UNIVERSITÀ DEGLI STUDI DI PISA

FACOLTÀ DI SCIENZE MATEMATICHE, FISICHE E NATURALI

CORSO DI LAUREA IN MATEMATICA

TESI DI LAUREA
27 Settembre 2007

L'algoritmo F5 di Faugère: una dimostrazione alternativa ed alcune generalizzazioni

Candidato
Alberto Arri

Relatore
Prof. Massimo Caboara
Università di Pisa

Controrelatore
Prof. Carlo Traverso
Università di Pisa

ANNO ACCADEMICO 2006/2007

Introduzione

Lo scopo di questa tesi è quello di fornire una trattazione ed un'implementazione completa dell'algoritmo "F5" proposto da J.C. Faugère. F5 è un algoritmo per il calcolo delle basi Gröbner, uno strumento chiave nello studio effettivo di ideali e moduli.

Attualmente le basi di Gröbner sono uno strumento fondamentale nella "computer algebra".

In particolare, nel campo dell'algebra commutativa computazionale, la teoria delle basi di Gröbner permette di rispondere ad alcune domande fondamentali: stabilire se un polinomio dato appartiene ad un ideale o no (ideal membership), stabilire se due ideali dati per generatori sono uguali o no, eliminare un'indeterminata da un ideale, risolvere equazioni polinomiali, trasformare le equazioni parametriche di una varietà in equazioni cartesiane e viceversa. Esse sono anche uno strumento per effettuare operazioni sugli ideali, quali calcolo di somma, prodotto, intersezione, radicale, saturato di ideali.

Sin dal momento della loro introduzione, nel 1965, l'algoritmo usuale per il calcolo delle basi di Gröbner è il celebre algoritmo di Buchberger [Buc65] che è stato oggetto di studi e continue migliorie, per citare le più famose, i criteri di Gebauer e Möller [GM88].

Il monopolio dell'algoritmo di Buchberger è stato rotto solo in tempi relativamente recenti con l'introduzione degli algoritmi "F4" ed "F5" da parte di J.C. Faugère in [Fau99, Fau02]. Di F4 esistono implementazioni nei sistemi di calcolo *magma* e *maple*, ed è comunemente riconosciuto come molto efficiente in pratica, tuttavia di nessuna di queste implementazioni è stato reso pubblico il sorgente. F5 è successivo a F4 e, al momento, non ci sono implementazioni disponibili. L'idea principale di F5 è di tenere parzialmente traccia, per qualsiasi polinomio f prodotto durante l'esecuzione, della relazione che lega f ai generatori dell'ideale dati in partenza. Nello specifico, se si sta cercando la base di Gröbner di un ideale omogeneo $I = (f_1, \ldots, f_m)$, e $f = \sum_{i=1}^{k} f_i g_i$, con $g_k \neq 0$, allora ad f si associa il termine di modulo $\mathrm{LT}(g_k)e_k$. Questa informazione aggiuntiva permette di generare esplicitamente basi degli spazi vettoriali $(I)_d$, costituiti dagli elementi di I di grado d assegnato.

Faugère descrive "F5" solo nel caso in cui questo venga applicato ad una successione regolare di polinomi e ne dimostra parzialmente la correttezza e la terminazione solo sotto questa ipotesi.

In questa tesi verrà descritto il funzionamento della versione cosiddetta matriciale di "F5" per successioni regolari, introdotta da M. Bardet nella sua tesi di dottorato [Bar06], in cui tale algoritmo viene solo presentato, senza alcuna giustificazione della sua correttezza. Daremo una dimostrazione originale e completa della correttezza e della terminazione per tale algoritmo. Le idee usate in questa dimostrazione permetteranno, poi, di generalizzare i risultati e di rimuovere l'ipotesi di regolarità. Questi risultati verranno usati per presentare alcune varianti e generalizzazioni dell'algoritmo che risultano più efficienti e flessibili.

Tutti gli algoritmi descritti sono stati effettivamente implementati in un linguaggio di basso livello, il C++, facendo uso della libreria CoCoA5 [CoC]. Questa costituisce al momento l'unica implementazione dell'algoritmo F5 in un linguaggio compilato con sorgente disponibile.

Indice

Capitolo 1

Basi di Gröbner e algoritmo di Buchberger

In questo capitolo daremo le basi della teoria delle basi di Gröbner: definizioni, proprietà più importanti e prime applicazioni pratiche; inoltre introdurremo la funzione di Hilbert che è uno strumento importante da un punto vista teorico.
Considereremo noti i concetti di algebra commutativa di base: gruppi, anelli, campi, moduli.

1.1 Term ordering

In questa sezione tratteremo il problema di mettere un ordine sugli addendi di un polinomio multivariato, in modo da poter parlare, per un polinomio non nullo, di termine di testa. Nel caso univariato in genere ci sono solo due possibili convenzioni "naturali": ordinare i monomi in ordine crescente o in ordine decrescente: $f = x^7 + 12x^6 - 2x + 1$ oppure $f = 1 - 2x + 12x^6 + x^7$. Se invece si sta lavorando con polinomi in più di un'indeterminata ci sono molte più possibilità.

Definizione 1.1.1. Sia $n \geq 1$, un polinomio $f \in R[x_1, \ldots, x_n]$ della forma $f = x_1^{\alpha_1} \ldots x_n^{\alpha_n}$, con $\alpha_1, \ldots, \alpha_n \in \mathbb{N}$ si dice *termine*. L'insieme di tutti i termini di $R[x_1, \ldots, x_n]$ è indicato con $\mathbb{T}^n$.
Dato un termine $t = x_1^{\alpha_1} \ldots x_n^{\alpha_n} \in \mathbb{T}^n$, il numero naturale $\deg(t) = \sum_{i=1}^n \alpha_i$ è detto il *grado* di t.
Se $r \geq 1$ e $M = (R[x_1, \ldots, x_n])^r$ è il $R[x_1, \ldots, x_n]$-modulo libero e finitamente generato con base canonica $\{e_1, \ldots, e_n\}$, allora un *termine* di M è un elemento della forma te_i, dove $t \in \mathbb{T}^n$ e $1 \leq i \leq r$. L'insieme dei termini di M è indicato con $\mathbb{T}^n\langle e_1, \ldots, e_n\rangle$.

L'insieme $\mathbb{T}^n$ ha una naturale struttura di monoide commutativo con identità $1 = x_1^0 \ldots x_n^0$, e non dipende dall'anello dei coefficienti R.

Inoltre in $\mathbb{T}^n$ vale la legge di cancellazione: $\forall t, u, v \in T^n : tu = tv$ implica $u = v$.
Come monoide, $\mathbb{T}^n$ è isomorfo a $\mathbb{N}^n$ tramite la mappa $\log$ definita da

$$x_1^{\alpha_1} \dots x_n^{\alpha_n} \mapsto (\alpha_1, \dots, \alpha_n)$$

Definizione 1.1.2. Sia $n \geq 1$, sia $f = \sum_{\alpha \in \mathbb{N}^n} c_\alpha t_\alpha \in R[x_1, \dots x_n]$ un polinomio e sia $m = \sum_{i=1}^{r} \sum_{\alpha \in \mathbb{N}^n} c_{\alpha,i} t_{\alpha,i} \in M = (R[x_1, \dots x_n])^r$.

- Per ogni $\alpha \in \mathbb{N}^n, 1 \leq i \leq r$, l'elemento $\mathrm{Coeff}(t_\alpha e_i, m) = c_{\alpha,i} \in R$ è detto *coefficiente* del termine $t_\alpha e_i$ in m.
- L'insieme $\mathrm{Supp}(m) = \{t_\alpha e_i \in \mathbb{T}^n \langle e_1, \dots, e_r \rangle \mid c_{\alpha,i} \neq 0\}$ è chiamato *supporto* di m.
- Se $f \neq 0$ allora il numero naturale $\deg(f) = \max\{\deg(t_\alpha) \mid t_\alpha \in \mathrm{Supp}(f)\}$ è detto *grado* di f.

Definizione 1.1.3 (Term ordering)**.** Una relazione d'ordine totale σ su $\mathbb{T}^n$ si dice *term ordering*, o *ordinamento*, se valgono:

- $\forall t_1, t_2, t_3 \in \mathbb{T}^n : t_1 \geq_\sigma t_2 \rightarrow t_1 t_3 \geq_\sigma t_2 t_3$
- $\forall t \in \mathbb{T}^n : t \geq_\sigma 1$

Definizione 1.1.4 (Ordine `Lex`)**.** Dati $t_1, t_2 \in \mathbb{T}^n$ scriviamo $t_1 \geq_{\mathtt{Lex}} t_2$ se e solo se la prima componente non nulla di $\log(t_1) - \log(t_2)$ è positiva, oppure se $t_1 = t_2$. Quest'ordine è chiamato *ordinamento lessicografico* ed è denotato da `Lex`.

Definizione 1.1.5 (Ordine `DegLex`)**.** Dati $t_1, t_2 \in \mathbb{T}^n$ scriviamo $t_1 \geq_{\mathtt{DegLex}} t_2$ se vale $\deg(t_1) > \deg(t_2)$, oppure se $\deg(t_1) = \deg(t_2)$ e $t_1 \geq_{\mathtt{Lex}} t_2$. Quest'ordine è chiamato *ordinamento grado-lessicografico* ed è denotato da `DegLex`.

Definizione 1.1.6 (Ordine `DRL`)**.** Dati $t_1, t_2 \in \mathbb{T}^n$ scriviamo $t_1 \geq_{\mathtt{DRL}} t_2$ se vale $\deg(t_1) > \deg(t_2)$, oppure se $\deg(t_1) = \deg(t_2)$ e l'ultima componente non nulla di $\log(t_1) - \log(t_2)$ è negativa, o se $t_1 = t_2$. Quest'ordine è chiamato *ordinamento grado-inverso-lessicografico* ed è denotato da `DRL`.

Definizione 1.1.7 (Term ordering grado compatibile)**.** Un term ordering σ su $\mathbb{T}^n$ è detto *grado compatibile* se $\deg(t_1) > \deg(t_2)$ implica $t_1 \geq_\sigma t_2$.

Ad esempio i term ordering `DegLex` e `DRL` sono grado compatibili.

Definizione 1.1.8 (Ordinamenti di modulo)**.** Una relazione d'ordine totale σ su $\mathbb{T}^n \langle e_1, \dots, e_r \rangle$ è detta *module ordering* o *ordinamento di modulo* se:

- $\forall s_1, s_2 \in \mathbb{T}^n \langle e_1, \dots, e_r \rangle, t \in \mathbb{T}^n : s_1 \geq_\sigma s_2 \rightarrow t s_1 \geq_\sigma t s_2$

- $\forall s \in \mathbb{T}^n\langle e_1, \dots, e_r\rangle, t \in \mathbb{T}^n : ts \geq_\sigma s$.

Definizione 1.1.9 (Ordinamenti `PosTo`e `ToPos`)**.** Sia `To`un ordinamento su $\mathbb{T}^n$.

- Dati $t_1e_i, t_2e_j \in \mathbb{T}^n\langle e_1, \dots, e_r\rangle$, dove $t_1, t_2 \in \mathbb{T}^n$, e $i, j \in \{1, \dots, r\}$, si definisce:

$$t_1e_i \geq_{\texttt{ToPos}} t_2e_j \iff t_1 >_{\texttt{To}} t_2 \text{ oppure } (t_1 = t_2 \wedge i \leq j)$$

  Questo è l'ordinamento di modulo `ToPos`su $\mathbb{T}^n\langle e_1, \dots, e_r\rangle$. Detto altrimenti, l'ordinamento `ToPos`prima confronta i termini con l'ordinamento `To`, in caso di parità, si usa l'ordine sui generatori del modulo.

- Nelle stesse ipotesi di prima:

$$t_1e_i \geq_{\texttt{PosTo}} t_2e_j \iff i < j \text{ oppure } (i = j \wedge t_1 \leq_{\texttt{To}} t_2)$$

  Questo definisce l'ordinamento `PosTo`su $\mathbb{T}^n\langle e_1, \dots, e_r\rangle$.

Teorema 1.1.1. *I term ordering ed i module ordering sono dei* buoni ordini[1] *rispettivamente su* $\mathbb{T}^n$ *e su* $\mathbb{T}^n\langle e_1, \dots, e_r\rangle$.

Avendo messo un ordine σ su $\mathbb{T}^n$ siamo in grado di scrivere in modo "unico" un polinomio multivariato: ordinando con σ i vari termini che ne costituiscono il supporto. In particolare possiamo dare le seguenti:

Definizione 1.1.10 (Termini e coefficienti di testa)**.** Sia $P = R[x_1, \dots x_n]$, siano $m \in P^r$, $m \neq 0$, e σ un ordinamento su P^r.

- $\mathrm{LT}_\sigma(m) = \max\{t \mid t \in \mathrm{Supp}(m)\} \in \mathbb{T}^n\langle e_1, \dots, e_r\rangle$ è detto *termine di testa* di m rispetto a σ.

- $\mathrm{LC}_\sigma(m) = \mathrm{Coeff}(\mathrm{LT}_\sigma(m), m) \in R$, il coefficiente del temine di testa è detto *coefficiente di testa* di m rispetto a σ.

- $\mathrm{LM}_\sigma(m) = \mathrm{LC}_\sigma(m)\,\mathrm{LT}_\sigma(m) \in P^r$ è detto *monomio di testa* di m rispetto a σ.

Definizione 1.1.11 (Modulo e ideale dei termini di testa)**.**
Sia $P = R[x_1, \dots x_n]$ e sia $M \subseteq P^r$ un P-sottomodulo.

- Il modulo $\mathrm{LT}_\sigma(M) = \langle \mathrm{LT}_\sigma(m) \mid m \in M \setminus \{0\}\rangle_P$ è chiamato *modulo dei termini di testa* di M rispetto a σ.

[1] Un buon ordine su un insieme X è un ordine totale tale che ogni sottoinsieme non vuoto di X ha minimo.

- Nel caso $r = 1$, cioè quando $M \subseteq P$ allora l'ideale $\mathrm{LT}_\sigma(M) \subseteq P$ è anche detto *ideale dei termini di testa* di M rispetto a σ.

Se $M = \{0\}$, allora $\mathrm{LT}_\sigma(M) = \{0\}$.

Teorema 1.1.2 (Teorema della base di Macaulay)**.** *Sia $\mathbb{K}$ un campo, sia $P = \mathbb{K}[x_1, \ldots, x_n]$ un anello polinomiale su $\mathbb{K}$, sia $M \subseteq P^r$ un P-sottomodulo, e sia σ un ordinamento di modulo su $\mathbb{T}^n\langle e_1, \ldots, e_r\rangle$. Sia inoltre $B = \mathbb{T}^n\langle e_1, \ldots, e_r\rangle \setminus \mathrm{LT}_\sigma(M)$. Allora le classi residue degli elementi di B formano una base del $\mathbb{K}$-spazio vettoriale P^r/M.*

1.2 L'algoritmo di divisione

Consideriamo cosa succede per classi residue di anelli della forma $\mathbb{K}[x]$. In questo caso ogni ideale è principale. Sia $I \subseteq \mathbb{K}[x]$ un ideale non vuoto e $f = a_d x^d + a_{d-1}x^{d-1} + \cdots + a_0$, con $a_d \neq 0$, un generatore di I. Usando l'algoritmo di divisione univariata tra polinomi, dato un qualsiasi $g \in \mathbb{K}[x]$, siamo in grado di trovare una rappresentazione $g = qf + p$, dove p o è zero o ha grado minore di d. Questo implica che gli elementi dell'anello quoziente $\mathbb{K}[x]/(f)$ possono essere rappresentati in modo *unico* come combinazioni lineari delle classi residue $1, \bar{x}, \ldots, \bar{x}^{d-1}$.
Questo è un caso particolare dell'appena citato teorema di Macaulay, ma il fatto che l'algoritmo di divisione permetta di rappresentare le classi residue è una particolarità degli anelli con un'unica indeterminata. Vedremo come questo si generalizza al caso multivariato e quali proprietà si manterranno.
Per brevità di notazione poniamo $P = \mathbb{K}[x_1, \ldots, x_n]$; cominciamo con il descrivere *l'algoritmo di divisione* per un generico elemento m del modulo libero P^r, dotato di ordinamento σ, rispetto ad una successione di s elementi $g_1, g_2, \ldots, g_s$ di P^r.

Algoritmo 1.2.1 (Algoritmo di divisione)**.**

Input: $m, g_1, \ldots, g_s \in P^r \setminus \{0\}$

Output: $q_1, \ldots, q_s \in P$, $p \in P^r$

1. For $i := 1$ to s do $q_i := 0$
2. $p := 0$
3. $v := m$
4. While $v \neq 0$
   (a) $i := \mathrm{FindReducer}(v, (g_1, \ldots, g_s))$
   (b) While $i \neq 0$

i. $q_i := q_1 + \frac{\mathrm{LM}_\sigma}{\mathrm{LM}_\sigma(g_i)}$

ii. $v := v - \frac{\mathrm{LM}\,\sigma(v)}{\mathrm{LM}_\sigma(g_i)} g_i$

(c) $p := p + \mathrm{LM}_\sigma(v)$

(d) $v := v - \mathrm{LM}_\sigma(v)$

5. Return $(q_1, \ldots, q_s)$, p

Dove la procedura $\mathrm{FindReducer}(v, (g_1, \ldots, g_s))$ restituisce il più piccolo $i \in \{1, \ldots, s\}$ tale che $\mathrm{LT}_\sigma(g_i) \mid \mathrm{LT}_\sigma(v)$ se dei tali i esistono e 0 altrimenti.

Teorema 1.2.1 (Proprietà dell'algoritmo di divisione)**.**
Se $p \in P^r$ e $(q_1, \ldots, q_s) \in P^s$ sono gli elementi restituiti dall'algoritmo di divisione applicato a m e $(g_1, \ldots, g_s)$ allora si ha:

$$m = \sum_{i=1}^{s} q_i g_i + p$$

Inoltre valgono le seguenti:

1. *Nessun elemento di* $\mathrm{Supp}(p)$ *è contenuto in* $\langle \mathrm{LT}_\sigma(g_1), \ldots, \mathrm{LT}_\sigma(g_s) \rangle_P$.
2. *Se* $q_i \neq 0$*, per qualche* $i \in \{1, \ldots, s\}$*, allora* $\mathrm{LT}_\sigma(q_i g_i) \leq \mathrm{LT}_\sigma(m)$.
3. *Per tutti gli* $i \in \{1, \ldots, s\}$ *e tutti i termini* t *nel supporto di* q_i *abbiamo* $t\,\mathrm{LT}_\sigma(g_i) \notin \langle \mathrm{LT}_\sigma(g_1), \ldots, \mathrm{LT}_\sigma(g_{i-1}) \rangle_P$.

Definizione 1.2.1 (Resto normale)**.** Se $p \in P^r$ e $\mathcal{G} = (q_1, \ldots, q_s) \in P^s$ sono gli elementi restituiti dall'algoritmo di divisione applicato ad m e $(g_1, \ldots, g_s)$, allora p è detto *resto normale* di m rispetto a $\mathcal{G}$, e si indica con $p = \mathrm{NR}_{\sigma,\mathcal{G}}(m)$[2].

Un importante osservazione da fare è che $\mathrm{NR}_\mathcal{G}(m)$ dipende dall'ordine degli elementi di $\mathcal{G}$, e, inoltre, non è in generale vero che se m appartiene all'ideale generato da $\mathcal{G}$, allora $\mathrm{NR}_\mathcal{G}(m) = 0$. Quindi questo algoritmo da solo non basta a risolvere il problema dall'appartenenza ad ideale o a sottomodulo.

1.3 Basi di Gröbner

Per brevità di notazione poniamo $P = \mathbb{K}[x_1, \ldots, x_n]$.

[2]Quando sarà chiaro dal contesto ometteremo il pedice σ.

