，给出矩阵型Gao-Volny-Wang(GVW)算法的一个实例，提出算法层次的优化设计方法。同时，该文还给出一个高效的约化准则。通过实验，该文比较了算法可用的各项准则及策略。实验结果表明，该文的矩阵型GVW实例在准则和策略的选取上是最优的。并且，矩阵型GVW在某些多项式系统(例如，Cyclic系列和Katsura系列多项式系统)下比Buchberger型GVW要快2~6倍。

**关键词**：密码学；Gröbner 基；标签；多项式；Gao-Volny-Wang (GVW)算法
**中图分类号**： TP309 **文献标识码**： A **文章编号**：1009-5896(2015)04-0881-06
**DOI**: 10.11999/JEIT140831

# Research on Signature-based Gröbner Basis Algorithms in Matrix Style

Pan Sen-shan Hu Yu-pu Wang Bao-cang
(*State Key Laboratory of Integrated Service Networks*, *Xidian University*, *Xi'an* 710071, *China*)

**Abstract**: The current signature-based Gröbner basis algorithms are mostly in Buchberger style and the researches related to matrix style often aim to analyze the complexity of algorithms. From a practical aspect, this paper provides a concrete Gao-Volny-Wang (GVW) algorithm in matrix style and presents optimization at the algorithmic level. Meanwhile, an efficient reduction criterion is given in the paper. Many popular criteria and strategies are compared by some experiments which show that the matrix version described in the paper is a combination of reasonable criteria and strategies. Moreover, the matrix-GVW is two to six times faster than the Buchberger style for some polynomial systems, *e.g.* Cyclic series and Katsura series.

**Key words**: Cryptography; Gröbner basis; Signature; Polynomial; Gao-Volny-Wang (GVW) algorithm

## 1 引言

Gröbner基是求解多元多项式系统的一个基本数学工具。求出了多项式组的Gröbner基，多项式系统的解就能很快算出。这一工具广泛应用于编码理论、密码学乃至物理、生物等自然科学领域。1965年，Buchberger[1]提出了第1个Gröbner基求解算法。1983年，为了分析Buchberger算法的复杂度，Lazard引入了线性代数的方法[2]。随后Faugère提出了基于线性代数的F4算法[3]和基于标签的F5算法[4]。F4, F5算法是目前公认的两个高效Gröbner基求解算法。Faugère和Joux在文献[5]中使用了矩阵F5算法成功地攻破了隐藏域方程(Hidden Field Equations, HFE)公钥密码系统的第1个挑战(80 bit)。虽然源码未曾公开，文献[6, 7]给出其算法的伪代码，文献[8]给出一个更详细的矩阵算法版本。矩阵F5算法的核心思想是借鉴F4中线性代数的方法来同时约化多个多项式。但是到目前为止没有任何文献专门研究矩阵型F5和Buchberger型F5孰优孰劣和如何设计准则和策略高效实现矩阵型F5。唯一可以知道的是，文献[9, 10]都有提及矩阵型F5一般要比F5算法要低效。

近年来涌现出一批基于标签的Gröbner基算法，例如Arri-Perry(AP)[11], Gao-Guan-Volny($G^2V$)算法[12]和Gao-Volny-Wang(GVW)算法[13]。它们都使用了Buchberger风格，但它们似乎又与F5算法截然不同。什么样的策略和准则对算法的帮助较大是一个值得研究的问题。

本文研究了矩阵型GVW算法的应用准则、策略和实现技巧。在设计上，为了减小约化的开销，本文采用了延迟求模的方法。在预处理过程中，本文对Macaulay矩阵进行并行化构造。除此之外，还使

2014-06-23 收到，2014-11-02 改回
国家自然科学基金(61173151, 61173152)资助课题
*通信作者：潘森杉 pansenshan@gmail.com

用了一个更高效的约化准则。在实现上，本文比较了在不同的模序、c-对选取序、重写序下的矩阵型GVW算法的实际效果。而且，实验得出了矩阵型GVW算法在某些多项式系统下要比Buchberger型GVW算法快2~6倍。最后本文还将其与突变GVW(M-GVW)算法[14]相比较，结果表明矩阵型GVW算法优势明显。

## 2 预备知识

令 $\mathcal{R}=\mathcal{K}[x_1,x_2,\cdots,x_n]$ 为域 $\mathcal{K}$ 上的 $n$ 变量多项式环， $\mathcal{M}$ 为单项式 $\left\{\prod_{i=1}^{n}x_i^{a_i}\mid a_i\in\mathbb{N}\right\}$。