Definizione 1.3.1 (Sizigie)**.** Siano $\mathcal{F} = (f_1, \ldots, f_m) \in P^m$, e $I = (\mathcal{F})$ l'ideale generato da $\mathcal{F}$, consideriamo

$$\begin{array}{rccc} \phi: & P^m & \to & I \\ & (g_1, \ldots, g_m) & \mapsto & \sum_{i=1}^{m} g_i f_i \end{array}$$

Si chiama *modulo delle sizigie* di $\mathcal{F}$ il P-sottomodulo di P^m:

$$\mathrm{Syz}(\mathcal{F}) = \ker(\phi)$$

Un elemento g di $\mathrm{Syz}(\mathcal{F})$ sarà chiamato *sizigia*. Una sizigia $g = (g_1, \ldots, g_m)$ tale che se $g_i \neq 0$ allora g_i è omogeneo e $\deg(g_i) + \deg(f_i)$ è costante per ogni i è detta *sizigia omogenea*.

Osservazione 1.3.1. Un'osservazione importante: le sizigie sono definite per la successione di polinomi e non per l'ideale che essi generano, come il seguente esempio illustra:

Esempio 1. Consideriamo $I = (x) = (x, x^2) \subseteq \mathbb{Q}[x]$: si ha $\mathrm{Syz}((x)) = \emptyset$, mentre $(x, -1) \in \mathrm{Syz}((x, x^2)) \neq \emptyset$.

Definizione 1.3.2 (Sollevamento)**.** Un elemento di P^m è chiamato *sollevamento* di un elemento $\overline{m} \in P^m$ se vale $\mathrm{LT}(m) = \overline{m}$.

Definizione 1.3.3. Siano $g_1, \ldots, g_s \in P^r \setminus \{0\}$, e sia $G = \{g_1, \ldots, g_s\}$.

- Siano $m_1, m_2 \in P^r$, e si supponga esistano $c \in \mathbb{K}$, un termine $t \in \mathbb{T}^n$ ed un naturale $1 \leq i \leq s$ tali che $m_2 = m_1 - ctg_i$ e che $t\,\mathrm{LT}_\sigma(g_i) \notin \mathrm{Supp}(m_2)$. Allora si dice che m_1 *si riduce ad* m_2 *in un passo* usando *la regola di riscrittura* definita da g_i, e si scrive $m_1 \xrightarrow{g_i} m_2$. Il passaggio da m_1 ad m_2 è anche detto *passo di riduzione*.

- La chiusura transitiva delle relazioni $\xrightarrow{g_1}, \ldots, \xrightarrow{g_2}$ è chiamata *relazione di riscrittura* definita da G e si denota con $\xrightarrow{G}$.

- Un elemento $m_1 \in P^r$ tale che non esiste nessun $i \in \{1, \ldots, s\}$ e nessun $m_2 \in P^r \setminus \{m_1\}$ tale che $m_1 \xrightarrow{g_i} m_2$ è chiamato *irriducibile* rispetto a G.

- La relazione di equivalenza indotta da $\xrightarrow{G}$ si indicherà con $\xleftrightarrow{G}$.

Teorema 1.3.1 (Caratterizzazione delle basi di Gröbner)**.** *Dato un insieme di elementi $G = \{g_1, \ldots, g_s\} \subseteq P^r \setminus \{0\}$ che genera un sottomodulo $M = \langle g_1, \ldots, g_s \rangle_P \subseteq P^r$, sia $\xrightarrow{G}$ la regola di riscrittura definita da G e sia $\mathcal{G} = (G)$. Allora le seguenti condizioni sono equivalenti:*

A_1 *Per ogni elemento* $m \in M \setminus \{0\}$*, esistono* $f_1, \ldots, f_s \in P$ *tali che* $m = \sum_{i=1}^{s} f_i g_i$ *e* $\mathrm{LT}_\sigma(m) \geq_\sigma \mathrm{LT}_\sigma(f_i g_i)$ *(se* $f_i g_i \neq 0$*).*

A_2 *Per ogni elemento* $m \in M \setminus \{0\}$ *esistono* $f_1, \ldots, f_s \in P$ *tali che* $m = \sum_{i=1}^{s} f_i g_i$ *e* $\mathrm{LT}_\sigma(m) = \max_\sigma\{\mathrm{LT}_\sigma(f_i g_i) \mid f_i g_i \neq 0\}$*.*

B *L'insieme* $\{\mathrm{LT}_\sigma(g_1), \ldots, \mathrm{LT}_\sigma(g_s)\}$ *genera* $\mathrm{LT}_\sigma(M)$*.*

C_1 *Dato* $m \in P^r$*, si ha* $m \xrightarrow{G} 0$ *se e solo se* $m \in M$*.*

C_2 *Se* $m \in M$ *è irriducibile rispetto a* $\xrightarrow{G}$*, allora* $m = 0$*.*

C_3 *Dato un elemento* $m_1 \in P^r$*, c'è un unico elemento* $m_2 \in P^r$*, tale che* $m_1 \xrightarrow{G} m_2$ *ed* m_2 *è irriducibile rispetto* $\xrightarrow{G}$*.*

C_4 *Se* $m_1, m_2, m_3 \in P^r$ *soddisfano* $m_1 \xrightarrow{G} m_2$ *e* $m_1 \xrightarrow{G} m_3$*, allora esiste un elemento* $m_4 \in P^r$ *tale che* $m_2 \xrightarrow{G} m_4$ *e* $m_3 \xrightarrow{G} m_4$*.*

D_1 *Ogni elemento omogeneo di* $\mathrm{Syz}(\mathrm{LM}(G))$ *ha un sollevamento in* $\mathrm{Syz}(\mathcal{G})$*.*

D_2 *Esiste un sistema di generatori omogenei di* $\mathrm{Syz}(\mathrm{LM}(G))$ *costituito interamente da elementi che hanno sollevamento in* $\mathrm{Syz}(G)$*.*

D_3 *Esiste un sistema finito di generatori omogenei di* $\mathrm{Syz}(\mathrm{LM}(G))$ *costituito interamente da elementi che hanno sollevamento in* $\mathrm{Syz}(G)$*.*

Definizione 1.3.4 (Base di Gröbner)**.** Sia $G = \{g_1, \ldots, g_s\} \subseteq P^r \setminus \{0\}$ un insieme di elementi che genera un sottomodulo $M = \langle g_1, \ldots, g_s \rangle_P \subseteq P^r$. Se le condizioni del teorema 1.3.1 sono soddisfatte allora G è chiamata *base di Gröbner* di M rispetto a σ, o *σ-base di Gröbner* .
Nel caso $M = \{0\}$, diciamo che $G = \emptyset$ è una base di Gröbner di M.

Proposizione 1.3.2 (Esistenza delle basi di Gröbner)**.** *Sia M un sottomodulo di P^r.*

- *Dati* $g_1, \ldots, g_s \in M \setminus \{0\}$ *tali che* $\mathrm{LT}_\sigma(M) = \langle \mathrm{LT}_\sigma(g_1), \ldots, \mathrm{LT}_\sigma(g_s) \rangle_P$*, abbiamo* $M = \langle g_1, \ldots, g_s \rangle_P$*, e l'insieme* $G = \{g_1, \ldots, g_s\}$ *è una σ-base di Gröbner di M.*
- *Il modulo M ha una σ-base di Gröbner* $G = \{g_1, \ldots, g_s\} \subseteq M \setminus \{0\}$*.*

Definizione 1.3.5. Un anello, o un modulo, si dice *noetheriano* se ogni catena ascendente di ideali diventa stazionaria.

Proposizione 1.3.3. *Sia R un anello e sia M un R-modulo. Allora le seguenti sono equivalenti:*

- *Ogni sottomodulo M è finitamente generato.*

- *Ogni catena ascendente di sottomoduli di M è stazionaria.*
- *Ogni insieme, non vuoto, di sottomoduli di M ha un elemento massimale rispetto all'inclusione.*

Teorema 1.3.4 (Teorema della base di Hilbert)**.** *Ogni modulo finitamente generato su una $\mathbb{K}$-algebra finitamente generata è* noetheriano*. In particolare $P = \mathbb{K}[x_1, \ldots, x_n]$ è un anello noetheriano.*

1.3.1 Forme normali

La prima applicazione delle basi di Gröbner è quella di rendere effettive le operazioni nei moduli quozienti, cioè della forma P^r/M. Il problema centrale è quello di definire un rappresentante unico delle classi di equivalenza che costituiscono il quoziente. Usando le basi di Gröbner si può facilmente ovviare a questa difficoltà.
Supponiamo di avere una base di Gröbner $G = \{g_1, \ldots, g_s\} \subseteq P^r$ di $M = \langle g_1, \ldots, g_s \rangle_P$ rispetto ad un qualche ordinamento σ. La condizione C_3 della caratterizzazione delle basi di Gröbner dice che, dato $m \in P^r$, esiste un unico m_G che soddisfa $m \xrightarrow{G} m_G$, con m_G irriducibile rispetto a G. m_G pare quindi un buon candidato per essere il rappresentante "speciale" della classe di equivalenza di m modulo M; tuttavia per quanto visto sinora l'elemento m_G pare dipendere dalla scelta della base di Gröbner G.

Proposizione 1.3.5 (Unicità della forma normale)**.** *Nelle ipotesi precedenti, m_G dipende solo dall'ordinamento, e non dalla base di Gröbner G. In particolare m_G può essere caratterizzato come quell'elemento tale che $m - m_G \in M$ e $\mathrm{Supp}(m_G) \cap \mathrm{LT}_\sigma(M) = \emptyset$.*

Definizione 1.3.6 (Forma normale)**.** L'elemento m_G definito sopra è detto *forma normale* di m rispetto a σ, e si denota con la notazione $\mathrm{NF}_{\sigma(m),M}$ [3]. Se per un certo m vale $m = m_G$ diciamo che m è in forma normale rispetto a M.

Proposizione 1.3.6 (Proprietà delle forme normali)**.**

- *Se $m \in P^r$ allora $\mathrm{NR}_\sigma(m)$ coincide con $\mathrm{NF}_\sigma(m)$*
- *Dati $m_1, m_2 \in P^r$ e $\alpha, \beta \in \mathbb{K}$ allora $\mathrm{NF}_\sigma(\alpha m_1 + \beta m_2) = \alpha\,\mathrm{NF}_\sigma(m_1) + \beta\,\mathrm{NF}_\sigma(m_2)$. Cioè NF_σ è un'applicazione $\mathbb{K}$-lineare.*

Corollario 1.3.7. *La forma normale di $m \in P^r$ rispetto ad una base di Gröbner $\mathcal{G}$ può essere calcolata applicando l'algoritmo 1.2.1.*

Proposizione 1.3.8 (Test di appartenenza a sottomoduli)**.** *Dato $m \in P^r$, questo appartiene ad M se e solo se $\mathrm{NF}_M(m) = 0$. Dove si intende la forma normale fatta rispetto ad un qualsiasi ordinamento di P^r.*

[3] In generale, quando sarà ovvio dal contesto, ometteremo i pedici σ e/o M.

1.3.2 Basi ridotte

Come si è visto le basi Gröbner esistono sempre, ma non sono uniche, in questo paragrafo daremo delle condizioni ulteriori per ottenere l'unicità.

Definizione 1.3.7 (Base di Gröbner ridotta)**.** Sia $G = \{g_1, \dots, g_s\} \subseteq P^r \setminus \{0\}$ una base di Gröbner di un sottomodulo M. Diremo che G è una *σ-base di Gröbner ridotta* se:

- g_i è monico $\forall i = 1, \dots, s$.
- L'insieme $\{\mathrm{LT}_\sigma(g_i) \mid i = 1, \dots, s\}$ è un sistema minimale di generatori di $\mathrm{LT}_\sigma(M)$.
- $\mathrm{Supp}(g_i - \mathrm{LT}_\sigma(g_i)) \cap \mathrm{LT}_\sigma(M) = \emptyset$ per $i = 1, \dots, s$.

Teorema 1.3.9 (Esistenza ed unicità delle basi di Gröbner ridotte)**.** *Per ogni sottomodulo $M \subseteq P^r$, esiste un'unica σ-base di Gröbner ridotta.*

1.4 Algoritmo di Buchberger

Nelle sezioni precedenti abbiamo parlato delle forme normali e delle loro prime applicazioni. Tuttavia, affichè queste siano effettive in pratica è necessario trovare un modo di calcolare una base di Gröbner di un sottomodulo o di un ideale dato per generatori.
Questo problema è stato risolto da Bruno Buchberger nella sua tesi di dottorato [Buc65] introducendo il concetto stesso di base di Gröbner e l'algoritmo, che porta il suo nome, per il calcolo delle stesse.

Definizione 1.4.1 (S-polinomio e S-vettore)**.** Siano g_1 e g_2 due elementi di P^r tali che $\mathrm{LM}(g_i) = c_i t_i e_k$, dove $c_i \in \mathbb{K}$, $t_i \in \mathbb{T}^n$ e $1 \leq k \leq r$, in particolare si noti che i due monomi di testa sono nella stessa componente, la k-esima, di P^r. Siano poi $t_{ij} = \frac{\mathrm{lcm}(t_i, t_j)}{t_i} = \frac{t_j}{\gcd(t_i, t_j)} \in \mathbb{T}^n$, con $\{i, j\} = \{1, 2\}$, allora si definisce:

$$S = \mathrm{SPoly}(g_1, g_2) = \frac{1}{c_1} t_{12} g_1 - \frac{1}{c_2} t_{21} g_2 \quad \in \quad M$$

S-vettore di g_1 e g_2. Nel caso $r = 1$ allora $S \in P$, e si chiama *S-Polinomio*.

Proposizione 1.4.1 (Criterio di Buchberger)**.** *Sia $M \subseteq P^r$ un P-sottomodulo generato da $G = \{g_1, \dots, g_s\} \subseteq P^r \setminus \{0\}$. Sia inoltre $S_{i,j}$ l'S-vettore (o S-polinomio), se esiste, tra g_i e g_j, con $i \neq j \in \{1, \dots, s\}$. Allora le seguenti condizioni sono equivalenti:*

- *G è una base di Gröbner di M.*
- *Ogni S-Vettore $S_{i,j}$ verifica $\mathrm{NR}_G(S_{i,j}) = 0$.*

Questa proposizione permette facilmente di dare un algoritmo effettivo per il calcolo di una base di Gröbner: dato un insieme di generatori $\mathcal{F}$ considero l'insieme degli S-polinomi, cioè delle coppie, calcolo la riduzione normale rispetto ad $\mathcal{F}$ di ciascuno di questi, ed aggiungo ad $\mathcal{F}$ quelli che non si sono ridotti a zero. Ripeto finché ad un passo non aggiungo ad $\mathcal{F}$ nuovi polinomi. È ovvio a questo punto che l'insieme $\mathcal{F}$ che ho costruito soddisfa il criterio di Buchberger ed è, pertanto, una base di Gröbner.
Formalizzando si arriva al seguente:

Algoritmo 1.4.1 (Algoritmo di Buchberger)**.**

Input: $\mathcal{G} = (g_1, \ldots, g_m) \in (P^r)^s$.

Output: Una base di Gröbner del sottomodulo $M = (\mathcal{G})$.

1. $\mathbb{B} := \{(g_i, g_j) \mid \mathrm{LT}(g_i) = c_i t_i e_{\gamma_i}, \mathrm{LT}(g_j) = c_j t_j e_{\gamma_j} \text{ con } \gamma_i = \gamma_j\}$
2. While $\mathbb{B} \neq \emptyset$
   (a) Choose $(g_i, g_j) \in \mathbb{B}$
   (b) $\mathbb{B} := \mathbb{B} \setminus \{(g_i, g_j)\}$
   (c) $g := \mathrm{NF}_{\mathcal{G}}(\mathrm{SPoly}(g_i, g_j))$
   (d) If $g \neq 0$ then
      i. $\mathcal{G} := \mathcal{G} \cup \{g\}$
      ii. Update $\mathbb{B}$
3. Return $\mathcal{G}$.

La procedura "Update" aggiunge a $\mathbb{B}$ le coppie che coinvolgono l'elemento g appena aggiunto a $\mathcal{G}$.

Teorema 1.4.2. *L'algoritmo 1.4.1 termina e restituisce una base di Gröbner del sottomodulo M generato da G.*

Dimostrazione. La terminazione è una conseguenza della noetherianità di P^r: ogni volta che un polinomio non si riduce a zero, si aggiunge un generatore all'ideale $\mathrm{LT}(M)$. Per la proposizione 1.4.1, dal momento che, alla fine, $\mathcal{G}$ è un insieme tale che tutti gli S-vettori si riducono a zero, il valore restituito è una base di Gröbner di M. □

Descritto in questo modo l'algoritmo ha ancora alcuni gradi di libertà: in particolare due sono i più importanti: la scelta della coppia da ridurre, cioè in che ordine processare le coppie rimanenti, e quali riduttori usare per fare le riduzioni normali.
Matematicamente queste scelte sono ininfluenti sul risultato finale dell'algoritmo, tuttavia è noto, dall'evidenza sperimentale, che queste scelte sono fattori determinanti per la performance dell'algoritmo; differenti criteri

di scelta possono tranquillamente cambiare di vari ordini di grandezza il tempo di esecuzione e l'occupazione della memoria.

Nel corso degli anni sono state sviluppate varie tecniche, alcune di natura più matematica ed altre di natura più strettamente informatica ed implementativa che sono oggi universalmente accettate, anche se non esiste ancora una teoria completa che sia in grado di prevedere quali scelte siano le migliori ad ogni passo.

Un fattore che rallenta molto l'esecuzione dell'algoritmo è il fatto che, in generale, molti degli S-polinomi si riducono a zero. Attualmente sono noti dei criteri, dovuti allo stesso Buchberger (in [Buc79]) ed, in seguito, a Gebauer e Möller (in [GM88]) che garantiscono che alcune coppie si riducono a zero solo guardando i termini di testa dei polinomi.

Un altro fattore determinante per il costo, sia di tempo che di memoria, dell'algoritmo di Buchberger è dato dall'ordinamento scelto; sebbene anche su questo non ci siano risultati dimostrati, è noto che ordinamenti grado compatibili, quali `DRL`, danno tempi di esecuzione più bassi. Mentre ordinamenti quali `Lex`, che producono basi di Gröbner con proprietà matematiche più interessanti[4] richiedono tempi di esecuzione più lunghi.

Esempio 2 (Influenza dell'ordinamento sui tempi)**.** Riportiamo le misurazioni di alcuni tempi di esecuzione dell'algoritmo di Buchberger e cardinalità della base di Gröbner prodotta.

Ideale	`Lex`		`DRL`	
	tempo	$\|GB\|$	tempo	$\|GB\|$
Ciclico 5	$0.05s$	43	$0.03s$	38
Ciclico 6	$5.9s$	162	$0.4s$	99
Katsura 7	$298s$	677	$0.88s$	74
Denso 4 6 9	$6.00s$	632	$1.80s$	409
Denso 4 7 10	$10.3s$	767	$3.12s$	505

Tutti gli esempi sono a coefficienti in $\mathbb{Z}/(32003)$. Gli ideali ciclici e di Katsura sono tra i benchmark standard per algoritmi che calcolano basi di Gröbner. Denso $a\ b\ c$ è un ideale omogeneo di $\mathbb{Z}/(32003)[x_1, \ldots, x_a]$ generato da b polinomi di grado c con i coefficienti scelti a caso, con distribuzione di probabilità uniforme in $\mathbb{Z}/(32003)$. Questi esperimenti sono stati fatti con l'implementazione dell'algoritmo di Buchberger presente nella libreria CoCoA5 su un Athlon 64 3200+ con 1Gb di ram.

[4] Permettono di eliminare indeterminate (vedere la sezione 1.6.1) e di trovare gli zeri di sistemi zero dimensionali.

1.5 Graduazioni

In questa sezione specializzeremo parte della teoria sinora sviluppata nel caso in cui questa venga applicata ad anelli graduati.

Definizione 1.5.1 (Anello graduato)**.** Sia $(\Gamma, +)$ un monoide.
Un anello R è detto Γ-*graduato* se esiste una famiglia di sottogruppi additivi di R, $\{R_\gamma\}_{\gamma\in\Gamma}$ tali che:

1. $R = \bigoplus_{\gamma\in\Gamma} R_\gamma$.
2. $R_\gamma \cdot R_{\gamma'} \subseteq R_{\gamma+\gamma'}$ per ogni $\gamma, \gamma' \in \Gamma$.

Gli elementi di R_γ sono detti elementi *omogenei di grado* γ. Dato $r \in R$, diremo $\deg(r) = \gamma$.
Ogni elemento $r \in R$ ha un'unica scrittura della forma $r = \sum_{\gamma\in\Gamma} r_\gamma$, dove $r_\gamma \in R_\gamma$. r_γ è detto *componente omogenea di grado* γ

Un esempio è costituito da $R = \mathbb{K}[x_1, \ldots, x_n]$, dove $\Gamma = \mathbb{N}$

$$R_i = \{f \in R \mid \forall t \in \mathrm{Supp}(f) : \deg(t) = i\}$$

Gli elementi di R_i sono chiamati *polinomi omogenei di grado* i oppure *forme di grado* i. Questa graduazione è usualmente chiamata *graduazione standard*. Inoltre gli insiemi R_i sono dei $\mathbb{K}$-spazi vettoriali di dimensione finita, per la precisione $\dim_{\mathbb{K}} R_i = \binom{n+i-1}{i}$.

Definizione 1.5.2 (Modulo graduato)**.** Sia R un anello graduato su un monoide Γ, e sia M un R-modulo. Diremo che M è Γ-graduato se esiste una famiglia di sottogruppi di M, $\{M_\gamma\}_{\gamma\in\Gamma}$ tali che:

- $M = \bigoplus_{\gamma\in\Gamma} M_\gamma$
- $R_\gamma \cdot M_{\gamma'} \subseteq M_{\gamma\gamma'}$, per ogni $\gamma, \gamma' \in \Gamma$

Negli anelli graduati è possibile definire una particolare classe di ideali: ideali che si "comportano bene" rispetto alla graduazione:

Definizione 1.5.3 (Ideali omogenei)**.** Sia R un anello graduato su Γ.
Diremo che un ideale I di R è *omogeneo* o Γ-*omogeneo* se, definito $(I)_\gamma = I \cup R_\gamma$, valgono:

1. $I = \bigoplus_{\gamma\in\Gamma} (I)_\gamma$.
2. $R_\gamma \cdot (I)_{\gamma'}$ per ogni $\gamma, \gamma' \in \Gamma$.

L'insieme $(I)_\gamma$ è detto *componente omogenea dell'ideale di grado* γ.

Nuovamente nel caso di $R = \mathbb{K}[x_1, \ldots, x_n]$ e graduazione standard, se I è un ideale omogeneo si ha che I_i, con $i \in \mathbb{N}$, è un $\mathbb{K}$-spazio vettoriale di dimensione finita.

Osservazione 1.5.1. Un criterio semplice per stabilire se un ideale è omogeneo è il seguente: se $I = (f_1, \ldots, f_m)$ ed f_i è omogeneo per ogni i, allora I è omogeneo.
Il contrario è ovviamente falso.

Proposizione 1.5.1 (Proprietà dei polinomi omogenei)**.**
Sia $R = \mathbb{K}[x_1, \ldots, x_n]$ un anello polinomiale graduato con un ordinamento compatibile con la graduazione (cioè il termine di testa di un polinomio è un termine di grado massimale nel supporto). Siano $f, g, f_1, \ldots, f_m$ polinomi omogenei. Allora valgono:

- *L'S-polinomio di f e g è omogeneo e ha grado $\geq \max\{\deg(f), \deg(g)\}$.*
- *Se $f \xrightarrow{f_1,\ldots,f_m} h$ allora h è omogeneo.*
- *Un polinomio f è riducibile solo da polinomi omogenei di grado minore od uguale a quello di f.*

Dalle prime due proprietà si deduce che, se i generatori omogenei di un ideale sono processati dall'algoritmo di Buchberger, tutti i polinomi prodotti saranno omogenei; la terza proprietà, infine, suggerisce di calcolare una base di Gröbner per un ideale omogeneo grado per grado.

Definizione 1.5.4 (Basi di Gröbner troncate)**.** Siano $R = \mathbb{K}[x_1, \ldots, x_n]$ un anello graduato, I un ideale omogeneo di R e d un naturale. Sia $G = \{g_1, \ldots, g_k\}$ un insieme di polinomi che generano un ideale $J \subseteq I$ tale che $(I)_k = (J)_k$ per ogni $k \leq d$. Un tale G si dice *base di Gröbner troncata al grado* d.

Adattare l'algoritmo di Buchberger per calcolare basi troncate è banale: basta scegliere dell'insieme delle coppie da processare una di grado minimo. Quando il minimo grado degli S-polinomi da processare è maggiore di d, allora la base calcolata fino a quel momento sarà una base troncata al grado d.
In sostanza una base di Gröbner troncata di un ideale I al grado d è un insieme di polinomi che genera un ideale che coincide con I fino al grado d. In particolare per d sufficientemente grande una base di Gröbner troncata al grado d sarà anche una base di Gröbner.

1.5.1 Algoritmo di Buchberger omogeneo

Queste osservazioni portano all'algoritmo noto come *algoritmo di Buchberger omogeneo*. Questo algoritmo procede come l'algoritmo usuale, solo che

ogni volta che deve scegliere una coppia, sceglie quella di grado minimo. Quindi procede esattamente come l'algoritmo per le basi troncate, solo che si arresta quando le coppie sono finite.
Nella pratica si è verificato, sperimentalmente, che questo modo di procedere è più efficiente in termini di velocità.

1.5.2 Funzione di Hilbert

Definizione 1.5.5 (Funzione di Hilbert)**.** Sia M un $\mathbb{K}[x_1, \ldots, x_n]$-modulo graduato su un monoide Γ, allora si definisce *funzione di Hilbert* di M:

$$\begin{array}{rccc} H_M: & \Gamma & \to & \mathbb{N} \\ & \gamma & \mapsto & \dim_{\mathbb{K}}(M_\gamma). \end{array}$$

La funzione di Hilbert che incontreremo più spesso sarà nel caso di I ideale omogeneo, di $P = \mathbb{K}[x_1, \ldots, x_n]$ con graduazione standard.

Esempio 3. Se $I = P = \mathbb{K}[x_1, \ldots, x_n]$ con graduazione standard, allora

$$H_I(d) = \binom{n+d-1}{n-1}$$

Definizione 1.5.6 (Serie di Hilbert-Poincaré)**.** Se H_I è la funzione di Hilbert di un ideale I, si definisce *serie di Hilbert-Poincaré* la sua serie generatrice:

$$HP_I = \sum_{d=0}^{\infty} H_I(d) z^d$$

che è un elemento di $\mathbb{Z}[[z]]$.

Teorema 1.5.2 (Hilbert)**.** *Se M è un modulo graduato finitamente generato su $\mathbb{K}[x_1, \ldots, x_n]$ allora $H_M(d)$ coincide, per d abbastanza grande, con un polinomio di grado $\leq n-1$.*

Definizione 1.5.7 (Polinomio di Hilbert)**.** Questo polinomio, denotato con $P_M(d)$, è detto *polinomio di Hilbert di M*.

Definizione 1.5.8 (Indice di regolarità)**.** Il più piccolo intero d_{reg} tale che per ogni $d \geq d_{reg}$ vale $H_M(s) = P_M(s)$ è detto *indice di regolarità di M*.

Teorema 1.5.3. *Se M è un modulo graduato finitamente generato su P, dove $P = \mathbb{K}[x_1, \ldots, x_n]$, e se d_{reg} è il suo grado di regolarità, allora se $\mathcal{G}$ è una base di Gröbner ridotta di M, vale $\forall m \in \mathcal{G} : \deg(m) \leq d_{reg}$.*

1.6 Applicazioni delle basi di Gröbner

In questa sezione introdurremo alcune semplici applicazioni delle basi di Gröbner che ci serviranno nel seguito.

1.6.1 Eliminazione

Eliminazione di indeterminate

Definizione 1.6.1 (Ideale di eliminazione)**.** Sia $P = \mathbb{K}[x_1, \ldots, x_n]$ un anello polinomiale e $L \subsetneq \{x_1, \ldots, x_n\}$ un sottoinsieme proprio delle indeterminate di P, e sia $\widehat{P} = \mathbb{K}[L]$ un sottoanello di P. Dato $I \subseteq P$ cerchiamo $I \cap \widehat{P}$ che è un ideale di $\widehat{P}$, detto *ideale di eliminazione di I rispetto a L.*

Definizione 1.6.2 (Ordinamento di eliminazione)**.** Con le definizioni precedenti, un ordinamento σ su $\mathbb{T}^n$ è detto *ordinamento di eliminazione* per L se, per ogni $f \in P \setminus \{0\}$ tale che $\mathrm{LT}_\sigma(f) \in \widehat{P}$, si ha $f \in \widehat{P}$.

Osservazione 1.6.1. La definizione sopra è equivalente al fatto che, per l'ordinamento σ, le indeterminate contenute in L sono minori delle altre, cioè se $x_i \in L$ e $x_j \notin L$ allora $x_i < x_j$.

Teorema 1.6.1 (Calcolo degli ideali di eliminazione)**.** *Sia $P = \mathbb{K}[x_1, \ldots, x_n]$ un anello polinomiale con ordinamento σ, I un ideale di P, $L \subsetneq \{x_1, \ldots, x_n\}$ un sottoinsieme proprio delle indeterminate di P, e sia $\widehat{P} = \mathbb{K}[L]$ un sottoanello di P. Inoltre sia $\hat{\sigma}$ la restrizione di σ ai termini $\widehat{\mathbb{T}}^n$ di $\widehat{P}$. Allora valgono le seguenti:*

1. *Abbiamo $\mathrm{LT}_{\hat{\sigma}}(I \cap \widehat{P}) = \mathrm{LT}_\sigma(I) \cap \widehat{P}$,*
2. *Sia G una σ-base di Gröbner di I, e sia $\widehat{G}$ l'insieme di tutti gli elementi di G che sono contenuti in $\widehat{P}$, allora l'insieme $\widehat{G}$ è una $\hat{\sigma}$-base di Gröbner di $I \cap \widehat{P}$,*
3. *Sia G la σ-base di Gröbner ridotta di I, allora $\widehat{G} = G \cap \widehat{P}$ è la $\hat{\sigma}$-base di Gröbner ridotta di $I \cap \widehat{P}$.*

Osservazione 1.6.2. L'ordinamento `Lex`, dato nella definizione 1.1.4, è un ordinamento di eliminazione per un qualsiasi $L = \{x_j, \ldots, x_n\}$, dove $1 < j < n$.

Eliminazione di componenti di modulo

Definizione 1.6.3 (Ordinamento di eliminazione di componente)**.** Sia $P = \mathbb{K}[x_1, \ldots, x_n]$ un anello polinomiale, sia $r \geq 1$ e sia $M \subseteq P^r$ un P-sottomodulo generato da $\mathcal{G} = \{g_1, \ldots, g_s\}$. Siano inoltre σ un ordinamento di modulo su $\mathbb{T}^n\langle e_1, \ldots, e_r\rangle$ e $L \subsetneq \{1, \ldots, r\}$. Definiamo $\widehat{P^r} = \bigoplus_{i \in \{1,\ldots,r\} \setminus L} P e_i$. L'ordinamento di modulo σ è detto *ordinamento di eliminazione di componente* per L se ogni elemento $m \in P^r \setminus \{0\}$ tale che $\mathrm{LT}_\sigma(m) \in \widehat{P^r}$ è contenuto in $\widehat{P^r}$. Il modulo $\widehat{M} = M \cap \widehat{P^r}$ è detto *modulo di eliminazione di componente* di M rispetto ad L.

Teorema 1.6.2 (Calcolo dei moduli di eliminazione)**.** *Nelle stesse ipotesi della definizione precedente sia $\hat{\sigma}$ la restrizione di σ ai termini di $\widehat{P^r}$, allora valgono le seguenti:*

1. $\mathrm{LT}_{\hat{\sigma}}(M \cap \widehat{P^r}) = \mathrm{LT}_{\sigma}(M) \cap \widehat{P^r}$,
2. *Se G è una σ-base di Gröbner di M l'insieme $\widehat{G} = G \cap \widehat{P^r}$ è una σ-base di Gröbner del modulo di eliminazione di componente $M \cap \widehat{P^r}$.*

1.6.2 Calcolo delle sizigie

Sia $P = \mathbb{K}[x_1, \ldots, x_n]$, consideriamo $f_1, \ldots, f_m \in P$, siamo interessati al calcolo delle sizigie di $\mathcal{F} = (f_1, \ldots, f_m)$. L'idea che useremo, introdotta in [CT98], è in certo senso simile a quella dell'algoritmo di Euclide esteso: calcoliamo una base di Gröbner di $I = (\mathcal{F})$ tenendo esplicitamente traccia della relazione che lega ogni polinomio che si andrà a costruire con i generatori iniziali dell'ideale.
Questo si può fare in modo trasparente usando nozioni già introdotte: consideriamo il P-modulo $P \oplus P^m \approx P^{m+1}$, e indichiamo con e_0 la base canonica di P, e con $e_1, \ldots, e_m$ la base canonica di P^m, consideriamo il modulo

$$M = \langle f_i e_0 + e_i \mid i = 1, \ldots, m \rangle_P .$$

Questi generatori possono anche essere rappresentati in una matrice:

$$\begin{pmatrix} f_1 & \cdots & f_m \\ 1 & & \\ & \ddots & \\ & & 1 \end{pmatrix}$$

dove la i-esima colonna rappresenta $g_i = f_i e_0 + e_i$. Sia $L = \{1, \ldots, m\}$, e mettiamo su $P \oplus P^m$ un qualsiasi ordinamento σ di eliminazione di componente di modulo per L.

Teorema 1.6.3 (Calcolo delle sizigie)**.** *Con le definizioni date sopra sia G una σ-base di Gröbner di M. Rappresentiamo gli elementi di G con una matrice, come si è fatto per i generatori di M, mettendo prima quegli elementi che hanno un elemento non nullo nella prima componente. Allora otteniamo una matrice della forma:*

$$\begin{pmatrix} \mathrm{GB}(I) & \\ \otimes & \mathrm{GB}(\mathrm{Syz}(\mathcal{F})) \end{pmatrix}$$

Dove $\mathrm{GB}(I)$ è una base di Gröbner di I rispetto a σ ristretto alla prima componente e $\mathrm{GB}(\mathrm{Syz}(\mathcal{F}))$ è una base di Gröbner delle sizigie di $\mathcal{F}$ rispetto a σ ristretto alle ultime componenti.

Dimostrazione. Basta osservare che M è il modulo costituito da tutti gli elementi della forma $(g, \phi(g))$, dove ϕ è la mappa che compare nella definizione di $\mathrm{Syz}(\mathcal{F})$. Pertanto si ha $m \in \mathrm{Syz}(\mathcal{F}) \iff m \in \ker(\phi) \iff (0, m) \in M$ e quindi $M \cap P^r$ è il modulo delle sizigie. □

Osservazione 1.6.3. Se, nell'esecuzione dell'algoritmo di Buchberger, consideriamo solo gli S-vettori tali che i loro termini di testa sono nella prima componente, allora la stessa procedura produce un sistema di generatori delle sizigie e non una base di Gröbner di queste.

Capitolo 2

Algoritmo F5 matriciale

Un problema dell'algoritmo di Buchberger è il fatto che quando si ottiene una riduzione a zero questa non ci dà nessuna informazione "nuova" sull'ideale né sulla sua base di Gröbner.
Le riduzioni a zero possono essere interpretate come la manifestazione di una sizigia del sistema di generatori $\mathcal{F}$, cioè una riduzione a zero costituisce una rappresentazione non banale di 0 come elemento dell'ideale I.
Questo deriva dal fatto più generale che non c'è un modo univoco di rappresentare un elemento di un ideale dato per generatori. Ad esempio, si consideri l'ideale $I = (\mathcal{F}) \subset \mathbb{Q}[x, y]$, con $\mathcal{F} = (x, y)$ ovviamente $xy \in I$, ma $xy = y \cdot x + 0 \cdot y$ come anche $xy = 0 \cdot x + x \cdot y$.
Il resto di questo capitolo sarà dedicato a definire un modo unico per rappresentare un elemento di un ideale. Da questo deriveremo un algoritmo che durante la sue esecuzione non produce informazioni inutili.

2.1 Nozioni preliminari

Sia $\mathbb{K}$ un campo sia e $n \in \mathbb{N}$. Poniamo $P = \mathbb{K}[x_1, \ldots, x_n], \tau$ l'anello polinomiale su $\mathbb{K}$ con n indeterminate e ordinamento τ.
Sia poi $\mathcal{F} = (f_1, \ldots, f_m) \in P^m$, dove $m \in \mathbb{N}$ e definiamo $I = (\mathcal{F})$ l'ideale di P generato da $\mathcal{F}$. Definiamo inoltre $I_0 = (0)$ e $I_k = (f_1, \ldots, f_k)$ $\forall k \in \{1, \ldots m\}$, da cui $I_m = I$.
Sia inoltre:

$$\begin{array}{rccc} \phi: & P^m & \to & I \\ & (g_1, \ldots, g_m) & \mapsto & \displaystyle\sum_{i=1}^{m} g_i f_i \end{array}$$

Dal momento che $I = \{\sum_{i=1}^{m} g_i f_i \mid (g_1, \ldots, g_m) \in P^m\}$ abbiamo che ϕ è surgettiva.

P^m può essere visto come $\mathbb{K}$-spazio vettoriale:

$$P^m = \langle te_i | t \in \mathbb{T}^n, i \in \{1, \ldots, m\} \rangle_{\mathbb{K}}$$

dove $\mathbb{T}^n$ è l'insieme dei termini di P, oppure come P-modulo libero:

$$P^m = \langle e_i | i = 1, \ldots, m \rangle_P$$

Mettiamo un ordine `PosTo` σ su P^m :

$$\begin{cases} te_i < t'e_j & \text{se } i < j \\ te_i < t'e_i & \text{se } t < t' \end{cases}$$

Definizione 2.1.1 (Sizigie principali)**.** Chiamiamo *sizigie principali* di $\mathcal{F}$ il P-sottomodulo:

$$\mathrm{PSyz}(\mathcal{F}) = \langle f_i e_j - f_j e_i | \, i \neq j = 1, \ldots, m \rangle_P \subseteq P^m$$

cioè l'insieme delle relazioni "ovvie" tra i generatori $\mathcal{F}$ dell'ideale I.

L'inclusione $\mathrm{PSyz}(\mathcal{F}) \subseteq \mathrm{Syz}(\mathcal{F})$ vale banalmente, dal momento che

$$\phi(f_i e_j - f_j e_i) = f_i f_j - f_j f_i = 0$$

da cui$f_i e_j - f_j e_i \in \ker \phi$.

Lemma 2.1.1. *Con le definizioni date precedentemente vale:*

$$\pi_k(\mathrm{PSyz}((f_1, \ldots, f_k))) = I_{k-1}$$

per ogni $k = 1, \ldots, m$. Dove π_k è la proiezione sulla k-esima componente, vale a dire l'insieme dei possibili coefficienti di e_k.