$\le$ 记为 $\mathcal{M}$ 上的可允许单项式序。$\mathcal{R}$ 上的一个非零多项式 $p$ 能写成关于序 $\le$ 的单项式 $\mathcal{K}$ 线性组合：$p=\sum_{a\in A}c_ax^a$ ，其中 $c_a\in\mathcal{K}\setminus\{0\}$， $x^a\in\mathcal{M}$， $A$ 为 $\mathbb{N}^n$ 上的一个有限集。如果 $x^b$ 是集合 $\{x^a\mid a\in A\}$ 的最大单项式，那么 $\mathrm{HM}(p)=x^b$ (相应地，$\mathrm{HT}(p)=c_bx^b$，$\mathrm{HC}(p)=c_b$ )叫做 $p$ 关于 $\le$ 的首单项式(相应地，首项，首项系数)。 $p$ 的次数记为 $\deg(p)$，若 $p\neq0$ 其次数为 $\max\left\{\sum_{i=1}^{n}a_i\mid a\in A\right\}$，否则为-1。

令 $\mathcal{I}$ 为集合 $F=\{f_1,f_2,\cdots,f_d\}\in\mathcal{R}$ 生成的理想，即 $\mathcal{I}=\{u_1f_1+u_2f_2\cdots+u_df_d\mid u_1,u_2,\cdots,u_d\in\mathcal{R}\}$。考虑映射：

$$\phi:\mathcal{R}^d\to\mathcal{I}$$

$$\sum_{i=1}^{d}u_i\boldsymbol{e}_i\mapsto\sum_{i=1}^{d}u_if_i$$

其中 $\boldsymbol{e}_i$ 是 $\mathcal{R}^d$ 的第 $i$ 个单位向量使得自由 $\mathcal{R}$ -模 $\mathcal{R}^d$ 由集合 $\boldsymbol{\Sigma}=\{\boldsymbol{e}_1,\boldsymbol{e}_2,\cdots,\boldsymbol{e}_d\}$ 生成。本文在 $\mathcal{M}_d=\{m\boldsymbol{e}_i\mid m\in\mathcal{M},i\in[1,d]\}$ 上定义一个模序 $\le_s$ 与 $\le$ 适配}(见文献[15, 16]): $m\le t\Rightarrow m\boldsymbol{e}_i\le_s t\boldsymbol{e}_i$。如果不产生歧义，$\le_s$ 简记为 $\le$。 $L=\{(\max\{HM(u_i)\boldsymbol{e}_i,\le\},p)\mid\phi(\boldsymbol{u})=p\in\mathcal{I}\}$ 被叫作 $\ell$ -多项式的集合，其中 $\boldsymbol{u}=\sum_{i=0}^{d}u_i\boldsymbol{e}_i\in\mathcal{R}^d$。令 $\alpha=(\boldsymbol{s},p)\in L^*$，其中 $L^*=L\setminus(\boldsymbol{0},0)$，第 1 部分 $\boldsymbol{s}=\max\{\mathrm{HM}(u_i)\boldsymbol{e}_i,\le\}$ 叫做 $\alpha$ 的标签，记为 $\mathrm{S}(\alpha)$，第 2 部分 $p=\mathrm{poly}(\alpha)$ 是其多项式部分。不失一般性，本文假定 $\mathrm{poly}(\alpha)$ 总是首一的。同样，定义 $\alpha$ 的首单项式为 $\mathrm{HM}(\alpha)=\mathrm{HM}(p)$，次数为 $\deg(\alpha)=\deg(\mathrm{S}(\alpha))=\max\{\deg(u_i)\}$ 。如果 $\mathrm{S}(\alpha)=t\boldsymbol{e}_j$，就把 $\mathrm{idx}(\alpha)=j$ 记为其索引。s-次数(见文献[17]) 定义为 $\deg_s(\alpha)=\deg(\mathrm{S}(\alpha))+\deg(f_{\mathrm{idx}(\alpha)})$。子集 $\mathrm{Syz}=\{(\boldsymbol{s},0)\in L^*\}$ 叫做 $L$ 的合冲子模， $\mathrm{NS}=L\setminus\mathrm{Syz}$ 被叫作非合冲多项式集。令 $(\boldsymbol{s}_1,p_1)$ 和 $(\boldsymbol{s}_2,p_2)$ 是NS 中的两个非合冲 $\ell$-多项式。由形如 $(p_2\boldsymbol{s}_1-p_1\boldsymbol{s}_2,0)$ 的合冲生成的模叫做主合冲子模PS。

一个 $\ell$-多项式 $\alpha\in L$ 是可预测的，如果一个 Gröbner 基算法已经找到一个合冲 $\delta\in\mathrm{Syz}$ 使得 $\mathrm{S}(\delta)\mid\mathrm{S}(\alpha)$。算法应当避免计算这样的 $\alpha$，因此其被称为冗余的。

$\alpha\in\mathrm{NS}$ 被称为是关于 $B\subset L^*$ 首可约的，若存在一个 $\ell$-多项式 $\beta\in B$ 满足下列条件之一，

(1) $\mathrm{HM}(t\beta)=\mathrm{HM}(\alpha)$ 且 $\mathrm{S}(t\beta)<\mathrm{S}(\alpha)$；

(2) $\mathrm{S}(t\beta)=\mathrm{S}(\alpha)$ 且 $\mathrm{HM}(t\beta)<\mathrm{HM}(\alpha)$；

(3) $\mathrm{HM}(t\beta)=\mathrm{HM}(\alpha)$ 且 $\mathrm{S}(t\beta)=\mathrm{S}(\alpha)$， $t\in\mathcal{M}$。

否则， $\alpha$ 关于 $B$ 首不可约。

$\alpha-t\beta$ 这一操作叫做 S-约化 (对应的，M-约化，超首约化)，若满足条件(1) (对应的，条件(2)，条件(3))。条件(1)中， $\beta$ 和 $t\beta$ 分别被称为 S-约化子和乘性 S-约化子。有时 $t\beta$ 也简称为 S-约化子。令 $\gamma$ 为用 $\alpha$ 去 S-约化 $t\beta$ 的结果，这一过程可表示成 $\alpha\xrightarrow[B]{}\gamma$。 $\xrightarrow[B]{*}$ 是 $\xrightarrow[B]{}$ 的自反传递闭包，即反复执行约化操作直到得到一个 S-不可约的 $\ell$-多项式。若不考虑标签的大小关系，这样的约化叫做 c-约化。

由文献[11]和文献[16]的结论可得到如下的性质。

**引理 1** 令 $\alpha$ 为 NS 中的非合冲元． $\alpha$ 是可以被 $L^*$ 来 $\mathcal{M}$-约化的，当且仅当它是可以被 $L^*$ 来 S-约化的。

利用上述引理的逆否命题，本文得出如下结果。

**推论 1**[18] 令 $\alpha,\beta\in L^*$ 使 $\mathrm{S}(\alpha)=\mathrm{S}(\beta)$ 且它们非合冲。若 $\alpha$ 和 $\beta$ 都 S-不可约，则 $\mathrm{HM}(\alpha)=\mathrm{HM}(\beta)$。

由文献[17]可知，若 $\alpha\in\mathrm{NS}$ 是齐次的，$\deg(\alpha)=\deg_s(\alpha)$，否则 $\deg(\alpha)\le\deg_s(\alpha)$。

若 $\alpha$ 是 S-不可约的且 $\deg(\alpha)<\deg_s(\alpha)$，则称其为一个突变 (原始定义见文献[19])。如果输入多项式是齐次的，那么 NS 中是不存在突变的。

一个集合 $G\subset L$ 叫做模 $L$ 的 S-Gröbner 基，如果任意的非合冲 $\alpha\in\mathrm{NS}$ 能够被 $G$ 首约化。

由引理 1 可知， $\mathcal{I}$ 中的每个非零多项式可以被 $\mathcal{I}$ 的 Gröbner 基 $\{\mathrm{poly}(\alpha)\mid\alpha\in G,\mathrm{poly}(\alpha)\neq0\}$ 约化。因此 S-Gröbner 基实际上是文献[11]的一个术语，它是文献[20]中“强 Gröbner 基”的精简版。由上述定义可知，合冲 $\ell$-多项式不是 S-Gröbner 基的必要组成部分。本文说两个 $\ell$-多项式 $\alpha$ 和 $\beta$ 等价，如果 $\mathrm{S}(\alpha)=\mathrm{S}(\beta)$ 且 $\mathrm{HM}(\alpha)=\mathrm{HM}(\beta)$。显然，所有的非合冲首不可约 $\ell$-多项式组成具有最少个数的 S-Gröbner 基。文献[11]，文献[21]和文献[16]指出，非合冲不可约 $\ell$-多项式只有有限多个(不计等价)。