Dimostrazione. Dalla definizione di $\mathrm{PSyz}((f_1, \ldots, f_k))$:

$$\mathrm{PSyz}((f_1, \ldots, f_k)) = \langle f_j e_i - f_i e_j | i \neq j = 1, \ldots, k \rangle_P$$

Applicando π_k al lato destro dell'uguaglianza si ottiene:

$$\begin{aligned} &\pi_k(\langle f_j e_i - f_i e_j | i \neq j = 1, \ldots, k \rangle_P) = \\ &\pi_k(\langle f_i e_k | i = 1, \ldots, k-1 \rangle_P) = \\ &(f_1, \ldots, f_{k-1}) = I_{k-1} \end{aligned}$$

Che è la tesi. □

Corollario 2.1.2. *Se consideriamo tutte le sizigie, anzichè solo quelle principali vale il risultato più debole:*

$$\pi_k(\mathrm{Syz}((f_1, \ldots, f_k))) \supseteq I_{k-1}.$$

2.1.1 Successioni regolari

Definizione 2.1.2 (Successione regolare)**.**
Una successione di polinomi $f_1, \ldots, f_m \in P$ si dice *regolare* se

1. $(f_1, \ldots, f_m) \neq P$,
2. f_k non è un divisore dello zero in $P/(f_1, \ldots, f_{k-1})$, per ogni $k = 1, \ldots m$.

In generale la regolarità di una successione dipende dall'ordine dei polinomi.

Esempio 4. Consideriamo $P = \mathbb{K}[x, y, z]$, $\mathcal{F} = (x, (x-1)y, (x-1)z)$ è una successione regolare: $(x, (x-1)y, (x-1)z) = (x, y, z) \neq P$. In $P/(x) \approx \mathbb{K}[y, z]$ si ha $(x-1)y \equiv -y \pmod{x}$ che non è un divisore dello zero in $\mathbb{K}[y, z]$, in $P/(x, (x-1)y) = P/(x, y) \approx \mathbb{K}[z]$ si ha $(x-1)z \equiv -z \pmod{(x, y)}$ che non è un divisore dello zero.
Consideriamo invece $\mathcal{F}' = ((x-1)y, (x-1)z, x)$: in $P/((x-1)y)$ si ha che $(x-1)z$ è un divisore dello zero i quanto è annullato da y $(x-1)z \cdot y = (x-1)y \cdot z = 0$, e $y \notin ((x-1)z)$. Perciò $\mathcal{F}'$ non è una successione regolare.

Tuttavia quando i polinomi f_k sono omogenei allora la regolarità non dipende dall'ordine.

Lemma 2.1.3. *Sia $\mathcal{F} = (f_1, \ldots, f_m)$ una successione di polinomi non nulli, anche non omogenei, in $P = \mathbb{K}[x_1, \ldots, x_n]$ tali che*

$$\mathrm{Syz}(\mathcal{F}) = \mathrm{PSyz}(\mathcal{F}),$$

cioè le sizigie non *sono solo le sizigie principali, allora $\mathcal{F}$ non è una successione regolare.*

Dimostrazione. Sia $h = (h_1, \ldots, h_m) \in \mathrm{PSyz}(\mathcal{F}) \setminus \mathrm{Syz}(\mathcal{F})$ una sizigia non principale, mettiamo un qualsiasi ordine del tipo `PosTo` su P^m, e supponiamo h in forma normale rispetto a $\mathrm{Syz}(\mathcal{F})$. Perciò h sarà della forma

$$h = (h_1, \ldots, h_k, 0, \ldots, 0),$$

con $h_k \notin I_{k-1}$ per il lemma 2.1.1, perció $h_k f_k = -\sum_{i=1}^{k-1} h_i f_i \in I_{k-1}$ quindi f_k è un divisore dello zero in I_{k-1}, e, dal momento che h_k non è zero nel quoziente P/I_{k-1} la successione $\mathcal{F}$ non è regolare. □

Visto che nel seguito tratteremo spesso successioni di polinomi omogenei, diremo che è una successione regolare per intendere che le sizigie sono solo quelle principali.

Proposizione 2.1.4 (Serie di Hilbert-Poincaré di una successione regolare)**.** *Se $\mathcal{F} = (f_1, \ldots, f_m)$ è una successione di polinomi omogenei con $d_i = \deg(f_i)$, e I è l'ideale generato da $\mathcal{F}$, allora $\mathcal{F}$ è regolare se e solo se la serie di Hilbert-Poincaré di P/I è:*

$$HP_{P/I} = \frac{1}{(1-z)^n} \prod_{i=1}^{m} \left(1 - z^{d_i}\right).$$

Dimostrazione. Dalla regolarità derivano le seguenti successioni esatte:

$$0 \to P/I_{i-1} \xrightarrow{f_i} P/I_{i-1} \to P/I_i \to 0$$

dove $I_i = (f_1, \ldots, f_i)$, allora si ha che le serie di Hilbert-Poincaré verificano $z^{d_i} HP_{P/I_{i-1}} - HP_{P/I_{i-1}} + HP_{P/I_i} = 0$, dove $d_i = \deg(f_i)$; sapendo che $HP_P = \frac{1}{(1-z)^n}$ si conclude:

$$HP_{P/I} = \frac{1}{(1-z)^n} \prod_{i=1}^{m} \left(1 - z^{d_i}\right)$$

Per il viceversa consideriamo le successioni esatte

$$0 \to J \to P/I_{i-1} \xrightarrow{f_i} P/I_{i-1} \to P/I_i \to 0$$

dove J è il nucleo della moltiplicazione per f_i nell'anello P/I_{i-1}. Con ragionamento analogo a quello precedente si deduce che la serie di Hilbert-Poincaré di J è identicamente nulla, perció $J = \{0\}$ e la successione è esatta. □

Corollario 2.1.5. *Se $\mathcal{F}$ è una successione di polinomi omogenei, per sapere se $\mathcal{F}$ è regolare è sufficiente calcolare una base di Gröbner dell'ideale generato da $\mathcal{F}$, e poi con questa ricavare la funzione di Hilbert ed usare la proposizione precedente.*

Proposizione 2.1.6. *Sia $\mathcal{F} = (f_1, \ldots, f_m)$ una successione di polinomi omogenei in $P = \mathbb{K}[x_1, \ldots, x_n]$ tali che $f_i \neq 0$ per ogni i, allora $\mathcal{F}$ è regolare se e solo se*

$$\mathrm{Syz}(\mathcal{F}) = \mathrm{PSyz}(\mathcal{F}),$$

cioè le sizigie coincidono con le sizigie principali.

Dimostrazione. Un'implicazione è un caso particolare del lemma 2.1.3, per il viceversa si veda [Eis95]. □

2.2 Isomorfismo con le sizigie

Supponendo di conoscere le sizigie di $\mathcal{F}$ allora possiamo usare effettivamente il seguente isomorfismo di moduli. Ricordando che $\ker(\phi) = \mathrm{Syz}(\mathcal{F})$

e che $\mathrm{Im}(\phi) = I$, si ha che la mappa indotta da ϕ sul modulo quoziente ${}^{P^m}/_{\mathrm{Syz}(\mathcal{F})}$,

$$\begin{aligned} \phi : \quad P^m/\operatorname{Syz}(\mathcal{F}) &\to I \\ (g_1, \ldots, g_m) &\mapsto \sum_{i=1}^{m} g_i f_i \end{aligned}$$

induce un isomorfismo di P-moduli:

$$I \simeq \frac{P^m}{\operatorname{Syz}(\mathcal{F})} \tag{2.2.1}$$

Supponendo, inoltre, di avere una base di Gröbner di $\mathrm{Syz}(\mathcal{F})$ possiamo rappresentare in modo unico un elemento di $P^m/\operatorname{Syz}(\mathcal{F})$, prendendo la forma normale rispetto ad una base di Gröbner di $\mathrm{Syz}(\mathcal{F})$, e quindi di I:

$$I = \left\{ \phi(\mathbf{g}) \,\middle|\, \mathbf{g} \in P^m,\ \mathbf{g} = \mathrm{NF}_{\mathrm{Syz}(\mathcal{F})}(\mathbf{g}) \right\} \tag{2.2.2}$$

Vedendo I come uno spazio vettoriale su $\mathbb{K}$, questa relazione si può riscrivere come:

$$I = \langle \phi(te_k) \,|\, te_k \notin \operatorname{LT}\operatorname{Syz}(\mathcal{F}) \quad \forall k = 1, \ldots, m \rangle_{\mathbb{K}} \tag{2.2.3}$$

In pratica l'idea è di rappresentare un $\phi(\mathbf{g}) = f \in I$ con la forma normale di $\mathbf{g}$ rispetto alle sizigie.

Osservazione 2.2.1. Le relazioni 2.2.2 e 2.2.3 valgono indipendentemente dalla regolarità di $\mathcal{F}$ e dall'ordine dato sul modulo delle sizigie.

Ora, apparentemente, per essere effettive le relazioni 2.2.2 e 2.2.3 richiedono la conoscenza di una base di Gröbner sizigie di $\mathcal{F}$, che sono più costose da calcolare della base di Gröbner di I.
Tuttavia, avendo a disposizione un insieme di generatori di $\mathrm{LT}(\mathrm{Syz}(\mathcal{F}))$, si può stabilire facilmente se un elemento è o non è in forma normale rispetto a $\mathrm{Syz}(\mathcal{F})$.

Lemma 2.2.1. *Sia S un insieme di generatori di $\mathrm{LT}(\mathrm{Syz}(\mathcal{F}))$ e sia $p \in P^m$, con $\mathrm{LT}(p) = te_k$, dove $t \in \mathbb{T}^n$, e sia $f = \phi(p)$ un elemento di I.*
Se esiste $t'e_k \in S$ tale che $t'|t$ allora esiste $p' \in P^m$ tale che $f = \phi(p')$ e $\mathrm{LT}(p') < \mathrm{LT}(p)$.

Dimostrazione. L'ipotesi implica che p non è in forma normale rispetto a $\mathrm{Syz}(\mathcal{F})$, perciò ponendo $p' = \mathrm{NF}_{\mathrm{Syz}(\mathcal{F})}(p)$ si ha che $p' \neq p$ da cui $\mathrm{LT}(p') < \mathrm{LT}(p)$. □

2.2.1 Ideali omogenei

D'ora in avanti lavoreremo con polinomi ed ideali **omogenei**, in quanto vorremo calcolare basi di Gröbner troncate successive.
Diamo le seguenti definizioni:

- $(P)_d = \{f \in P \mid \deg(f) = d\}$, cioè $(P)_d$ è l'insieme dei polinomi di grado d.
- $(\mathbb{T}^n)_d = \{t \in T^n \mid \deg(t) = d\}$ è l'insieme dei termini di grado d.
- $(I_k)_d = I_k \cap P_d$, cioè $(I_k)_d$ è la componente omogenea di I_k di grado d.
- $\deg(te_i) = \deg(t) + \deg(f_i)$.

Abbiamo quindi:

$$(I_k)_d = \left\{ \sum_{i=1}^{k} g_i f_i \,\middle|\, g_i \in P,\ \deg(g_i) + \deg(f_i) = d \right\}$$

Se prendiamo le forme normali dei $(g_1, \ldots, g_k, 0, \ldots, 0)$ otteniamo:

$$(I_k)_d = \left\{ \sum_{i=1}^{k} g_i f_i \,\middle|\, \deg(g_i) + \deg(f_i) = d,\ \mathbf{g} = \mathrm{NF}_{\mathrm{Syz}\,\mathcal{F}}(\mathbf{g}) \in P^m \right\}$$

Se $\mathcal{S}$ è un sistema di generatori per $\mathrm{LT}(\mathrm{Syz}(\mathcal{F}))$, e se con $\mathrm{Supp}(g_i)$ indichiamo il supporto di g_i, allora la condizione $\mathbf{g} = \mathrm{NF}_{\mathrm{Syz}(\mathcal{F})}(\mathbf{g})$ si riduce a $\forall t \in \mathrm{Supp}(g_i) : te_i \notin \mathrm{LT}(\mathrm{Syz}(\mathcal{F}))$, da cui:

$$(I_k)_d = \left\{ \sum_{i=1}^{k} g_i f_i \,\middle|\, \deg(g_i) + \deg(f_i) = d,\ \forall t \in \mathrm{Supp}(g_i) : te_i \notin \mathrm{LT}(\mathrm{Syz}\,\mathcal{F}) \right\} \tag{2.2.4}$$

Quest'ultima uguaglianza può essere letta in termini di spazi vettoriali come segue:

$$(I_k)_d = \langle tf_i \,|\, i \leq k, \quad \deg(tf_i) = d, \quad te_i \notin \mathrm{LT}(\mathrm{Syz}(\mathcal{F})) \rangle_{\mathbb{K}} \tag{2.2.5}$$

2.2.2 Caso regolare

Se aggiungiamo l'ipotesi che la successione $\mathcal{F}$ sia regolare possiamo usare il lemma 2.1.1 per esplicitare ulteriormente l'insieme $\mathrm{LT}(\mathrm{Syz}(\mathcal{F}))$ e riscrivere 2.2.5 nel seguente modo:

$$(I_k)_d = \langle tf_i \,|\, i \leq k, \quad \deg(tf_i) = d, \quad t \notin \mathrm{LT}(I_{i-1}) \rangle_{\mathbb{K}} \tag{2.2.6}$$

Quest'ultima rappresentazione dell'ideale è effettiva se nel momento in cui si considera I_k abbiamo a disposizione una base di Gröbner di I_{k-1}; questo suggerisce di procedere incrementalmente, vale a dire di calcolare prima una base di Gröbner di $I_1 = (f_1)$, poi di $I_2 = (f_1, f_2)$, e così via.

2.3 Polinomi etichettati

Al fine di poter usare il lemma 2.2.1 abbiamo bisogno di tener traccia della relazione che lega un polinomio $f = \phi(g)$ ai generatori dell'ideale $\mathcal{F}$; tuttavia, come visto precedentemente non abbiamo davvero bisogno di mantenere in memoria "tutto" g, è sufficiente avere a nostra disposizione $\mathrm{LT}(g)$.
Per brevità di notazione, d'ora in poi porremo

$$L = \mathbb{T}^n\langle e_1, \ldots, e_r\rangle = \{te_i \,|\, t \in \mathbb{T}^n,\, i = 1, \ldots, m\}.$$

Definizione 2.3.1 (Etichette)**.** D'ora in poi ci riferiremo agli elementi di L come *etichette*.

Definizione 2.3.2 (Etichette normali)**.** Diremo *etichette normali* l'insieme

$$L' = \{l \in L \,|\, l \notin \mathrm{LT}(\mathrm{Syz}(\mathcal{F}))\}$$

Per quanto detto precedentemente in 2.2.5 $\langle L'\rangle_{\mathbb{K}}$ è uno spazio vettoriale isomorfo a I ed una base, sempre come spazio vettoriale, di ${}^{P^m}/_{\mathrm{Syz}(\mathcal{F})}$.

Definizione 2.3.3 (Polinomio etichettato)**.** Chiameremo inoltre *polinomio etichettato* (rispetto ad $\mathcal{F}$) un elemento $(l, f) \in L \times P$.
Un polinomio etichettato (l, f) tale che esiste un $p \in P^m$ che verifica

- $\mathrm{LT}(p) = l$,
- $\phi(p) = f$.

sarà chiamato *polinomio etichettato valido*.
Un polinomio etichettato (l, f) la cui etichetta è normale sarà chiamato *polinomio etichettato normale*.

2.3.1 Operazioni sui polinomi etichettati

Abbiamo, in parte, visto che il termine di testa della relazione che lega un polinomio ai generatori dell'ideale cui appartiene permette di ricavare delle informazioni importanti. Bisogna, però, vedere quanto e come questa informazione si riesca a mantenere quando si fanno operazioni sui polinomi.

Definizione 2.3.4 (Operazioni sui polinomi etichettati)**.** Definiamo ora alcune operazioni sui polinomi etichettati che ci serviranno nell'algoritmo. Siano (l_f, f) e (l_g, g) due polinomi etichettati.

1. Se $c \in \mathbb{K}^*$ poniamo $c \cdot (l_f, f) = (l_f, cf)$
2. Se $l_f \neq l_g$ poniamo $(l_f, f) + (l_g, g) = (\max(l_f, l_g), f + g)$

3. Se $t \in \mathbb{T}^n$ poniamo $t \cdot (l_f, f) = (tl_f, tf)$

Lemma 2.3.1. *Se (l_f, f) e (l_g, g) sono validi, allora le tre operazioni definite sopra restituiscono polinomi etichettati validi.*

Dimostrazione. Siano p_f e $p_g \in P^m$ tali che $\phi(p_f) = f$, $\phi(p_g) = g$, $\mathrm{LT}(p_f) = l_f$ e $\mathrm{LT}(p_g) = l_g$.
Per l'operazione 1 basta considerare cp_f e si vede che questo soddisfa: $\phi(cp_f) = cf$ e $\mathrm{LT}(cp_f) = l_f$. Per l'operazione 2 basta considerare $p_f + p_g$ e osservare che $\phi(p_f + p_g) = f + g$ e $\mathrm{LT}(p_f + p_g) = \max(l_f, l_g)$, dal momento che, per via dell'ipotesi $l_f \neq l_g$, non ci può essere cancellazione dei termini di testa. Infine per l'operazione 3 basta considerare tp_f. □

Lemma 2.3.2. *Se (l_f, f) e (l_g, g) sono polinomi etichettati normali allora le operazioni 1 e 2 restituiscono polinomi etichettati normali.*

Dimostrazione. Per l'operazione 1 è ovvio, per l'operazione 2 è sufficiente notare che $l_f, l_g \in L'$ implica $\max(l_f, l_g) \in L'$. □

2.4 Matrice di Macaulay

Dato un ideale omogeneo $I = (f_1, \ldots, f_m) \in \mathbb{K}[x_1, \ldots, x_n]$ ed un intero d, le matrici di Macaulay sono un modo di rappresentare lo spazio vettoriale, di dimensione finita, $(I)_d$. Indichiamo con $(\mathbb{T}^n)_d$ l'insieme dei termini di grado d.

Definizione 2.4.1. Si definisce *matrice di Macaulay* $\mathcal{M}_d$ di I in grado d la matrice che ha una colonna per ogni elemento di $(\mathbb{T}^n)_d$, ed una riga per ogni polinomio della forma tf_h con $h = 1, \ldots, m$ e $t \in \mathbb{T}^n$ tale che $\deg(tf_h) = d$. $\mathcal{M}_d$ ha quindi $\binom{n+d-1}{d}$ colonne, ordinate rispetto al term ordering. L'elemento di $\mathcal{M}_d$ in posizione (tf_h, μ) con $\mu \in (\mathbb{T}^n)_d$ è $\mathrm{Coeff}(\mu, tf_h)$, il coefficiente di μ in tf_h:

$$\mathcal{M}_d = (\mathrm{Coeff}(\mu, tf_h))_{\mu \in (\mathbb{T}^n)_d, \quad \deg(tf_h) = d}$$

Osservazione 2.4.1. Le righe di $\mathcal{M}_d$ sono un sistema di generatori, come $\mathbb{K}$-spazio vettoriale, di $(I)_d$, tuttavia, in generale, questi non sono una base.

Per costruzione, il rango di $\mathcal{M}_d$ è la funzione di Hilbert dell'ideale valutata in d.

Osservazione 2.4.2. Un semplice algoritmo per calcolare basi di Gröbner troncate al grado d, usando le matrici di Macaulay, è il seguente:

1. Per ogni intero $n = 1, \ldots, d$ si costruisce matrice $\mathcal{M}_n$.

2. Si applica una riduzione di Gauss ad $\mathcal{M}_n$, ottenendo un'altra matrice $\mathcal{M}'_n$.

3. Per ogni riga di $\mathcal{M}'_n$ si considera il polinomio corrispondente e, se questo ha un termine di testa non multiplo di altri termini di testa, lo si aggiunge all'insieme di generatori.

2.4.1 Matrici di Macaulay ridotte

In questa sezione cercheremo, usando i polinomi etichettati, di dare una costruzione esplicita di una matrice che abbia ancora la proprietà di avere le righe (interpretate come polinomi) che generano la componente omogenea $(I_k)_d$ di un ideale, ma che sia anche di rango massimo, cioè con un numero di righe pari al suo rango.