## 3 重写序

本文引入文献[18]中定义在 $G$ 上的关于重写准

则的概念。一个 $\ell$-多项式 $\alpha \in G$ 是标签 $\boldsymbol{s}$ 的重写子，如果 $\alpha$ 是 $G$ 中使得 $\mathrm{S}(t\alpha)=\boldsymbol{s}$ 的 $\preceq$-最大元素，其中 $t \in \mathcal{M}$，$\preceq$ 为 $G$ 上一个线性序(称为重写序)。与文献[18]相比，这里对 $\preceq$ 没有过多的限制：它只要是 $G$ 上的线性序即可。有时本文也把 $t\alpha$ 叫做 $\boldsymbol{s}$ 的重写子。$m\beta \in \mathcal{M}\times G$ 是可重写的，如果 $\mathrm{S}(m\beta)$ 的重写子是 $\alpha \neq \beta$。现在用的最多的是两种重写序是文献[18]中的 $\preceq_r$ 和文献[4] Rules 集合中元素的排序(记为 $\preceq_{\mathrm{new}}$)。它们的定义如下：

$\alpha \preceq_r \beta$ 当且仅当 $\mathrm{r}(\alpha)<\mathrm{r}(\beta)$，或者 $\mathrm{r}(\alpha)=\mathrm{r}(\beta)$，$\mathrm{S}(\alpha)\le \mathrm{S}(\beta)$，其中 $\mathrm{r}(\alpha)=\dfrac{\mathrm{S}(\alpha)}{\mathrm{HM}(\alpha)}$ 在文献[18]中被称为 $\alpha$ 的 $s/\ell$ 比且模序 $\le$ 的定义延拓到了Laurent多项式环 $\mathcal{K}[x_1,x_2,\cdots,x_n,x_1^{-1},x_2^{-1},\cdots,x_n^{-1}]$ 上。

$\alpha \preceq_{\mathrm{new}} \beta$ 当且仅当 $i\le j, G_i=\alpha, G_j=\beta$，其中 $G=\{G_1,G_2,\cdots,G_k\}$ 且 $i,j\in[1,k]$。可以看出，上述两个序都是线性的，进而是重写序 $\preceq$ 的实例。本文将在后面的章节中讨论两个序的优劣。

令 $\alpha,\beta\in \mathrm{NS}$，$\Gamma_{\alpha\beta}$ 记为最小公倍数 $\mathrm{lcm}(\mathrm{HM}(\alpha),\mathrm{HM}(\beta))$。令 $m_\alpha=\dfrac{\Gamma_{\alpha\beta}}{\mathrm{HM}(\alpha)}$ 且 $m_\beta=\dfrac{\Gamma_{\alpha\beta}}{\mathrm{HM}(\beta)}$。若 $\mathrm{r}(\alpha)>\mathrm{r}(\beta)$，$m_\alpha\alpha$ 在文献[13]中被称为 $\alpha$ 和 $\beta$ 的J-对。$(\alpha,\beta)$ 叫做c-对，$\deg(\Gamma_{\alpha\beta})$ 叫做c-对 $(\alpha,\beta)$ 的(全)次数。$\mathrm{r}(\alpha)>\mathrm{r}(\beta)$ 时，$\deg_s(m_\alpha\alpha)$ 叫做c-对的s-次数。

## 4　矩阵型 Gröbner 基算法

首先，本文引入文献[18]中关于重写基的一些术语。$G$ 是标签 $\boldsymbol{s}$ 的重写基，如果 $\boldsymbol{s}$ 的重写子 $t\alpha\in\mathcal{M}\times G$ 不是 S-可约化的。对于所有小于 $\boldsymbol{s}$ 的标签 $\boldsymbol{s}'$，如果 $G$ 是标签 $\boldsymbol{s}'$ 的重写基，那么 $G$ 被称为直到 $\boldsymbol{s}$ 的重写基(记为 $G^{\mathrm{re}}_{<\boldsymbol{s}}$)。$\ell$-多项式集 $L_{<\boldsymbol{s}}$ 的 S-Gröbner 基也可以记为 $G^{\mathrm{sig}}_{<\boldsymbol{s}}$。记号 $G^{\mathrm{re}}_{\le\boldsymbol{s}}$ 和 $G^{\mathrm{sig}}_{\le\boldsymbol{s}}$ 有类似的定义。

**引理 2**　令 $G$ 为 $G^{\mathrm{sig}}_{<\boldsymbol{s}}$，如果 $\alpha$ 是首不可约 $\ell$-多项式且 $\mathrm{S}(\alpha)<\boldsymbol{s}$，则 $G$ 中有另一 $\ell$-多项式与 $\alpha$ 等价。

证明略。

**推论 2**　如果 $G$ 是 $G^{\mathrm{re}}_{<\boldsymbol{s}}$，则 $G$ 也是 $G^{\mathrm{sig}}_{<\boldsymbol{s}}$。

为了与文献[14]的 M-GVW 算法作比较，这里假设 $\boldsymbol{e}_1>\boldsymbol{e}_2\cdots>\boldsymbol{e}_d$。与文献[22]一样，本文可以推导出 $\boldsymbol{e}_1,\boldsymbol{e}_2,\cdots,\boldsymbol{e}_d$ 为首不可约标签。

给定一组多项式，矩阵型 GVW 算法求出其理想的 Gröbner 基，其中，$\mathrm{sort}(F,\le)$ (相应地，$\min(F,\le)$) 表示按序“$\le$”排列 (相应地，选取) 集合 $F$ 中的元素。顾名思义，$\mathrm{cpair}(\alpha,\beta)$ 为 $\alpha$ 和 $\beta$ 组成的 c-对。$\mathrm{spoly}(\alpha,\beta)$ 表示 $\alpha$ 和 $\beta$ 的 s-多项式 $m_\alpha\alpha-m_\beta\beta$，$\mathrm{concat}(A,B)$ 的意思是把集合 $B$ 中的元素排到集合 $A$ 的后面。子算法 S-REDUCE 利用 $G$ 来反复 s-约化 $\ell$-多项式组 $V$，记录下具有新的首项的不可约 $\ell$-多项式。对于 $\ell$-多项式组 $V$ 的所有单项式，子算法 SYMBOLIC_PREPROCESS 的目的是寻找满足准则的 c-约化子。如果将这些约化子的系数分别写进矩阵的各行，本文就构造出了 Macaulay 矩阵。

这里不给出矩阵型 GVW 算法，因其伪代码与文献[14]基本相同，本节将着重介绍对其子算法的改进及优化。

与矩阵型算法相对应的是 Buchberger 型算法，即算法每次只选择一个 c-对。对于计算 S-Gröbner 基的算法，文献[13]的实验得出两个高效的模序，记为 $\le_{\mathrm{POT}}$ (位置先于项)和 $\le_{\mathrm{SR}}$ (Schreyer 见文献[23])，它们的定义如下：

$m\boldsymbol{e}_i\le_{\mathrm{POT}} t\boldsymbol{e}_j$，如果 $i<j$，或者 $i=j$，$m\le t$，其中 $m,t\in\mathcal{M}$。

$m\boldsymbol{e}_i\le_{\mathrm{SR}} t\boldsymbol{e}_j$，如果 $\mathrm{HM}(mf_i)<\mathrm{HM}(tf_j)$，或者 $\mathrm{HM}(mf_i)=\mathrm{HM}(tf_j)$，$i<j$。

仅按照 s-次数比较 c-对可以记为 $\trianglelefteq_{\mathrm{SD}}$ (与 $\le_{\mathrm{SR}}$ 相比，$\trianglelefteq_{\mathrm{SD}}$ 只是一个偏序)，于是算法选取 s-次数最小的 c-对，其 s-次数为 $D$。

矩阵型 GVW 算法的正确终止性证明可利用推论 2，对标签 $\boldsymbol{s}$ 进行数学归纳得到，细节可以参见文献[13, 14]，这里不再赘述。本节主要讨论算法的优化设计，部分方法借鉴了 Fayssal Martani 对矩阵 F4 算法的优化实现。

### 4.1 延迟求模

约化操作是 Gröbner 基算法中开销最大的部分，当基域较小时，每次约化后多项式系数都要作求模操作。一个自然的想法就是延迟求模运算，即 S-约化 $\overline{V}$ 得到 S-不可约多项式 $\widetilde{V}$ 之后再对各多项式求模。这一技巧能加速计算 Gröbner 基，特别是当基域是 $\mathbb{F}_2$ 的时候。算法 S-REDUCE 用 $H$ 来记录 s-约化为零的那些合冲组成的子模。

### 4.2 高效约化准则

注意到，在选取 $t\beta$ 的时候，通常的做法是确保 $t\beta$ 的标签为非合冲的，并且使 $t\beta$ 不能被重写。实际上，如果算法检查每个 S-约化子是否满足这两个准则，那么算法的效率是相当低下的。所以本文给出了一个等价但更高效的准则：$\ell$-多项式 $\beta$ 的一个 S-约化子 $t\beta$ 被称为最小乘性 S-约化子，如果