Definizione 2.4.2 (Matrici di Macaulay ridotte)**.** Chiameremo $\overline{\mathcal{M}}_d$ *Matrice di Macaulay ridotta* di I in grado d una qualsiasi matrice che ha le stesse colonne di $\mathcal{M}_d$, e le cui righe generino I_d, in particolare vale: $\mathrm{Rk}(\overline{\mathcal{M}}_d) = \mathrm{Rk}(\mathcal{M}_d)$.

Lemma 2.4.1. *Consideriamo l'insieme di etichette normali* $L_{k,d} = \{te_i \,|\, i \leq k,\ \deg(te_i) = d\} \cap L'$ *e sia* $V_{k,d} = \langle L_{k,d} \rangle_{\mathbb{K}}$*, allora abbiamo, da 2.2.5 che:*

$$\phi_{k,d} = \phi|_{V_{k,d}} : V_{k,d} \to (I_k)_d$$

è un isomorfismo di $\mathbb{K}$ *spazi vettoriali.*

Lemma 2.4.2. *Sotto l'ipotesi di regolarità, usando 2.2.6 possiamo dare una descrizione più esplicita dell'insieme delle etichette* $L_{k,d}$*:*

$$L_{k,d} = \{te_i \,|\, i \leq k,\ \deg(te_i) = d,\ t \notin \mathrm{LT}(I_{i-1}))\}$$

Pertanto possiamo costruire una matrice di Macaulay ridotta $\overline{\mathcal{M}}_{k,d}$ inserendo solo i polinomi $\phi(L_{k,d})$, che sono, in effetti, un sottoinsieme dei polinomi inseriti nella matrice non ridotta.

Osservazione 2.4.3. Quando inseriamo un polinomio nella matrice teniamo traccia della sua etichetta, in pratica indicizziamo le righe con le stesse etichette dei polinomi.

Esempio 5. Siano $P = \mathbb{Q}[x, y]$, $\mathcal{F} = (x, x + y)$, $k = d = 2$. $\mathcal{F}$ è banalmente una successione regolare. Applicando la definizione si trova $L_{2,2} = \{xe_1, ye_1, ye_2\}$, dove $xe_2 \notin L_{2,2}$ perchè $x \in I_1 = (x)$.
Quindi la matrice $\mathcal{M}_{2,2}$ avrà $|L_{2,2}| = 3$ righe. L'insieme dei termini di grado 2 è $\{x^2, xy, y^2\}$, supponiamo che le colonne siano indicizzate con

questi termini in quest'ordine, che è ad esempio compatibile con un ordine Lex in cui $x > y$. Pertanto la matrice avrà questa forma:

$$\overline{\mathcal{M}}_{2,2} = \begin{matrix} xe_1 \\ ye_1 \\ ye_2 \end{matrix} \begin{pmatrix} 1 & 1 & 0 \\ 0 & 1 & 1 \\ 0 & 1 & 0 \end{pmatrix}$$

Dove i tre polinomi etichettati rappresentati della matrice sono:

$$(xe_1, x^2 + xy),\ (ye_1, xy + y^2),\ (ye_2, xy)$$

Rispetto alla matrice di Macaulay non ridotta è stata eliminata la riga corrispondente al polinomio (xe_2, x^2).

Un insieme di etichette valide dei polinomi $\phi(L_{k,d})$ è, ovviamente, $L_{k,d}$, tuttavia possiamo rimpiazzare l'insieme $\phi(L_{k,d})$ con un qualsiasi altro insieme di polinomi che generino lo spazio vettoriale $V_{k,d}$, in particolare vale il seguente:

Lemma 2.4.3. *Sia $X = \{(l_f, f)\}$ un insieme di polinomi etichettati validi, tale che $L_X = \{l_f \mid (l_f, f) \in X\}$, l'insieme delle etichette di X, sia uguale a $L_{k,d}$, allora $F_X = \{f \mid (l_f, f) \in X\}$, l'insieme dei polinomi di X, è una base di $(I_k)_d$:*

$$\langle F_X \rangle_{\mathbb{K}} = (I_k)_d$$

Dimostrazione. Dal momento che (l_f, f) è valido esiste un $p_f \in P^m$ tale che $\phi(p_f) = f$ e $\mathrm{LT}(p_f) = l_f$, chiamiamo P_X l'insieme dei p_f costruiti in questo modo.
Possiamo inoltre supporre, che ogni p_f sia in forma normale rispetto a $\mathrm{Syz}(\mathcal{F})$, quindi $p_f \in \langle L_{k,d} \rangle_{\mathbb{K}}$.
Ora basta osservare che ogni p_f ha un termine di testa differente, quindi P_X è un insieme di vettori linearmente indipendenti, e che $|P_X| = |L_{k,d}|$. Questo dà $\langle P_X \rangle_{\mathbb{K}} = \langle L_X \rangle_{\mathbb{K}} = \langle L_{k,d} \rangle_{\mathbb{K}} = V_{k,d}$.
Applicando ϕ ad entrambi i membri si ha:

$$\phi(\langle P_X \rangle_{\mathbb{K}}) = \langle F_X \rangle_{\mathbb{K}} = (I_k)_d$$

□

Quindi, per costruire una matrice le cui righe siano una base per $(I_k)_d$, serve soltanto un polinomio etichettato (l, f) per ciascuna etichetta $l \in L_{k,d}$.
Concludendo, se abbiamo un insieme di polinomi etichettati che generano I_k, il modo più generale per costruire una matrice di Macaulay ridotta in grado d è questo: per ogni etichetta $l \in L_{k,d}$ prendiamo un polinomio della forma $t(l_f, f)$, dove f è un generatore di I_k con etichetta l e $tl_f = l$.
Inoltre, avendo a disposizione i polinomi della forma $(1e_i, f_i)$, è evidente che ogni etichetta di $L_{k,d}$ è ottenibile con questo procedimento.

Osservazione 2.4.4. In generale succederà spesso di avere più polinomi etichettati la cui etichetta divide un'etichetta valida, algoritmicamente pare più efficiente scegliere il polinomio di grado più alto, cioè quello che deve essere moltiplicato per il termine di grado minore, oppure quello che porta ad avere termine di testa più piccolo.

2.4.2 Eliminazione di Gauss di matrici con etichette

Data una matrice $\overline{\mathcal{M}}_{k,d}$, le cui righe siano una base di $(I_k)_d$, possiamo fare un'eliminazione di Gauss per ottenere un insieme di polinomi con termini di testa distinti e quindi coincidente con $\mathrm{LT}((I_k)_d)$, da cui si può poi facilmente ottenere una base di Gröbner troncata dell'ideale.
L'unica cosa di cui bisogna aver cura sono le etichette se alla fine dell'eliminazione vogliamo ancora ottenere un insieme di polinomi etichettati. Le operazioni che abbiamo definito sui polinomi etichettati ci permettono di fare combinazioni lineari e quindi anche un'eliminazione di Gauss.
Tuttavia possiamo imporre ancora qualcosa in più: possiamo richiedere che alla fine dell'eliminazione l'insieme delle etichette coincida con quello di partenza. L'eliminazione si effettua facendo, in successione, una serie di combinazioni lineari di due righe. Questa operazione lascia dei gradi di libertà, in particolare si può scegliere di non modificare una delle due righe e di alterare solo l'altra.
Possiamo sfruttare questa libertà per "salvaguardare" al meglio le etichette: supponiamo di avere due righe (l_1, r_1), (l_2, r_2), con $l_1 < l_2$, tali che il primo elemento non nullo è nella stessa colonna, diciamo la j-esima:

$$\begin{aligned}(l_1, r_1) &= l_1 \quad (0, \ldots, 0, r_{1,j}, \ldots)\\ (l_2, r_2) &= l_2 \quad (0, \ldots, 0, r_{2,j}, \ldots)\end{aligned}$$

Dove $r_{1,j}, r_{2,j} \in \mathbb{K}^*$.
Ricordando come sono state definite le operazioni sui polinomi etichettati, sappiamo che se sommiamo alla prima riga r_1 un multiplo della seconda l'etichetta di questa diventerebbe $\max\{l_1, l_2\} = l_2$ che non andrebbe bene; allora sommiamo a r_2 un multiplo di r_1: sostituiamo r_2 con $r_2 - \frac{r_{2,j}}{r_{1,j}} r_1$ così facendo, l'etichetta della nuova riga r_2 è ancora l_2.
Abbiamo quindi dimostrato il seguente:

Proposizione 2.4.4. *Sia $\mathcal{M}$ una matrice di Macaulay con le righe etichettate con etichette tutte distinte X. È possibile ridurre la matrice a scala in modo da lasciare invariato l'insieme delle etichette.*

Esempio 6. Eseguiamo l'eliminazione sulla matrice $\overline{\mathcal{M}}_{2,2}$ costruita nell'esempio 5;

$$\overline{\mathcal{M}}_{2,2} = \begin{matrix} xe_1 \\ ye_1 \\ ye_2 \end{matrix} \begin{pmatrix} 1 & 1 & 0 \\ 0 & 1 & 1 \\ 0 & 1 & 0 \end{pmatrix}$$

Le uniche righe su cui bisogna intervenire sono le ultime due. Ora $ye_2 > ye_1$, perciò la riga corrispondente a ye_1 non verrà modificata, mentre alla terza riga verrà sottratta la seconda, ottenendo così:

$$\overline{\mathcal{M}}'_{2,2} = \begin{matrix} xe_1 \\ ye_1 \\ ye_2 \end{matrix} \begin{pmatrix} 1 & 1 & 0 \\ 0 & 1 & 1 \\ 0 & 0 & -1 \end{pmatrix}$$

Rileggendo le righe come polinomi, troviamo una base come $\mathbb{K}$-spazio vettoriale di $(I_2)_2$ costituita da polinomi con termini di testa diversi:

$$\left\{x^2,\ xy + y^2,\ -y^2\right\}.$$

2.5 L'algoritmo

Anzitutto, ogni polinomio che tratteremo avrà la sua etichetta, quindi cominciamo immergendo i generatori dell'ideale I in $L \times P$:

$$f_i \mapsto (1e_i, f_i) \qquad \forall i = 1, \ldots, m$$

Questi sono ovviamente polinomi etichettati validi.
L'idea è di calcolare basi di Gröbner in ordine per gli ideali $I_1, I_2, \ldots, I_m = I$, in modo da conoscere $\mathrm{LT}(I_{k-1})$ quando si calcola la base di Gröbner di I_k. Questo permette, usando il lemma 2.4.2, di determinare esplicitamente l'insieme $L_{k,d}$ delle etichette valide ad ogni passo.

2.5.1 Criterio di arresto

L'unica cosa che manca è come stabilire fino a che grado sia necessario procedere per essere certi di aver ottenuto una base di Gröbner di un ideale I_k, prima di poter passare all'ideale successivo I_{k+1}.
Nel caso regolare ci si può avvalere del seguente teorema:

Teorema 2.5.1 (Bound di Macaulay)**.** *Sia $(f_1, \ldots, f_m)$ una successione regolare, con $\deg(f_i) = d_i$ e sia d_{reg} l'indice di regolarità di $I = (f_1, \ldots, f_m)$, allora vale:*

$$d_{reg} \leq 1 + \sum_{k=1}^{m} (d_k - 1)$$

Dimostrazione. Per la dimostrazione vedere [Laz83]. □

Questo risultato, grazie al teorema 1.5.3, ci dice quando possiamo essere certi che una base di Gröbner troncata è una base di Gröbner.

2.5.2 Pseudocodice

Dati $\mathcal{F} = (f_1, \ldots, f_m)$ polinomi omogenei, $m \geq 2$, tali che $\mathcal{F}$ sia una successione regolare il seguente algoritmo calcola una base di Gröbner di $I = (f_1, \ldots, f_k)$.

Algoritmo 2.5.1 (F5 matriciale con generatori regolari)**.**

Input: $\mathcal{F} = (f_1, \ldots, f_m)$ successione regolare di polinomi omogenei.

Output: Una base di Gröbner dell'ideale $I = (\mathcal{F})$.

1. Etichetta i generatori: $f_i \mapsto (1e_i, f_i)$
2. $GB := \{(1e_1, f_1)\}$
3. For $k = 2, \ldots, m$ do:
   (a) $GB := GB \cup \{(1e_k, f_k)\}$
   (b) For $d = d_k, \ldots, 1 + \sum_{h=1}^{k}(d_h - 1)$:
      i. Calcola $L_{k,d}$ un insieme di etichette come definito nel lemma 2.4.2:
      $$L_{k,d} = \{te_i \mid i \leq k,\ \deg(te_i) = d,\ t \notin \mathrm{LT}(I_{i-1}))\}$$
      ii. Costruisci una matrice di Macaulay ridotta $\overline{\mathcal{M}}_{k,d}$ con righe etichettate degli elementi di $L_{k,d}$
      iii. Esegui un'eliminazione di Gauss su $\overline{\mathcal{M}}_{k,d}$ per ottenere $\widetilde{\mathcal{M}}_{k,d}$
      iv. Per ogni riga (te_s, f) di $\widetilde{\mathcal{M}}_{k,d}$, se $\mathrm{LT}(f) \notin \mathrm{LT}(GB)$ aggiungi (te_s, f) a GB
4. Return $\{f \mid (l_f, f) \in GB\}$.

Teorema 2.5.2. *L'algoritmo 2.5.1 restituisce una base di Gröbner dell'ideale generato da $f_1, \ldots, f_m$.*

Dimostrazione. Per ottenere la tesi è sufficiente dimostrare che all'inizio di ogni iterazione del ciclo for interno, passo 3b, GB contiene una base di Gröbner di I_{k-1} ed una base di Gröbner troncata al grado $d-1$ di I_k, e gli elementi di GB di grado $\leq d-1$ sono esattamente una base di Gröbner troncata al grado $d-1$ di I_k.
Quando viene applicato il lemma 2.4.2 questo è effettivo, poichè è noto $\mathrm{LT}(I_{k-1})$.
Il ciclo più interno, passo 3b, calcola una base troncata di I_k in grado d partendo da una base troncata in grado $d-1$ per il lemma 2.4.3.
Quando viene lasciato il ciclo più interno GB contiene una base di Gröbner di I_k per il teorema 2.5.1.

Nel ciclo esterno, passo 3, vengono solo aggiunti a GB i generatori f_i uno alla volta, questa operazione manda una base di Gröbner di I_k in una base troncata di I_{k+1} al grado d_k. □

Proposizione 2.5.3. *Il ciclo 3b si può far cominciare dal passo $d = d_k + 1$ se si sostituisce f_k con $\mathrm{NF}_{I_{k-1}}(f_k)$.*

Dimostrazione. Nel passo con $d = d_k$, $L_{k,d_k} \setminus \{1e_k\}$ contiene solo etichette della forma te_s con $t \in T^n$ e $s < k$. Alla fine dell'eliminazione a queste corrisponderanno polinomi appartenenti ad I_{k-1}, quindi non avranno termini di testa nuovi. L'unico possibile termine di testa nuovo che si può scoprire è quello associato all'etichetta $1e_k$.
Se f_k, che è il polinomio associato ad $1e_k$ prima dell'eliminazione, è in forma normale rispetto a I_{k-1}, allora non potrà mai succedere che questo abbia lo stesso termine di testa di un altro polinomio presente nella matrice dal momento che questi sono contenuti in I_{k-1}. Perciò alla fine dell'eliminazione a $1e_k$ corrisponderà ancora f_k. □

2.6 Osservazioni finali

L'algoritmo 2.5.1, qui presentato, è quello descritto in [Bar06]. Nello pseudocodice ci sono ancora dei gradi di libertà, in particolare legati alla costruzione delle matrici; alla fine di un passo anzichè "buttare" tutti i polinomi che non danno termini di testa nuovi questi vengono tenuti in memoria. In questo modo si può dire che questo algoritmo costruisca una successione di basi per le varie componenti omogenee dell'ideale. Tenendo tutti questi polinomi si ha una grande libertà nella costruzione delle matrici successive. Intuitivamente pare (e risulta sperimentalmente) conveniente costruire le matrici prendendo multipli di polinomi di grado più alto possibile, o equivalentemente prendendo quei polinomi che devono essere moltiplicati per termini di grado minimo che in pratica sarà 1. L'effettiva implementazione di queste parti dell'algoritmo è trattata in maggiore profondità nel capitolo 4.
L'idea che sta dietro al procedere in questo modo è che così facendo si "ricicla" quanta più possibile informazione prodotta precedentemente e, soprattutto, si evita di calcolare due riduzioni una multipla dell'altra: ad esempio se $f - g = h$ in un certo grado d, allora nei gradi più alti vale $tf - tg = th$.

Regolarità L'ipotesi di regolarità comporta vari problemi: anzitutto non ci sono algoritmi efficienti[1] per determinare quando una successione è regolare. Se è, a priori, noto che una successione è regolare, allora ne è

[1] dove per efficienti si intende più efficienti dello stesso calcolo di una base di Gröbner.

nota immediatamente la funzione di Hilbert per il lemma 2.1.4, e, quando è nota la funzione di Hilbert di un ideale, ci sono varianti ottimizzate dell'algoritmo di Buchberger, i cosiddetti algoritmi "Hilbert-driven" ([Tra96]).
Inoltre se il numero di polinomi è maggiore al numero di indeterminate dell'anello polinomiale in cui si sta lavorando allora la successione non può essere regolare.

2.7 Esempio

Riportiamo qui un esempio completo dei passi compiuti dall'algoritmo 2.5.1. Poniamo $\mathbb{K} = \mathbb{F}_7 = \mathbb{Z}/7\mathbb{Z}$, $P = \mathbb{K}[x, y, z]$ con ordinamento `DRL` e $f_1 = x^2 + 2xy - 3y^2 - 3xz + yz + 3z^2$, $f_2 = 2x^2 - xy - 3y^2 - 2xz + 3yz - 2z^2$, $f_3 = -2x^2 - 3xy + 3y^2 - 2yz + 3z^2$. Cerchiamo la base di Gröbner di $I = (\mathcal{F}) = (f_1, f_2, f_3)$. Come nel resto del capitolo indichiamo con k il polinomio che stiamo trattando e d il grado in cui stiamo operando.

$\mathbf{k = 1}$

All'inizio sappiamo già che $I_1 = (f_1)$, e quindi con $k = 1$ non c'è nulla da fare.

$\mathbf{k = 2}$

Sostituiamo f_2 con $\mathrm{NF}_{I_1}(f_2) = 2xy + 3y^2 - 3xz + yz - z^2$ e cominciamo con $d = 3$, $L_{2,3} = \{xe_1, ye_1, ze_1, xe_2, ye_2, ze_2\}$. Costruiamo $\overline{\mathcal{M}}_{2,3}$ prendendo multipli di f_1 ed f_2, ed otteniamo una matrice con 6×9:

$\overline{\mathcal{M}}_{2,3} =$

	x^3	x^2y	xy^2	y^3	x^2z	xyz	y^2z	xz^2	yz^2	z^3
ze_1					1	2	4	4	1	3
ye_1		1	2	4		4	1		3	
xe_1	1	2	4		4	1		3		
ze_2						2	3	4	1	6
ye_2			2	3		4	1		6	
xe_2		2	3		4	1		6		

Per calcolare l'eliminazione bastano due combinazioni di righe e si ottiene:

$\widetilde{\mathcal{M}}_{2,3} =$

	x^3	x^2y	xy^2	y^3	x^2z	xyz	y^2z	xz^2	yz^2	z^3
ze_1					1	2	4	4	1	3
ye_1		1	2	4		4	1		3	
xe_1	1	2	4		4	1		3		
ze_2						2	3	4	1	6
ye_2			2	3		4	1		6	
xe_2				**4**	4	2	2	6	4	

Una sola riga risulta differente: quella con etichetta xe_2. Questa dà un termine di testa nuovo, y^3. Aggiungiamo quindi $4y^3+4x^2z+2xyz+2y^2z+6xz^2+4yz^2$ a GB. Con $k=2$, d deve raggiungere $1+d_1-1+d_2-1=3$, quindi con $k=2$ abbiamo finito.