(1) $\mathrm{S}(t\beta)$ 是所有 S-约化子中最小的标签；

(2)若有多个 S-约化子满足条件 1，选取 $\mathrm{S}(\beta)$ 最

大的一个。

**事实 1** 令 $\alpha \in L^*$，$G$ 为 $G^{\mathrm{re}}_{<\mathrm{S}(\alpha)}$。对于矩阵型 GVW 算法，$\alpha$ 的 S-约化子 $t\beta$ 通过合冲和重写准则当且仅当其是最小乘性 S-约化子。

鉴于篇幅，本文给出其证明思路：证明其逆否命题的充分和必要性。

### 4.3 并行构造 Macaulay 矩阵

与文献[14]相同的是，子算法 SYMBOLIC_PREPROCESS 函数不指定 $\boldsymbol{s}$ 的选取先后顺序。这样的好处是可以对该函数进行并行化处理。程序使用了 Inter TBB (Thread Building Blocks)库来实现这一操作。具体来说，在构造 Macaulay 矩阵的时候(即把多项式集合 $\overline{V}$ 写成 $|\overline{V}|\times|T|$ 的矩阵，一行代表 $\overline{V}$ 中的一个多项式，其中列元素记录了该多项式关于 $T$ 的系数)，本文用多个线程将 $\overline{V}$ 中不同多项式写成矩阵的行向量，如线程 1 当前处理的单项式在 $T \setminus \mathrm{HM}(\overline{V})$ 中，则线程 1 得到互斥锁，处理完一个单项式后再释放互斥锁。鉴于篇幅，本文省略算法的具体流程。

## 5 实验数据

本文代码是基于 Mathicgb 库[24]的 C++实现。硬件平台为 Intel Core i3 2.40 GHz，运行环境为 64-bit Ubuntu 14.04 操作系统。基域为 $\mathbb{F}_{32003}$，单项式序为 $\leq_{\mathrm{DRL}}$。为了检验线性代数方法对于 Gröbner 基算法是否有加速作用，本文对矩阵型和 Buchberger 型 GVW 算法进行实验比较。表 1~表 4 中，矩阵型 GVW 算法使用了 $\leq_{\mathrm{SR}}$ 模序，$\trianglelefteq_{\mathrm{SD}}$ (按 s-次数排列 c-对)，重写序为 $\preceq_r$。实验显示，线性代数方法对 Cyclic 系列和 Katsura 系列多项式系统能加速 2~6 倍，但该技术不具有普适性。例如 GVW 算法能在不到 2 s 的时间内求出 Jason210 多项式系统的 Gröbner 基，而矩阵型 GVW 算法在 45 min 内都不能算出结果。

表 2 可以看出矩阵版本需要消耗更多的内存，这是由于把多项式集合 $\overline{V}$ 存储成 Macaulay 矩阵的开销很大，尽管本文已经使用了稀疏矩阵作为其存储结构。

**表1 运行时间(s)**

| 例子 | 矩阵型GVW | GVW |
|---|---|---|
| Jason210 | $>2700$ | 1.1 |
| Cyclic7 | 0.3 | 1.1 |
| Cyclic8 | 27.8 | 67.1 |
| Katsura11 | 3.8 | 12.8 |
| Katsura12 | 24.4 | 105.0 |
| Katsura13 | 198.0 | 921.7 |
| Eco10 | 690.9 | 35.5 |
| Joswig | $>2700$ | 225.6 |
| Mayr42 | $>2700$ | 205.5 |

**表 2 内存占用情况 (MB)**

| 例子 | 矩阵型 GVW | GVW |
|---|---|---|
| Jason210 | | 83.4 |
| Cyclic7 | 66.7 | 24.0 |
| Cyclic8 | 1022.2 | 342.4 |
| Katsura11 | 298.4 | 86.2 |
| Katsura12 | 1331.2 | 344.8 |
| Katsura13 | 5836.8 | 1433.6 |
| Eco10 | 270.4 | 251.1 |
| Joswig | | 461.0 |
| Mayr42 | | 228.7 |

除了上节伪代码所描述的矩阵型算法外，本文还实现了基于其它策略或准则的矩阵型算法。例如，表 3 第 3 列是使用 F5 算法的重写准则：选取 $G$ 中最新的 S-约化子。显然，在实现上 $\preceq_{\mathrm{new}}$ 比 $\preceq_r$ (第 2 列)要稍快。实验结果显示，对于 Cyclic 系列和 Eco 系列方程组，$\preceq_{\mathrm{new}}$ 比 $\preceq_r$ 要差的多。对于 Katsura 系列多项式系统，$\preceq_{\mathrm{new}}$ 只需要在 $\preceq_r$ 下一半的运行时间。这一特殊情况是由于算法关于两个重写序所算出的 S-Gröbner 基是相同，并且由表 4 可知 $\preceq_r$ 需要更多的约化操作。

**表3 矩阵型算法运行时间(s)**

| 例子 | 矩阵型GVW | $\preceq_{\mathrm{new}}$ | $\trianglelefteq_{\mathrm{TD}}$ | $\leq_{\mathrm{POT}}$ | M-GVW |
|---|---|---|---|---|---|
| Cyclic7 | 0.3 | $>2700$ | 0.9 | 0.5 | 0.8 |
| Cyclic8 | 27.8 | 内存不足 | $>2700$ | 247.9 | 345.5 |
| Katsura11 | 3.8 | 2.4 | 3.7 | 2535.0 | 2671.0 |
| Katsura12 | 24.4 | 13.3 | 24.4 | 内存不足 | 内存不足 |
| Katsura13 | 198.0 | 93.7 | 203.4 | 内存不足 | 内存不足 |
| Eco10 | 690.9 | $>2700$ | $>2700$ | 2698.5 | $>2700$ |

表 3 第 4 列表示算法矩阵型 GVW 按照全次数从小到大的顺序来选取 c-对。文献[3]表示，$\trianglelefteq_{\mathrm{TD}}$ 对于 F4 算法比 $\trianglelefteq_{\mathrm{SD}}$ 要高效。然而，对于基于标签的 Gröbner 基算法，本文的实验结果说明了这一论断是不成立的。

表 3 第 5 列是算法用了模序 $\leq_{\mathrm{POT}}$ 而不是 $\leq_{\mathrm{SR}}$ 的结果。需要注意的是，在排列 c-对时算法仍然使用 $\trianglelefteq_{\mathrm{SD}}$，这样一来，每一个 Macaulay 矩阵就能包含尽可能多的多项式。表格第 6 列是文献[14]中的矩阵型 M-GVW 算法，其模序为 $\leq_{\mathrm{POT}}$，按 $\trianglelefteq_{\mathrm{TD}}$ 选取 c-对。M-GVW 使用了一个新的策略：如果算法计算出一个突变，则将其 c-约化后赋以新的标签。也就是说，算法将其看成一个新的输入多项式。这样，新的 $\ell$-多项式在已算出的 $G$ 中是关于模序最小的，算法就不会因为其它准则来丢掉与突变相关的 c-对。注意，M-GVW 只对次数小于某一常数的突变进行处理，而在文献[14]中没有给出这一常数的具体值。所以本文在实现矩阵型 M-GVW 的时候只对算法找到的第 1 个突变重赋标签。这样做的目的是确保 M-GVW 不会像文献[14]所说的那样降低性能，然而运行结果出乎意料：M-GVW 计算表 3 的多项式系统要比 $\leq_{\mathrm{POT}}$ 还要差些。从表 3 可以得出，无论是 $\leq_{\mathrm{POT}}$ 还是矩阵型 M-GVW，它们在计算 Gröbner 基的效果上都比矩阵型 GVW 要差。

综上所述，本文所实例化的矩阵型 GVW 算法是权衡了各项准则及策略的实现，具有相当的实用性。找到一个对所有多项式系统都行之有效的算法是相当困难的，即使是 F4 和 F5 算法也做不到。因此，怎样设计更好更快的 Gröbner 基算法的研究是值得继续研究的问题。

**表4 约化总步数**

| 例子 | 矩阵型GVW | $\preceq_{\mathrm{new}}$ |
|---|---|---|
| Katsura11 | 1489738 | 1281310 |
| Katsura12 | 6278351 | 4981658 |
| Katsura13 | 32435088 | 23170749 |

## 参 考 文 献


[1] Buchberger B. Ein algorithmus zum auffinden der basiselemente des restklassenrings nach einem nulldimensionalen polynomideal[D]. [Ph.D. dissertation], Universität Innsbruck, Austria, 1965.

[2] Lazard D. Gröbner-bases, Gaussian elimination and resolution of systems of algebraic equations[C]. Proceedings of the European Computer Algebra Conference on Computer Algebra, London, UK, 1983: 146–156.

[3] Faugère J C. A new efficient algorithm for computing Gröbner bases (F4)[J]. *Journal of Pure and Applied Algebra*, 1999, 139(1-3): 61–88.

[4] Faugère J C. A new efficient algorithm for computing Gröbner bases without reduction to zero (F5)[C]. Proceedings of the 27th International Symposium on Symbolic and Algebraic Computation, New York, USA, 2002: 75–83.