$\mathbf{k=3}$

Sostituiamo f_3 con $\mathrm{NF}_{I_2}(f_3) = -y^2-xz+3yz-z^2$, cominciamo quindi con $d=3$. $L_{3,3} = \{xe_1, ye_1, ze_1, xe_2, ye_2, ze_2, xe_3, ye_3, ze_3\}$ Costruiamo $\overline{\mathcal{M}}_{3,3}$ che è una matrice 9×10:

$\overline{\mathcal{M}}_{3,3} =$

	x^3	x^2y	xy^2	y^3	x^2z	xyz	y^2z	xz^2	yz^2	z^3
ze_1					1	2	4	4	1	3
ye_1		1	2	4		4	1		3	
xe_1	1	2	4		4	1		3		
ze_2						2	3	4	1	6
ye_2			2	3		4	1		6	
xe_2		2	3		4	1		6		
ze_3							6	6	3	6
ye_3				6		6	3		6	
xe_3			6		6	3		6		

L'eliminazione di Gauss produce:

$\widetilde{\mathcal{M}}_{3,3} =$

	x^3	x^2y	xy^2	y^3	x^2z	xyz	y^2z	xz^2	yz^2	z^3
ze_1					1	2	4	4	1	3
ye_1		1	2	4		4	1		3	
xe_1	1	2	4		4	1		3		
ze_2						2	3	4	1	6
ye_2			2	3		4	1		6	
xe_2				4	4	2	2	6	6	
ze_3							6	6	3	6
ye_3								**1**	3	3
xe_3									**2**	5

Qui troviamo due nuovi elementi della base di Gröbner: $xz^2+3yz^2+3z^3$ e $2yz^2-2z^3$.

Infine, l'ultimo grado da esaminare è $d = 4$, qui $L_{3,4} = \{z^2e_1, yze_1, xze_1, y^2e_1, xye_1, x^2e_1, z^2e_2, yze_2, xze_2, y^2e_2, xye_2, z^2e_3, yze_3, xze_3, y^2e_3\}$.
Costruiamo $\overline{\mathcal{M}}_{3,4}$ che è una matrice 15×15 per ragioni di spazio scriviamo solo i polinomi che corrispondono ad ogni etichetta:

$$\overline{\mathcal{M}}_{3,4} = \begin{cases} z^2e_1 & x^2z^2 + 2xyz^2 + 4y^2z^2 + 4xz^3 + yz^3 + 3z^4 \\ yze_1 & x^2yz + 2xy^2z + 4y^3z + 4xyz^2 + y^2z^2 + 3yz^3 \\ xze_1 & x^3z + 2x^2yz + 4xy^2z + 4x^2z^2 + xyz^2 + 3xz^3 \\ y^2e_1 & x^2y^2 + 2xy^3 + 4y^4 + 4xy^2z + y^3z + 3y^2z^2 \\ xye_1 & x^3y + 2x^2y^2 + 4xy^3 + 4x^2yz + xy^2z + 3xyz^2 \\ x^2e_1 & x^4 + 2x^3y + 4x^2y^2 + 4x^3z + x^2yz + 3x^2z^2 \\ z^2e_2 & 2xyz^2 + 3y^2z^2 + 4xz^3 + yz^3 + 6z^4 \\ yze_2 & 2xy^2z + 3y^3z + 4xyz^2 + y^2z^2 + 6yz^3 \\ xze_2 & 4y^3z + 4x^2z^2 + 2xyz^2 + 2y^2z^2 + 6xz^3 + 4yz^3 \\ y^2e_2 & 2xy^3 + 3y^4 + 4xy^2z + y^3z + 6y^2z^2 \\ xye_2 & 4y^4 + 4x^2yz + 2xy^2z + 2y^3z + 6xyz^2 + 4y^2z^2 \\ z^2e_3 & 6y^2z^2 + 6xz^3 + 3yz^3 + 6z^4 \\ yze_3 & xz^3 + 3yz^3 + 3z^4 \\ xze_3 & 2yz^3 + 5z^4 \\ y^2e_3 & xyz^2 + 3y^2z^2 + 3yz^3 \end{cases}$$

Facendo l'eliminazione otteniamo $\widetilde{\mathcal{M}}_{3,4}$:

$$\widetilde{\mathcal{M}}_{3,4} = \begin{cases} z^2e_1 & x^2z^2 + 2xyz^2 + 4y^2z^2 + 4xz^3 + yz^3 + 3z^4 \\ yze_1 & x^2yz + 2xy^2z + 4y^3z + 4xyz^2 + y^2z^2 + 3yz^3 \\ xze_1 & x^3z + 2x^2yz + 4xy^2z + 4x^2z^2 + xyz^2 + 3xz^3 \\ y^2e_1 & x^2y^2 + 2xy^3 + 4y^4 + 4xy^2z + y^3z + 3y^2z^2 \\ xye_1 & x^3y + 2x^2y^2 + 4xy^3 + 4x^2yz + xy^2z + 3xyz^2 \\ x^2e_1 & x^4 + 2x^3y + 4x^2y^2 + 4x^3z + x^2yz + 3x^2z^2 \\ z^2e_2 & 2xyz^2 + 3y^2z^2 + 4xz^3 + yz^3 + 6z^4 \\ yze_2 & 2xy^2z + 3y^3z + 4xyz^2 + y^2z^2 + 6yz^3 \\ xze_2 & 4y^3z + 4x^2z^2 + 2xyz^2 + 2y^2z^2 + 6xz^3 + 4yz^3 \\ y^2e_2 & 2xy^3 + 3y^4 + 4xy^2z + y^3z + 6y^2z^2 \\ xye_2 & 4y^4 + 4x^2yz + 2xy^2z + 2y^3z + 6xyz^2 + 4y^2z^2 \\ z^2e_3 & 6y^2z^2 + 6xz^3 + 3yz^3 + 6z^4 \\ yze_3 & xz^3 + 3yz^3 + 3z^4 \\ xze_3 & 2yz^3 + 5z^4 \\ y^2e_3 & \mathbf{6z^4} \end{cases}$$

Da quest'ultima matrice ricaviamo un altro elemento che andrà nella base di Gröbner: $6z^4$. Con $k = 3$, d deve raggiungere $1+d_1-1+d_2-1+d_3-1 = 4$, quindi con $k = 3$ questo era l'ultimo grado da esaminare.

A questo punto l'algoritmo termina. La base di Gröbner cercata era:

$$\begin{aligned} GB = \quad & \{x^2 + 2xy - 3y^2 - 3xz + yz + 3z^2 \\ & 2xy + 3y^2 - 3xz + yz - z^2 \\ & -3y^3 - 3x^2z + 2xyz + 2y^2z - xz^2 - 3yz^2 \\ & -y^2 - xz + 3yz - z^2 \\ & xz^2 + 3yz^2 + 3z^3 \\ & 2yz^2 - 2z^3 \\ & 6z^4\} \end{aligned}$$

Capitolo 3

Algoritmo F5 generalizzato

In questo capitolo generalizzeremo le idee del precedente e ridurremo le ipotesi necessarie per il funzionamento dell' algoritmo, in particolare l'ipotesi di regolarità e la necessità di procedere incrementalmente negli ideali.
Sia $\mathbb{K}$ un campo, e $P = \mathbb{K}[x_1, \ldots, x_n]$ l'anello dei polinomi in n indeterminate a coefficienti in $\mathbb{K}$ con ordinamento τ. Sia $\mathcal{F} = (f_1, \ldots, f_m) \in P^m$ una successione di m polinomi omogenei di grado rispettivamente $d_1, \ldots, d_m$. Sia $I = (\mathcal{F})$ e P^m il modulo libero di rango m su P, sia inoltre:

$$\begin{array}{rccc} \phi: & P^m & \to & I \\ & (g_1, \ldots, g_m) & \mapsto & \sum_{i=1}^m g_i f_i \end{array}$$

Consideriamo un ordinamento di modulo qualsiasi σ su P^m. D'ora in avanti quando useremo la notazione $\mathrm{LT}(\ldots)$, o i simboli $<, \leq, >, \geq$, li intenderemo rispetto all'ordine τ o σ a seconda che gli argomenti siano nell'anello P o nel modulo P^m.
La relazione $I \cong P^m/\ker(\phi)$ continua a valere, dal momento che non dipende dalla regolarità nè dagli ordinamenti dati su P e P^m.
Ragionando in modo simile a quanto fatto nel capitolo precedente otteniamo il seguente:

Teorema 3.0.1. *Usando le definizioni date sopra di $\mathbb{K}$, P ed $\mathcal{F}$, sia $d \in \mathbb{N}$, sia $\mathcal{S}$ un sottoinsieme di $\mathrm{LT}(\mathrm{Syz}\,\mathcal{F})$. E sia $M_{\mathcal{S}}$ il P sottomodulo di P^m generato da $\mathcal{S}$. Allora valgono le seguenti:*

1. $$(I)_d = \left\{ \sum_{i=1}^m g_i f_i \;\middle|\; g_i \in P,\ \partial g_i + \partial f_i = d,\ \forall t \in \mathrm{Supp}(g_i) : te_i \notin M_{\mathcal{S}} \right\} \tag{3.0.1}$$

2. $$(I)_d = \langle tf_i \mid \deg(tf_i) = d, \quad te_i \notin M_{\mathcal{S}} \rangle_{\mathbb{K}} \tag{3.0.2}$$

Dimostrazione. Dato $g \in P^m$, allora

$$g = \mathrm{NF}_{\mathrm{Syz}\,\mathcal{F}}(g) \iff g = \mathrm{NF}_{\mathrm{LT}(\mathrm{Syz}\,\mathcal{F})}(g) \Rightarrow g = \mathrm{NF}_{M_{\mathcal{S}}}(g)$$

L'essere in forma normale rispetto a $M_{\mathcal{S}}$ è una proprietà più debole rispetto all'essere in forma normale rispetto a $\mathrm{Syz}\,\mathcal{F}$ od a $\mathrm{LT}(\mathrm{Syz}\,\mathcal{F})$.
La tesi è quindi un indebolimento delle relazioni 2.2.4 e 2.2.5.
Detto altrimenti, per rappresentare l'ideale I, ci si può limitare a considerare quei g che sono in forma normale rispetto a $M_{\mathcal{S}}$. □

Useremo, come nel capitolo precedente, la nozione di polinomio etichettato. La definizione è analoga, l'unica differenza è che l'ordine sul P^m è diverso, perciò quando si prende il massimo tra due etichette, si intende il massimo rispetto all'ordinamento del modulo P^m.

3.1 Passo F5 generalizzato

Analizziamo ora cosa succede costruendo matrici di Macaulay ridotte e facendone l'eliminazione di Gauss quando si conosce solo un sottoinsieme dei generatori del modulo dei termini di testa delle sizigie.
Nel seguito di questa sezione assumiamo le seguenti definizioni: $\mathcal{F} = (f_1, \ldots, f_m) \in P^m$ è una successione di polinomi omogenei, $X = \{(l_f, f)\}$ è un insieme (finito) di polinomi etichettati, $I_X \subset I$ è l'ideale generato dai polinomi di X, e L_X l'insieme delle etichette di X, d è un naturale, ed, infine, $L = \{tl_f \mid t \in \mathbb{T}^n,\ l_f \in L_X\}$ è l'insieme delle etichette "costruibili" come multipli di polinomi etichettati di X.

Definizione 3.1.1 (Etichette $\mathcal{S}$-normali)**.** Sia $\mathcal{S}$ un sottoinsieme finito di $\mathrm{LT}(\mathrm{Syz}\,\mathcal{F})$ e sia $M_{\mathcal{S}}$ il P modulo generato da $\mathcal{S}$. Definiamo

$$L^{\mathcal{S}} = \{l \in L \mid l \notin M_{\mathcal{S}}\}$$

insieme delle etichette *$\mathcal{S}$-normali*. Chiameremo nel seguito *$\mathcal{S}$-normali* polinomi etichettati la cui etichetta sia $\mathcal{S}$-normale.

Osservazione 3.1.1. Se $M_{\mathcal{S}} = \mathrm{LT}(\mathrm{Syz}\,\mathcal{F})$ allora l'insieme delle etichette $\mathcal{S}$-normali coincide con l'insieme delle etichette normali definito nel capitolo precedente.

Denoteremo con $L_d^{\mathcal{S}} = \{l \in L^{\mathcal{S}} \mid \deg(l) = d\}$ l'insieme delle etichette $\mathcal{S}$-normali di grado d.

Teorema 3.1.1 (Proprietà del passo F5)**.** *Si consideri una matrice di Macaulay ridotta $\mathcal{M}_d^{\mathcal{S}}$ avente come righe un insieme di polinomi le cui etichette siano $L_d^{\mathcal{S}}$.*

Sia $\widetilde{\mathcal{M}_d^{\mathcal{S}}}$ la matrice ottenuta facendo un'eliminazione di Gauss su $\mathcal{M}_d^{\mathcal{S}}$ come descritto nel capitolo precedente:

$$\widetilde{\mathcal{M}_d^{\mathcal{S}}} = \begin{matrix} L^+ \\ L^0 \end{matrix} \begin{pmatrix} \mathcal{M}^+ \\ 0 \end{pmatrix}$$

Dove $\mathcal{M}^+$ è una matrice con tutte le righe non nulle, e L^+ e L^0 sono gli insiemi delle etichette corrispondenti rispettivamente alle righe non nulle ed alle righe nulle.
Allora valgono le seguenti:

1. *Le righe di $\mathcal{M}^+$, interpretate come polinomi, sono una base come spazio vettoriale di $(I_X)_d$.*
2. *$L^0 \subseteq \mathrm{LT}(\mathrm{Syz}\,\mathcal{F}) \setminus M_{\mathcal{S}}$.*

Dimostrazione. Per il teorema 3.0.1, applicato con $I = I_X$, si ha che i polinomi $\phi(L_d^{\mathcal{S}})$ sono un sistema di generatori per $(I_X)_d$, inoltre ragionando come nel lemma 2.4.3 si vede che un qualsiasi insieme di polinomi che ha come insieme delle etichette $L_d^{\mathcal{S}}$ costituiscono un sistema di generatori per $(I_X)_d$ come $\mathbb{K}$ spazio vettoriale. Pertanto facendo l'eliminazione di Gauss e prendendo i polinomi non nulli che si ottengono, si ha una base, come $\mathbb{K}$ spazio vettoriale, questo prova il primo punto della tesi.
Consideriamo ora le righe nulle della matrice: queste sono polinomi etichettati validi della forma $(\tilde{l}, 0)$, quindi $0 = \phi(g)$, con $g \in P^m$ tale che $\mathrm{LT}(g) = \tilde{l}$, perciò necessariamente vale: $g \in \mathrm{Syz}\,\mathcal{F}$, da cui $\tilde{l} \in \mathrm{LT}(\mathrm{Syz}\,\mathcal{F})$. Ricordando che l'insieme di etichette finali coincide con quello iniziale, per il lemma 2.4.4, $L_d^{\mathcal{S}}$, e che $L_d^{\mathcal{S}} \cap M_{\mathcal{S}} = \emptyset$ si ha $\tilde{l} \notin M_{\mathcal{S}}$, che è il secondo punto della tesi. □

Nel seguito chiameremo *passo F5* la procedura enunciata nel teorema precedente. Questo teorema ci dice che, anzitutto, possiamo applicare il passo F5 anche se non abbiamo una conoscenza completa dei termini di testa delle sizigie, ma, ben più importante, che dal risultato dell'eliminazione possiamo ricavare i termini di testa delle sizigie che prima non erano noti in grado d, e quindi aggiungerli ad $\mathcal{S}$.

Algoritmo 3.1.1 (Passo F5)**.**

Input: X un insieme di polinomi etichettati,

$\mathcal{S}$ un sottoinsieme di $\mathrm{LT}(\mathrm{Syz}\,\mathcal{F})$,

$d \in \mathbb{N}$.

Output: $X' \supset X$ un insieme di polinomi etichettati che contengono una base di Gröbner troncata di I_X al grado d.

$\mathcal{S}' \supset \mathcal{S}$ un insieme di termini di testa delle sizigie di grado d.

1. Calcola $L_d^{\mathcal{S}} = \{l \in L^{\mathcal{S}} \mid \deg(l) = d,\ l \notin M_{\mathcal{S}}\}$.
2. Costruisci un insieme di polinomi etichettati $X_d^{\mathcal{S}}$, multipli di polinomi di X, che abbiano come insieme delle etichette $L_d^{\mathcal{S}}$.
3. Sia $\mathcal{M}_d^{\mathcal{S}}$ la matrice che ha come righe $X_d^{\mathcal{S}}$.
4. Esegui un'eliminazione di Gauss su $\mathcal{M}_d^{\mathcal{S}}$ per ottenere $\widetilde{\mathcal{M}_d^{\mathcal{S}}}$.

$$\widetilde{\mathcal{M}_d^{\mathcal{S}}} = \begin{matrix} L^+ \\ L^0 \end{matrix} \begin{pmatrix} \mathcal{M}^+ \\ 0 \end{pmatrix}$$

5. $X' := X \cup \{$ polinomi corrispondenti alle righe di $\mathcal{M}^+\}$.
6. $\mathcal{S}' := \mathcal{S} \cup L^0$.
7. Return X',$\mathcal{S}'$.

Osservazione 3.1.2 (Numero riduzioni a zero)**.** Se l'insieme $\mathcal{S}$ contiene tutti i generatori dei termini di testa delle sizigie in grado $d-1$, allora il numero di righe nulle dopo l'eliminazione sarà pari al numero di elementi di grado d della base di Gröbner ridotta delle sizigie di $I_k = (f_1, \ldots, f_k)$, dove k è quel numero naturale tale che $I_X = I_k$.

3.1.1 Caso σ = POS + τ

Nel caso in cui l'ordine σ dato sul modulo P^m sia del tipo `PosTo`, con ordinamento τ, si è in grado di conoscere alcuni termini di testa delle sizigie senza che questi vengano "scoperti" dal passo F5.

Lemma 3.1.2. *Sia $\mathcal{F} = (f_1, \ldots, f_m) \in P^m$, sia $1 \leq k \leq m$, e sia $\pi_k : P^m \to P$ definita da $\pi_k(\sum_{i=1}^m h_i e_i) = h_k$ la proiezione sulla k-esima componente. Allora $I_{k-1} = (f_1, \ldots, f_{k-1}) \subseteq \pi_k(\mathrm{Syz}(f_1, \ldots, f_k)) \subseteq \pi_k(\mathrm{Syz}\,\mathcal{F})$. Se inoltre su P^m c'è un ordine del tipo `PosTo`, con term ordering τ allora $\forall k = 1, \ldots, m$ vale $\mathrm{LT}(I_{k-1})e_k \subseteq \mathrm{LT}(\mathrm{Syz}\,\mathcal{F})$.*

Dimostrazione. Usando il lemma 2.1.1 si ha $I_{k-1} = \pi_k(\mathrm{PSyz}(f_1, \ldots, f_k)) \subseteq \pi_k(\mathrm{Syz}(f_1, \ldots, f_k))$. La seconda parte della tesi si ottiene prendendo i termini di testa. □

Lemma 3.1.3 (Numero riduzioni a zero, σ = POS + τ)**.** *Si supponga di inserire nell'insieme $\mathcal{S}$ tutti gli elementi della forma $\mathrm{LT}(I_{k-1})e_k$ nel momento in cui $I_X = I_k$, e di eseguire i passi F5 per i gradi d in ordine crescente: $d = \deg(f_k), \deg(f_k)+1, \ldots$.*
Allora il numero di riduzioni a zero al passo d è pari al numero minimo di elementi che bisogna aggiungere alla base di Gröbner ridotta di $\mathrm{PSyz}(f_1, \ldots, f_k)$ in grado d per ottenere una base di Gröbner di $\mathrm{Syz}(f_1, \ldots, f_k)$.