[5] Faugère J C and Joux A. Algebraic cryptanalysis of hidden field equation (HFE) cryptosystems using Gröbner bases[C]. Proceedings of the Advances in Cryptology-CRYPTO 2003, Springer Berlin Heidelberg, Santa Barbara, USA, 2003, 2729: 44–60.

[6] Bardet M. Étude des systèmes algébriques surdéterminés. applications aux codes correcteurs et à la cryptographie[D]. [Ph.D. dissertation], Université Pierre et Marie Curie-Paris VI, 2004.

[7] Faugère J C and Rahmany S. Solving systems of polynomial equations with symmetries using SAGBI-Gröbner bases[C]. Proceedings of the 34th International Symposium on Symbolic and Algebraic Computation, New York, USA, 2009: 151–158.

[8] Albrecht M and Perry J. F4/5[OL]. http:// adsabs. harvard. edu/abs/2010arXiv1006.4933A. 2010.

[9] Faugère J C, Safey El Din M, and Verron T. On the complexity of computing Gröbner bases for quasi-homogeneous systems[C]. Proceedings of the 38th International Symposium on Symbolic and Algebraic Computation, New York, USA, 2013: 189–196.

[10] Bardet M, Faugère J C, and Salvy B. On the complexity of the F5 Gröbner basis algorithm[OL]. http://arxiv.org/abs/1312.1655. 2013.

[11] Arri A and Perry J. The F5 criterion revised[J]. *Journal of Symbolic Computation*, 2011, 46(9): 1017–1029.

[12] Gao Shu-hong, Guan Yin-hua, and Volny F IV. A new incremental algorithm for computing Groebner bases[C]. Proceedings of the 35th International Symposium on Symbolic and Algebraic Computation, New York, USA, 2010: 13–19.

[13] Gao Shu-hong, Volny F, and Wang Ming-sheng. A new algorithm for computing Gröbner bases[OL]. http://www.math.clemson.edu/~sgao/papers/gvw_R130704.pdf. 2010.

[14] Sun Yao, Lin Dong-dai, and Wang Ding-kang. An improvement over the GVW algorithm for inhomogeneous polynomial systems[OL]. http://arxiv.org/abs/1404.1428. 2014.

[15] Huang Lei. A new conception for computing Gröbner basis and its applications[OL]. http://arxiv.org/abs/1012.5425. 2010.

[16] Pan Sen-shan, Hu Yu-pu, and Wang Bao-cang. The termination of the F5 algorithm revisited[C]. Proceedings of the 38th International Symposium on Symbolic and Algebraic Computation, New York, USA, 2013: 291–298.

[17] Eder C. An analysis of inhomogeneous signature-based Gröbner basis computations[J]. *Journal of Symbolic Computation*, 2013, 59(0): 21–35.

[18] Eder C and Roune B H. Signature rewriting in Gröbner basis computation[C]. Proceedings of the 38th International Symposium on Symbolic and Algebraic Computation, New York, USA, 2013: 331–338.

[19] Ding Jin-tai, Cabarcas D, Schmidt D, *et al*.. Mutant Gröbner basis algorithm[C]. Proceedings of the 1st International Conference on Symbolic Computation and Cryptography, Beijing, China, 2008: 23–32.

[20] Sun Yao and Wang Ding-kang. A generalized criterion for signature related Gröbner basis algorithms[C]. Proceedings of the 36th International Symposium on Symbolic and Algebraic Computation, New York, USA, 2011: 337–344.

[21] Eder C and Perry J. Signature-based algorithms to compute Gröbner bases[C]. Proceedings of the 36th International Symposium on Symbolic and Algebraic Computation, New York, USA, 2011: 99–106.

[22] Volny F. New algorithms for computing Gröbner bases[D]. [Ph.D. dissertation], Clemson University, 2011.

[23] Greuel G M and Pfister G. A Singular Introduction to Commutative Algebra[M]. New York: Springer Berlin Heidelberg, 2008: 161–162.

[24] Roune B H and Stillman M. Practical Gröbner basis computation[C]. Proceedings of the 37th International Symposium on Symbolic and Algebraic Computation, New York, USA, 2012: 203–210.

潘森杉： 男，1986 年生，博士生，研究方向为多变量公钥密码、Gröbner 基.

胡予濮： 男，1955 年生，博士，博士生导师，教授，研究方向为格公钥密码、流密码等.

王保仓： 男，1979 年生，博士，硕士生导师，副教授，研究方向为格公钥密码、多变量密码等.