Corollario 3.1.4. *Se la successione $\mathcal{F}$ è regolare allora il numero di riduzioni a zero è zero.*

Osservazione 3.1.3. Se la successione $\mathcal{F}$ è regolare allora le operazioni compiute da questo algoritmo coincidono con quelle compiute dall'algoritmo del capitolo precedente.

3.2 Algoritmi

3.2.1 Criteri di Arresto

In generale rimuovendo l'ipotesi di regolarità il teorema 2.5.1 non è valido, e quindi non può essere usato per stimare il grado di regolarità dell'ideale su cui si sta lavorando.

Coppie critiche Un modo per risolvere questo problema è di procedere in modo simile a quanto viene fatto nell'algoritmo di Buchberger omogeneo: ogni volta che un polinomio viene aggiunto alla base di Gröbner si generano le coppie critiche, eventualmente usando i criteri di Gebauer e Möller.
Sapendo quali sono le coppie critiche, in particolare conoscendo i loro gradi, si sa che è necessario procedere solo fino al più grande grado in cui c'è almeno una coppia, anche se questo può essere molto alto.

Hilbert-driven F5 Se è nota, si può usare la funzione di Hilbert, da questa è immediato il calcolo del grado di regolarità. Questo sarebbe ottimale, in quanto garantisce di non costruire matrici in gradi inutili. Nella sezione 4.3.2 vengono forniti esempi che illustrano il tempo "perso" in gradi inutili.

3.2.2 Algoritmo incrementale

Quello descritto in questo paragrafo è la più diretta generalizzazione dell'algoritmo di [Bar06].
In generale d'ora in poi *incrementale* sarà usato per indicare algoritmi che procedono calcolando prima basi di Gröbner di I_1, poi di I_2, ...

Algoritmo 3.2.1 (Incrementale)**.**

Input: $\mathcal{F} = (f_1, \ldots, f_m) \in P^m$, f_k omogeneo $\forall k$

Output: Una base di Gröbner di $I = (f_1, \ldots, f_m)$.

1. Etichetta i generatori: $f_i \mapsto (1e_i, f_i)$
2. Metti un ordine $\sigma = POS + \tau$ su P^m.

3. $\mathbb{B} := \emptyset$
4. $GB := \{(1e_1, f_1)\}$
5. $\mathcal{S} := \{e_1 LT(f_1)\}$
6. For $k = 2, \ldots, m$ do
   (a) $GB := GB \cup \{(1e_k, f_k)\}$
   (b) UpdatePairs $(\mathbb{B})$
   (c) $d := \deg(f_k)$
   (d) Finchè ci sono coppie di grado $\geq d$
      i. $GB, \mathcal{S} :=$ PassoF5 $(GB, \mathcal{S}, d)$
      ii. UpdatePairs $(\mathbb{B})$
      iii. $d := d + 1$
   (e) $\mathcal{S} := \mathcal{S} \cup \{e_k \,\mathrm{LT}(f) \mid (te_k, f) \in GB\}$
7. Return $\{f \mid (l_f, f) \in GB\}$

Teorema 3.2.1. *L'algoritmo 3.2.1 calcola una base di Gröbner di I.*

Dimostrazione. La dimostrazione è analoga a quella del teorema 2.5.2. La correttezza del passo F5 è dimostrata nel teorema 3.1.1.
Il fatto che tutte le coppie si riducano a zero garantisce di ottenere tutta la base di Gröbner alla fine dell'esecuzione. □

Nel caso in cui la successione $\mathcal{F}$ è regolare, allora l'algoritmo compie le stesse operazioni dell'algoritmo 2.5.1.

3.2.3 Algoritmo non incrementale

Un altro modo di usare l'algoritmo 3.1.1 è quello di procedere in modo non incrementale, cioè di calcolare direttamente basi di Gröbner di tutto l'ideale I senza passare per le basi di Gröbner degli ideali "intermedi" I_k.

Algoritmo 3.2.2 (Non incrementale)**.**

Input: $\mathcal{F} = (f_1, \ldots, f_m) \in P^m$, f_k omogeneo $\forall k$

Output: Una base di Gröbner di $I = (f_1, \ldots, f_k)$.

1. Etichetta i generatori: $f_i \mapsto (1e_i, f_i)$.
2. Metti un ordine $\sigma = POS + \tau$ su P^m.
3. $\mathbb{B} := \emptyset$.

4. $GB := \{(1e_k, f_k) \mid k = 1, \ldots, m\}$.
5. $\mathcal{S} := \{e_k LT(f_k) \mid k = 1, \ldots, m\}$.
6. UpdatePairs ($\mathbb{B}$).
7. $d := \min\{\deg(f_k) \mid k = 1, \ldots, m\}$.
8. Finchè ci sono coppie di grado $\geq d$.
   (a) $GB, \mathcal{S} :=$ PassoF5 $(GB, \mathcal{S}, d)$.
   (b) UpdatePairs ($\mathbb{B}$).
   (c) $\mathcal{S} := \mathcal{S} \cup \{e_k \,\mathrm{LT}(f) \mid (te_k, f) \in GB\}$.
   (d) $d := d + 1$.
9. Resturn $\{f \mid (l_f, f) \in GB\}$.

Questo algoritmo può facilmente essere modificato per calcolare basi di Gröbner troncate.

3.2.4 Algoritmo con ordini diversi

Come presentato nei paragrafi precedenti non è necessario che l'ordinamento dato su P^m sia in alcun modo legato a quello su P. In particolare è interessante osservare che l'algoritmo può essere usato per produrre informazioni sull'ideale dei termini di testa delle sizigie, è sufficiente che alla fine dell'esecuzione queste informazioni vengano restituite.
Non riporteremo i listati delle procedure che applicano il passo F5 con ordini diversi, in quanto sono identiche a quelli già presenti.

Proposizione 3.2.2. *Supponiamo che l'ordine su P^m sia $\sigma = POS + \eta$, nel caso in cui la successione $\mathcal{F}$ sia regolare allora alla fine dell'esecuzione dell'algoritmo, se l'ultimo grado considerato è stato d, saranno noti tutti i generatori dell'ideale dei termini di testa di $\mathrm{LT}_\eta(f_1, \ldots, f_{m-1})$ di grado minore od uguale a $d - \deg(f_m)$.*

3.3 Anelli quoziente

Se implementato come descritto, l'algoritmo incrementale 3.2.1 spesso costruirà matrici comunque molto grandi e quasi già ridotte a scala. Nell'implementazione che è stata fatta abbiamo notato che la maggior parte del tempo veniva passata a creare la matrice piuttosto che a fare operazioni su di essa.
In questa sezione descriveremo una strategia che permetterà, aumentando la complessità dell'algoritmo, di ridurre ulteriormente il numero di righe della matrice.

Sia $1 < k \leq m$, supponiamo, durante l'esecuzione dell'algoritmo incrementale 3.2.1, di star considerando il polinomio f_k. Sia $d > \deg(f_k)$, consideriamo la matrice di Macaulay ridotta $\mathcal{M}_{k,d}$, questa conterrà alcuni polinomi con etichetta della forma te_k ed altre con etichette te_s per qualche $s < k$. Sia ora la matrice ottenuta facendo un'eliminazione di Gauss, $\widetilde{\mathcal{M}}_{k,d}$, le righe che davvero interessano di questa matrice sono quelle il cui termine di testa non è multiplo di altri termini di testa già noti. Osservando che, se (te_s, f) è un polinomio etichettato valido allora $f \in I_s \setminus I_{s-1}$, e che $\mathrm{LT}(I_{k-1})$ è già noto, deduciamo che le uniche righe che possono dare termini di testa "nuovi" sono quelle corrispondenti ad etichette della forma te_k, per qualche $t \in \mathbb{T}^n$. Abbiamo quindi dimostrato il seguente:

Lemma 3.3.1. *Nelle solite ipotesi, siano $1 < k \leq m$, $d > \deg(f_k)$, sia $\mathcal{M}_{k,d}$ la matrice di Macaulay ridotta e sia $\widetilde{\mathcal{M}}_{k,d}$ la matrice ottenuta facendone un'eliminazione di Gauss. Allora solo le righe con etichetta della forma te_k, per un qualche $t \in \mathbb{T}^n$ possono avere un termine di testa $\mathrm{LT}(f)$ non multiplo di altri termini di testa già noti, cioè $\mathrm{LT}(f) \in I_k \setminus I_{k-1}$.*

Questo risultato suggerisce di cercare di eliminare le righe corrispondenti a etichette della forma $te_{<k}$: basta osservare che queste hanno l'unico effetto di portare in forma normale rispetto ad I_{k-1} le righe con la cui etichetta è della forma te_k.

Proposizione 3.3.2. *Sia (te_k, f) un polinomio etichettato valido, allora*

$$(te_k, \mathrm{NF}_{I_{k-1}}(f))$$

è un polinomio etichettato valido.

Dimostrazione. Basta osservare che $\mathrm{NF}_{I_{k-1}}(f) = f + g$ per un qualche $g \in I_{k-1}$ necessariamente esiste un'etichetta $t'e_{k-1}$ tale che $(t'e_{k-1}, g)$ è un polinomio etichettato valido, da cui

$$(te_k, f) + (t'e_{k-1}, g) = (\max(te_k, t'e_{k-1}), f + g) = (te_k, \mathrm{NF}_{I_{k-1}}(f))$$

che è un polinomio etichettato valido perchè somma di polinomi etichettati validi (lemma 2.3.1). □

In conclusione possiamo costruire una matrice $\mathcal{M}'_{k,d}$ in cui non mettiamo i polinomi le cui etichette sono della forma $te_{<k}$, e mettiamo le forme normali rispetto ad I_{k-1} dei polinomi le cui etichette sono della forma te_k. Questa nuova matrice $\mathcal{M}'_{k,d}$ non rispetta più la definizione di matrice di Macaulay ridotta, tuttavia essa è una matrice di Macaulay ridotta per l'ideale I_k dell'anello quoziente $\mathbb{K}[x_1,\ldots,x_n]/I_{k-1}$.

Osservazione 3.3.1. Questa ulteriore riduzione della dimensione delle matrici ha apparentemente un costo in termini di tempo: quello impiegato nel calcolare tutte le forme normali.
Tuttavia questo è nella quasi totalità dei casi vantaggioso: il tempo speso nel costruire la matrice $\mathcal{M}_{k,d}$ è di gran lunga superiore al tempo necessario per le forme normali; ed inoltre questo abbatte i requisiti di memoria dell'algoritmo, che spesso potevano diventare proibitivi.

Capitolo 4

Implementazione ed esperimenti

Tutti gli algoritmi qui esposti sono stati implementati in C++ usando la libreria CoCoA5.

4.1 Costruzione delle matrici

Una quantità di tempo non indifferente viene passata nel costruire le matrici su cui l'algoritmo opera. Quando si costruisce una matrice $\mathcal{M} = \overline{\mathcal{M}}_{k,d}$ il modo più efficace di procedere è di costruirla a partire dalla matrice $\mathcal{M}' = \overline{\mathcal{M}}_{k,d-1}$.
L'idea base è di costruire ogni riga di $\mathcal{M}$ come multiplo per un'indeterminata di una riga di $\mathcal{M}'$, ovviamente questo lascia della libertà. Ad esempio se xye_1 è un'etichetta valida, questa può essere generata sia come multiplo di un polinomio con etichetta xe_1, sia di uno con etichetta ye_1.
Sperimentalmente questa scelta influenzava in modo significativo il numero di operazioni che l'eliminazione di Gauss deve compiere.
Indichiamo con $\mathcal{S}$ un sottoinsieme di $\mathrm{LT}(\mathrm{Syz}\,\mathcal{F})$.
Descriveremo qui due strategie per costruire la matrice $\mathcal{M}$ a partire dalla matrice $\mathcal{M}'$.

Strategia Bardet-Faugère Per un termine $t \in \mathbb{T}^n$ sia $D(t) = \max\{i : x_i | t\}$, cioè $x_{D(t)}$ è l'indeterminata con indice massimo che divide t. Per ogni riga (te_k, f) di $\mathcal{M}'$, aggiungiamo la riga $x_j(te_k, f)$ a $\mathcal{M}$ per ogni $j = D(t), \ldots, n$ tale che l'etichetta $x_j te_k$ sia $\mathcal{S}$-normale.
Per vedere che in questo modo si generano tutti le possibili etichette $\mathcal{S}$-normali di grado d una sola volta basta osservare che se te_k è una etichetta $\mathcal{S}$-normale di grado d allora questa verrà costruita solo come: $t = \frac{t}{x_{D(t)}} D(t)$.

Strategia alternativa Consideriamo tutti i multipli per un'indeterminata $\mathcal{S}$-normali di ogni riga di $\mathcal{M}'$; così facendo ad ogni possibile etichetta l di grado d corrispondono diversi polinomi, tra questi scegliamo quello con termine di testa minimo f, e aggiungiamo la riga (l, f) a $\mathcal{M}$.
Il chiaro svantaggio di questo approccio è che sono necessari più controlli di $\mathcal{S}$-normalità, cioè di appartenenza ad ideale monomiale; mentre il vantaggio è che la matrice così costruita è, in certo senso, più "vicina" ad essere già ridotta a scala.

4.1.1 Test veloce di $\mathcal{S}$-normalità

Dalla implementazione fatta è risultato che la fase di costruzione della matrice poteva essere molto dispendiosa in termini di tempo di esecuzione, spesso anzi veniva speso più tempo nella costruzione della matrice piuttosto che nell'eliminazione di Gauss. A sua volta gran parte del tempo passato a costruire la matrice era impiegato nello stabilire se una determinata etichetta era $\mathcal{S}$-normale o no.
Ricordiamo brevemente che te_k è $\mathcal{S}$-normale se $te_k \notin M_{\mathcal{S}}$, dove $M_{\mathcal{S}}$ è il modulo generato da $\mathcal{S}$. Se indichiamo con $\mathcal{S}_k$ l'insieme degli elementi di $\mathcal{S}$ della forma $\mathbb{T}^n e_k$, allora il modo usuale per determinare se un elemento appartiene a $M_{\mathcal{S}}$ è quello di usare la definizione:

$$te_k \in M_{\mathcal{S}} \iff \exists t' \in \mathcal{S}_k : t'|t$$

quindi facendo nel caso peggiore $|\mathcal{S}_k|$ test di divisibiltà di termini.
In questo caso particolare è tuttavia possibile fare di meglio:

Lemma 4.1.1. *In $\mathbb{K}[x_1, \ldots, x_n]$ sia $1 \leq j \leq n$, sia $t = \prod_{i=1}^n x_i^{\alpha_i} \in \mathbb{T}^n$, e sia T un sottoinsieme finito di $\mathbb{T}^n$, tale che $t \notin (T)$. Se $x_j t \in (T)$ allora per ogni $t' = \prod_{i=1}^n x_i^{\beta_i} \in T$ tale che $t'|x_j t$ vale $\beta_j = \alpha_j + 1$.*

Dimostrazione. $t'|x_j t$ implica che $\beta_j \leq \alpha_j + 1$, se per assurdo fosse stato $\beta_j < \alpha_j + 1$, allora $t'|t$ da cui $t \in (T)$ contrario all'ipotesi. □

Osservazione 4.1.1. Questo lemma dice semplicemente che se un termine t non appartiene ad un ideale monomiale e se, aumentando l'esponente della j-esima indeterminata, invece vi appartiene, allora tra i generatori dell'ideale ce n'è uno in cui la j-esima indeterminata compare esattamente con quell'esponente e che divide $x_j t$.

L'applicazione di questo lemma è immediata: anzichè tenere $\mathcal{S}_k$ come una semplice collezione di termini, teniamo per ogni indeterminata x_j e per esponente α con cui questa compare in un termine una lista $C_{j,k}^{\alpha}$ di tutti i termini in cui questa compare esattamente con esponente α. In simboli:

$$C_{j,k}^{\alpha} = \left\{ t \in \mathcal{S}_k : x_j^{\alpha} \mid\mid t \right\}$$

Dove la scrittura $x_j^\alpha \mid\mid t$ indica che α è il massimo esponente con cui x_j divide t. Questa struttura ovviamente occupa più spazio in memoria, ma nel complesso questo è trascurabile rispetto all'occupazione di memoria dei polinomi e delle matrici.
Con questa struttura il test di $\mathcal{S}$-normalità per un termine della forma $x_j t e_k$, dove $x_j^\alpha \mid\mid x_j t$, diventa:

$$x_j t e_k \in M_{\mathcal{S}} \iff \exists t' \in C_{j,k}^\alpha : t' | t$$

questo riduce significativamente il numero dei test di divisibilità che devono essere fatti, al massimo sono $|C_{j,k}^\alpha|$, in alcuni casi può anche capitare che $C_{j,k}^\alpha = \emptyset$, nel qual caso il test è istantaneo. In alcuni casi l'impatto di questa semplice idea arriva a ridurre il tempo di esecuzione globale del 20%.

4.2 Riduzione delle matrici

Le matrici di Macaulay ridotte che si incontrano sono sparse e, spesso, anche già quasi ridotte a scala; da un punto di vista implementativo non sembrava che una implementazione di vettori sparsi per rappresentare le righe fosse particolarmente più efficiente della struttura che rappresenta polinomi, che è ottimizzata per polinomi sparsi.
Inoltre le restrizioni imposte sulle operazioni ammissibili per preservare le etichette non avrebbero permesso di usare una comune implementazione dell'algoritmo di Gauss.
Abbiamo quindi reimplementato l'algoritmo, dove anzichè fare combinazioni di righe di una matrice, facciamo combinazioni lineari di polinomi etichettati.
D'ora in avanti, se M è una matrice di polinomi etichettati la tratteremo indifferentemente come matrice o come insieme di polinomi etichettati.

Definizione 4.2.1 (Ordine sulle righe)**.** Sia M un insieme di polinomi etichettati, e siano $(l_1, f_1), (l_2, f_2) \in M$; mettiamo su questi un ordine $\prec$:

$$(l_1, f_1) \prec (l_2, f_2)$$

Definito nel seguente modo:

$$\begin{array}{ll} \text{Se } \mathrm{LT}(f_1) < \mathrm{LT}(f_2) & (l_1, f_1) \prec (l_2, f_2) \\ \text{Se } \mathrm{LT}(f_1) = \mathrm{LT}(f_2) \text{ e } l_1 > l_2 & (l_1, f_1) \prec (l_2, f_2) \end{array}$$

Cioè se i termini di testa sono diversi, allora si usa l'ordine di $\mathbb{T}^n$ sui termini di testa, nel caso questi coincidano, si usa l'ordine sulle etichette invertito.

Code con priorità Abbiamo inoltre fatto uso di una struttura dati chiamata *priority queue* o *coda con priorità*; senza scendere nei dettagli tecnici una priority queue S è una struttura dati che usata per contenere un insieme di elementi X su cui è definito un ordine. Se S è una priority queue, su questa si possono effettuare le seguenti operazioni:

1. $\mathrm{Top}(S)$: restituisce l'elemento massimo di S,
2. $\mathrm{Pop}(S)$: rimuove l'elemento massimo di S,
3. $\mathrm{Push}(S, x)$: inserisce l'elemento x in S.

L'operazione Top richiede tempo costante, mentre le altre due operazioni richiedono un tempo $O(\log n)$, dove n è il numero di elementi contenuti nella priority queue. Per dettagli sull'implementazione ed il funzionamento di questa struttura rimandiamo a [CLR90].

Algoritmo 4.2.1 (Eliminazione di Gauss per matrici sparse)**.**

Input: M: una matrice di polinomi etichettati

Output: M': la riduzione a scala della matrice M, come descritta nel lemma 2.4.4.

1. $M' := M$
2. Costruisci una priority queue S contente (riferimenti a) tutte le righe di M', ordinate con $\prec$.
3. While $S \neq \emptyset$:
   (a) $(l, f) := \mathrm{Top}(S)$
   (b) $\mathrm{Pop}(S)$
   (c) While $S \neq \emptyset$ and $\mathrm{LT}(\mathrm{Top}(S))$[1] $= \mathrm{LT}(f)$
      i. $(l', f') := \mathrm{Top}(S)$
      ii. $\mathrm{Pop}(S)$
      iii. $f' := f' - \frac{\mathrm{LC}(f')}{\mathrm{LC}(f)} f$
      iv. If $f' \neq 0$ then $\mathrm{Push}(S, (l', f'))$
4. Return M'

[1] Con $\mathrm{LT}((l, f))$ si intende $\mathrm{LT}(f)$.

Correttezza dell'algoritmo La priority queue contiene gli elementi su cui è possibile si debbano ancora fare delle operazioni. L'ordine $\prec$ fa in modo che gli elementi con lo stesso termine di testa siano consecutivi e che, a parità di termine di testa, il massimo di questi sia quello con etichetta minima, cioè quello che può essere usato per ridurre tutti gli altri.
All'inizio in S vengono inseriti tutti i polinomi etichettati (righe della matrice).
Procediamo in questo modo: finché ci sono elementi in S, rimuoviamo il massimo, (l, f), da S, poi se esistono altri elementi di S con lo stesso termine di testa, (l', f'), questi soddisfano per le proprietà dell'ordine $l' > l$ e quindi possiamo sommare a f' multipli di f lasciando l'etichetta invariata. Infine, se il nuovo f' non è zero, lo rimettiamo (l', f') in S e procediamo.

4.3 Esperimenti

Da un punto di vista della performance l'algoritmo più efficiente si è rivelato l'algoritmo F5 incrementale 3.2.1, la più diretta generalizzazione di quello di Faugère, anche perchè questo è il solo algoritmo cui sono applicabili le osservazioni della sezione 3.3, che permettono di contenere di molto i tempi di esecuzione. Quindi le statistiche qui presentate si riferiscono solo a quella versione.

4.3.1 Tempi di Esecuzione

Questi sono i risultati di alcuni esperimenti:

Ideale			T_F	T_B
Nome	τ	$\mathbb{K}$		
Ciclico 5	DRL	$\mathbb{Q}$	0.044	0.032
Ciclico 5	Lex	$\mathbb{Q}$	0.136	0.048
Ciclico 6	DRL	$\mathbb{Q}$	0.70	0.35
Ciclico 6	Lex	$\mathbb{Q}$	/	6.16
Ciclico 6	DRL	$\mathbb{Z}/(32003)$	0.368	0.06
Ciclico 7	DRL	$\mathbb{Q}$	163	42.4
Ciclico 7	DRL	$\mathbb{Z}/(32003)$	12.2	5.12
Ciclico 8	DRL	$\mathbb{Z}/(32003)$	/	224
Katsura 7	DRL	$\mathbb{Z}/(32003)$	**0.31**	**0.41**
Katsura 7	DRL	$\mathbb{Q}$	**0.87**	**3.08**
Katsura 9	DRL	$\mathbb{Z}/(32003)$	**10.8**	**39.9**
Katsura 9	DRL	$\mathbb{Q}$	**35.3**	**328**
Denso 5 6 4	DRL	$\mathbb{Z}/(32003)$	2.73	0.27
GraphCol1	DRL	$\mathbb{Z}/(32003)$	27.5	0.044
Kin 1	DRL	$\mathbb{Z}/(32003)$	25.1	0.95
Pavelle 1	DRL	$\mathbb{Z}/(32003)$	0.24	0.076
Pavelle 1	DRL	$\mathbb{Q}$	0.49	0.316

La colonna T_F contiene i tempi, in secondi, richiesti dall'F5 mentre la colonna T_B i tempi di Buchberger.
Su alcuni esempi F5 non è riuscito a terminare perchè è stata esaurita la memoria fisica.
Questi esperimenti sono stati fatti con l'implementazione dell'algoritmo di Buchberger presente nella libreria CoCoA5 su un Athlon 64 3200+ con 1Gb di ram.
Da questa tabella si possono fare le seguenti osservazioni:

- Sull'ordinamento Lex questa implementazione F5 è strettamente inferiore di Buchberger.

- Per l'ordinamento `DRL`, invece, ci sono dei casi (pochi) in cui F5 risulta decisamente superiore: sostanzialmente gli ideali di Katsura.
- In generale in tutti gli esempi in cui la base di Gröbner contiene elementi di grado alto F5 non è efficiente, questo, in generale, comprende tutti i calcoli effettuati con ordinamento `Lex`.
- Nei confronti di Buchberger, F5 è mediamente più efficiente sui razionali, intuitivamente questo è dovuto al fatto che una frazione non trascurabile del tempo viene impiegato a fare operazioni che non coinvolgono i coefficienti dei polinomi: costruzione delle matrici e test di $\mathcal{S}$-normalità.

4.3.2 Esempi in dettaglio

Riportiamo le seguenti statistiche sull'esecuzione, su alcuni degli esempi precedenti, ad ogni passo:

k	indice polinomio
d	grado
$\#\widetilde{\mathcal{M}}_{k,d}$	righe della matrice di Macaulay ridotta di $\mathbb{K}[x_1,\ldots,x_n]/I_{k-1}$
L	Lunghezza massima dei coefficienti (numero di cifre in base 10)
rsteps	riduzioni per portare le righe in forma normale rispetto a I_{k-1}
C_{time}	tempo per generare la matrice
G_{time}	tempo per l'eliminazione di Gauss
R	riduzioni a zero
S	combinazioni per effettuare l'eliminazione di Gauss
T	righe effettivamente coinvolte nella riduzione
New	nuovi elementi inseriti nella base di Gröbner

La granularità dei tempi è limitata dal sistema operativo (nel nostro caso GNU/linux), e i tempi inferiori al centesimo di secondo sono riportati come zero.

Katsura 7 `DRL` **su** $\mathbb{Q}$

k	d	$\#\widetilde{\mathcal{M}}_{k,d}$	L	rsteps	C_{time}	G_{time}	R	S	T	New
2	3	8	4	2	0	0	0	1	2	1
2	4	35	6	0	0	0	0	0	0	0
3	3	8	4	9	0	0	0	2	3	2
3	4	34	7	0	0	0	0	3	4	1
3	5	104	9	0	0	0	0	0	0	0
4	3	8	5	17	0	0	0	2	4	3
4	4	33	9	14	0	0	0	11	7	3
4	5	96	9	0	0	0	0	3	4	1
4	6	225	11	0	0.01	0	0	0	0	0
5	3	8	8	32	0	0	0	1	5	4
5	4	32	13	51	0	0	0	20	11	6
5	5	88	14	15	0	0.01	0	27	13	4
5	6	192	14	0	0	0	0	3	4	1
5	7	360	14	0	0.01	0	0	0	0	0
6	3	8	6	42	0	0	0	3	6	5
6	4	31	10	124	0	0	0	21	16	10
6	5	80	17	135	0	0.01	0	78	26	10
6	6	160	24	23	0	0.01	0	44	17	5
6	7	272	24	0	0	0	0	3	4	1
6	8	416	24	0	0.01	0.01	0	0	0	0
6	9	592	24	0	0.03	0	0	0	0	0
7	3	8	6	63	0	0	0	1	7	6
7	4	30	11	252	0	0	0	30	22	15
7	5	72	37	481	0	0.01	0	129	42	20
7	6	129	37	297	0	0.02	0	228	51	15
7	7	192	38	34	0.01	0.01	0	65	21	6
7	8	256	38	0	0	0.02	0	3	4	1
7	9	320	38	0	0.02	0.01	0	0	0	0
7	10	384	38	0	0.02	0	0	0	0	0

Questo è uno dei casi in cui il guadagno su Buchberger è più significativo, come si nota, in questo caso, non ci sono mai riduzioni a zero, e il numero di righe delle matrici resta decisamente contenuto, il massimo è infatti 592. Un fenomeno curioso che si verifica quasi sempre è che il numero di righe coinvolte nelle operazioni (colonna T) è molto minore del numero di righe della matrice. Inoltre il numero delle operazioni necessarie per ridurre a scala una matrice, fissato il polinomio, ha un andamento interessante:

cresce fino ad un certo punto, per circa metà dei gradi da considerare, poi decresce.

Katsura 9 `DRL` **su** $\mathbb{Z}/(32003)$

k	d	$\#\widetilde{\mathcal{M}}_{k,d}$	rsteps	C_{time}	G_{time}	R	S	T	New
2	3	10	2	0	0	0	1	2	1
2	4	54	0	0	0	0	0	0	0
3	3	10	9	0	0	0	2	3	2
3	4	53	0	0	0	0	3	4	1
3	5	200	0	0	0	0	0	0	0
4	3	10	17	0	0	0	2	4	3
4	4	52	14	0	0	0	11	7	3
4	5	190	0	0	0	0	3	4	1
4	6	553	0	0.01	0	0	0	0	0
5	3	10	32	0	0	0	1	5	4
5	4	51	51	0	0	0	20	11	6
5	5	180	15	0	0.01	0	27	13	4
5	6	501	0	0.01	0	0	3	4	1
5	7	1182	0	0.04	0.01	0	0	0	0
6	3	10	42	0	0	0	3	6	5
6	4	50	124	0	0	0	21	16	10
6	5	170	135	0.01	0	0	78	26	10
6	6	450	23	0.01	0.01	0	44	17	5
6	7	1002	0	0.04	0.01	0	3	4	1
6	8	1970	0	0.13	0.03	0	0	0	0
6	9	3530	0	0.24	0.04	0	0	0	0
7	3	10	63	0	0	0	1	7	6
7	4	49	252	0	0.01	0	30	22	15
7	5	160	481	0.01	0.01	0	129	42	20
7	6	400	297	0.02	0.03	0	228	51	15
7	7	832	34	0.05	0.03	0	65	21	6
7	8	1520	0	0.11	0.03	0	3	4	1
7	9	2528	0	0.17	0.06	0	0	0	0
7	10	3920	0	0.29	0.1	0	0	0	0
8	3	10	77	0	0	0	4	8	7
8	4	48	417	0	0.01	0	39	29	21
8	5	150	1234	0	0.06	0	185	63	35
8	6	351	1618	0.02	0.12	0	541	97	35
8	7	672	619	0.06	0.1	0	508	79	21
8	8	1120	47	0.1	0.06	0	90	25	7
8	9	1696	0	0.16	0.09	0	3	4	1
8	10	2400	0	0.26	0.11	0	0	0	0

8	11	3232	0	0.44	0.16	0	0	0	0
8	12	4192	0	0.54	0.21	0	0	0	0
9	3	10	104	0	0.01	0	1	9	8
9	4	47	695	0	0.03	0	50	37	28
9	5	140	2702	0.01	0.15	0	294	92	56
9	6	303	5250	0.02	0.4	0	942	157	70
9	7	522	4556	0.06	0.5	0	1814	191	56
9	8	769	1158	0.1	0.29	0	1001	112	28
9	9	1024	62	0.15	0.12	0	119	29	8
9	10	1280	0	0.18	0.12	0	3	4	1
9	11	1536	0	0.22	0.16	0	0	0	0
9	12	1792	0	0.27	0.17	0	0	0	0
9	13	2048	0	0.32	0.2	0	0	0	0

In questo caso si ripropongono gli stessi fenomeni evidenziati nel caso precedente. È da osservare che, su questo esempio, l'algoritmo procede fino a gradi in cui non ci sono elementi nuovi della base di Gröbner, ma che spesso capita che le matrici costruite siano già ridotte a scala. Tutti i tempi riportati nell colonne C_{time} e G_{time} sono in un certo senso "inutili", sommandoli tutti si ottiene un guadagno di circa 4 secondi, per nulla trascurabili se si considera che il tempo totale è di 35 secondi.

Ciclico 6 `DRL` **su** $\mathbb{Q}$

k	d	$\#\widetilde{\mathcal{M}}_{k,d}$	L	rsteps	C_{time}	G_{time}	R	S	T	New
2	4	6	2	3	0	0	0	0	1	1
2	5	20	2	0	0	0	0	0	0	0
3	5	6	2	6	0	0	0	3	3	2
3	6	20	2	21	0	0	0	13	10	3
3	7	49	2	8	0	0	0	10	8	3
3	8	99	3	4	0	0.01	0	7	8	2
3	9	176	5	22	0	0.01	0	9	10	1
3	10	286	5	0	0	0	0	0	0	0
4	6	6	2	21	0	0	0	4	5	4
4	7	20	3	103	0	0.01	0	31	12	8
4	8	49	6	250	0	0.01	1	59	22	5
4	9	94	11	196	0.01	0.02	2	88	32	4
4	10	155	12	320	0	0.03	5	111	46	0
4	11	222	12	0	0.01	0	0	0	0	0
4	12	308	12	0	0.01	0.01	0	0	0	0
4	13	408	12	0	0.04	0	0	0	0	0

5	7	6	2	29	0	0	0	3	4	4
5	8	20	5	284	0	0.02	0	72	14	11
5	9	49	13	460	0	0.06	0	322	29	16
5	10	98	22	103	0.01	0.03	0	240	38	13
5	11	169	24	3	0.01	0.01	0	8	9	1
5	12	260	24	0	0.01	0.01	0	16	17	1
5	13	366	24	43	0.02	0.01	0	111	37	4
5	14	482	29	111	0.01	0.04	0	312	45	9
5	15	606	32	34	0.02	0.03	0	236	36	11
5	16	740	32	1	0.04	0.01	0	16	17	0
5	17	884	32	0	0.04	0.02	0	0	0	0
5	18	1038	32	0	0.04	0.01	0	0	0	0
5	19	1202	32	0	0.05	0.02	0	0	0	0

Capitolo 5

Conclusioni

In questo lavoro abbiamo fornito un solido background teorico per dimostrare il funzionamento dell'algoritmo F5 di Faugère, in particolare ne è stata dimostrata la correttezza in modo originale e completo. Inoltre è stata rimossa l'ipotesi di regolarità sull'input che restringeva il dominio di applicabilità dello stesso. Grazie al risultato fondamentale costituito dal teorema 3.1.1 è stato possibile produrre algoritmi con proprietà diverse dall'F5 originale: l'*F5 non incrementale* e l'*F5 con ordini diversi*.

5.1 Sviluppi futuri

L'implementazione dell'algoritmo F5 incrementale in CoCoA è nella maggior parte dei casi più lenta di quella di Buchberger, in alcuni casi ha tempi di esecuzione comparabili ed in pochi casi è decisamente più veloce. Tuttavia ci sono molti fenomeni che si verificano durante l'esecuzione che non sono compresi a fondo, e che offrono la possibilità di ulteriori ottimizzazioni:

- Quando questa implementazione risulta più inefficiente nei confronti di un normale Buchberger, capita spesso che le matrici diventino subito molto grandi, tuttavia, il numero di righe che vengono considerate per ottenere una matrice a scala è molto contenuto: è quindi possibile costruire matrici ancora più piccole, in modo che nello spazio vettoriale generato dalle loro righe si trovino tutti gli elementi di una base di Gröbner?

  Ad esempio, se si immerge l'ideale in un anello con più indeterminate, il numero delle righe delle matrici esplode, anche se questo non è un problema algoritmicamente più complesso.

- In tutti gli esperimenti, se, per un polinomio fissato, ad un certo punto si genera una matrice già ridotta a scala, allora tutte le successive

lo sono. In questo caso non si producono più elementi nuovi della base di Gröbner. Questo fenomeno si verifica sempre? In caso affermativo, si possono eliminare molti passaggi costosi sia in termini di tempo che di memoria.

- Durante l'esecuzione di F5, la maggior parte del tempo viene persa nei gradi più alti, dove le matrici hanno dimensioni maggiori e dove, in proporzione, ci sono meno elementi della base di Gröbner. Potrebbe, in questo caso, essere conveniente procedere fino ad un certo grado con F5, e poi per gli ultimi usare l'algoritmo di Buchberger; ma qual è una strategia effettiva per scegliere a che grado passare dall'uno all'altro?

- F5 produce una descrizione più ricca di un ideale, rispetto ad una semplice base di Gröbner, con queste informazioni addizionali, qual è il modo più efficiente di calcolare una forma normale?

- Usando le matrici di Macaulay di un ideale I_k di un anello quoziente P/I_{k-1}, è necessario tenere molti polinomi in forma normale rispetto a I_{k-1}. È possibile farlo in modo più efficiente di quanto finora implementato, visto che le uniche operazioni che si fanno su di essi sono somme e prodotti per indeterminate?

- Nell'algoritmo incrementale abbiamo sempre ordinato i polinomi per grado, tuttavia, non c'è nessun risultato che dica che questa sia la strategia migliore. Cambiando questo ordine l'algoritmo fa operazioni completamente diverse. Qual è l'ordine migliore in cui considerare i polinomi?

Bibliografia

[Bar06] M. Bardet, *Étude des systèmes algébriques surdéterminés. applications aux codes correcteurs et à la cryptographie*, Ph.D. thesis, LIP6, 2006.

[BCP97] W. Bosma, J. J. Cannon, e C. Playoust, *The Magma Algebra System I: The User Language*, Journal of symbolic computation **24** (1997), no. 3-4, 235–266.

[Buc65] B. Buchberger, *Ein Algorithmus zum Auffinden der Basiselemente des Restklassenringes nach einem nulldimensionalen Polynomideal*, Ph.D. thesis, 1965.

[Buc79] ———, *A criterion for detecting unnecessary reductions in the construction of groebner bases*, EUROSAM '79: Proceedings of the International Symposiumon on Symbolic and Algebraic Computation (London, UK), Springer-Verlag, 1979, pp. 3–21.

[CLR90] T. H. Cormen, C. E. Leiserson, e R. L. Rivest, *Introduction to Algorithms*, MIT Press, Cambridge MA, 1990.

[CoC] CoCoATeam, CoCoA*: a system for doing Computations in Commutative Algebra*, Available at `http://cocoa.dima.unige.it`.

[CT98] M. Caboara e C. Traverso, *Efficient algorithms for ideal operations*, ACM Press, 1998, Extended Abstract, pp. 147–152.

[Eis95] D. Eisenbud, *Commutative algebra: with a view toward algebraic geometry*, Springer Verlag, New York, 1995.

[Fau99] J. C. Faugère, *A new efficient algorithm for computing gröbner bases (f_4)*, Journal of Pure and Applied Algebra **139** (1999), no. 1-3, 61–88.

[Fau02] ———, *A new efficient algorithm for computing gröbner bases without reduction to zero ($f5$)*, ISSAC '02: Proceedings

of the 2002 international symposium on Symbolic and algebraic computation (New York, NY, USA), ACM Press, 2002, pp. 75–83.

[GM88] R. Gebauer e H. M. Möller, *On an installation of buchberger's algorithm*, J. Symb. Comput. **6** (1988), no. 2-3, 275–286.

[KR00] M. Kreuzer e L. Robbiano, *Computational commutative algebra. 1*, Springer-Verlag, Berlin, 2000. MR MR1790326 (2001j:13027)

[KR05] _____, *Computational commutative algebra. 2*, Springer-Verlag, Berlin, 2005. MR MR2159476 (2006h:13036)

[Lan65] S. Lang, *Algebra*, Addison-Wesley, 1965.

[Laz83] D. Lazard, *Gröbner-bases, gaussian elimination and resolution of systems of algebraic equations.*, EUROCAL (J. A. van Hulzen, ed.), Lecture Notes in Computer Science, vol. 162, Springer, 1983, pp. 146–156.

[MGH+05] M. B. Monagan, K. O. Geddes, K. M. Heal, G. Labahn, S. M. Vorkoetter, J. McCarron, e P. DeMarco, *Maple 10 programming guide*, Maplesoft, Waterloo ON, Canada, 2005.

[MMT92] H. M. Möller, T. Mora, e C. Traverso, *Gröbner bases computation using syzygies*, ISSAC '92: Papers from the international symposium on Symbolic and algebraic computation (New York, NY, USA), ACM Press, 1992, pp. 320–328.

[Tra96] C. Traverso, *Hilbert functions and the Buchberger algorithm*, Journal of Symbolic Computation **22** (1996), no. 4, 355–376 (English).