Superfluous S-polynomials in Strategy-Independent Gröbner Bases

Grant Olney Passmore and Leonardo de Moura

***Abstract*—Using the machinery of proof orders originally introduced by Bachmair and Dershowitz in the context of canonical equational proofs, we give an abstract, strategy-independent presentation of Gröbner basis procedures and prove the correctness of two classical criteria for recognising superfluous S-polynomials, Buchberger's criteria 1 and 2, w.r.t. arbitrary fair and correct basis construction strategies. To do so, we develop a general method for proving the strategy-independent correctness of superfluous S-polynomial criteria which seems to be quite powerful. We also derive a new superfluous S-polynomial criterion which is a generalization of Buchberger-1 for Gröbner basis procedures implementing a special form of eager simplification and is proved to be correct strategy-independently.**

I. Introduction

Buchberger's algorithm for constructing Gröbner bases of polynomial ideals is one of the central methods in computer algebra [4]. It constructs a canonical simplifier for ideals of polynomial rings over a field, and hence provides a basis for many problems in polynomial ideal theory. Buchberger's algorithm is very similar to completion procedures such as Knuth-Bendix [11]. The similarity was first observed in [12] and fully developed in [2].

The a priori recognition and discarding of superfluous critical pairs is an important component in modern Gröbner basis procedures. Such recognition is usually accomplished by a so-called *superfluous S-polynomial* or *reduction to zero* criterion which is a computationally efficient sufficient condition for recognizing S-polynomials that would reduce to zero with respect to the rewrite system being constructed. Gröbner basis procedures such as Buchberger's algorithm and its enhancements F4 [9] and F5 [10] prescribe a fixed execution strategy for the construction of S-polynomials, their reduction and simplification, and the subsequent extension of the current partial Gröbner basis until completion. Moreover, the proofs of correctness for the admissibility of reduction to zero criteria for such procedures are usually tied to the execution strategy of the algorithm for which they were introduced. For example, Buchberger's Criterion 1 [5] is introduced in the context of a fixed basis construction strategy (the classical Buchberger's algorithm), and the original proof of correctness of the criterion uses an inductive cut-point argument that makes explicit use of this strategy.

The idea of generalizing the essential features of a Gröbner basis procedure into a strategy-independent setting can perhaps be most immediately traced to the work of Bachmair and Dershowitz on canonical equational proofs [1]. In this work, the authors observe that different completion algorithms (such as Knuth-Bendix, Huet, ordered, etc.) can be seen to be merely particular *strategies* for organising a collection of primitive "abstract completion" inference rules. These inference rules crystallize operations common to all completion procedures they considered. By separating the primitive completion operations from the choice of strategy guiding their application, the authors are able to prove many important results whose justifications were once tied to a particular completion algorithm (i.e., strategy) in a strategy-independent way. By doing so, such results can then be carried over to other completion procedures for free. Other related and influential work includes that of Bachmair and Tiwari on a strategy-independent presentation of procedures for computing D-bases of polynomial ideals [3], and that of Winkler on the elimination of superfluous critical pairs from completion procedures in which the strategy of keeping all rules interreduced is used [14].

We wish to have an abstract framework for reasoning about Gröbner basis procedures with respect to a multitude of possible execution strategies. This goal began with a very practical motivation. In our work on using Gröbner basis calculations as part of an automated theorem proving system [8], we have experimented with computing Gröbner bases through a number of different simplification and reduction strategies. These strategies originate from the automated deduction community and include the so-called "Otter" and "Discount" loops used by modern superposition theorem provers [13]. Basing a Gröbner basis procedure on such a strategy can result in a procedure that behaves very differently than Buchberger's algorithm or F4 or F5, and we struggled with the fact that a number of the superfluous S-polynomials criteria we wished to exploit were not easily seen to be admissible in such a setting. We then learned of the Bachmair-Dershowitz work on abstract completion and proceeded to adapt it to solve our problem.

In this article, we develop a strategy-independent description of correct Gröbner basis procedures called *abstract Gröbner bases*, and then examine a number of classical superfluous S-polynomial criteria in this general setting. These classical criteria are the so-called Buchberger-1 and Buchberger-2, and a generalization of Buchberger-1 that we believe is novel. We then show how the technique of proof orders can be used to prove the correctness of all of these reduction to zero criteria, strategy-independently, using a uniform method. The key idea is to (i) define a formal notion of "proof" for abstract Gröbner basis procedures, (ii) define a well-ordering upon these proofs, and (iii) reduce the strategy-independent admissibility of reduction to zero criteria to the existence of "smaller" proofs

LFCS, University of Edinburgh, 10 Crichton Street, Edinburgh EH8 9AB, Scotland, UK. email: g.passmore@ed.ac.uk

Microsoft Research, One Microsoft Way, Redmond, WA, 98074, USA. email: leonardo@microsoft.com

in the absence of S-polynomials deemed superfluous by the criteria under investigation.

II. Foundations

In the sequel, let p_i denote polynomials in $\mathbb{Q}[\vec{x}] = \mathbb{Q}[x_1, \ldots, x_n]$. Given $\{p_1, \ldots, p_k\}$, a finite subset of $\mathbb{Q}[\vec{x}]$, the polynomial ideal $\mathcal{I}(\{p_1, \ldots, p_k\})$ is the set of polynomials $\{\sum_{i=1}^{k} p_i q_i \mid q_i \in \mathbb{Q}[\vec{x}]\}$. An element $x_1^{i_1} \ldots x_n^{i_n}$ in $\mathbb{Q}[x_1, \ldots, x_n]$ is called a *power-product* (or *term*), and an element $cx_1^{i_1} \ldots x_n^{i_n}$ with $c \in \mathbb{Q}$ and $x_1^{i_1} \ldots x_n^{i_n}$ a power-product is called a *monomial*. We say a monomial is *monic* if $c = 1$. (This terminology is not universally agreed upon.) We use $\mathbb{M}$ to denote the set of all power-products in $\mathbb{Q}[x_1, \ldots, x_n]$. From hereafter, we use p, q, r, s and t to denote polynomials, m to denote power-products and monic monomials, c to denote coefficients, and cm to denote monomials. We say a power-product $x_1^{i_1} \ldots x_n^{i_n}$ contains x_k if $i_k > 0$. Given two power-products $m_1 = x_1^{i_1} \ldots x_n^{i_n}$ and $m_2 = x_1^{j_1} \ldots x_n^{j_n}$, $m_1 m_2$ denotes the power-product $x_1^{i_1+j_1} \ldots x_n^{i_n+j_n}$, if $i_k \geq j_k$ for $k \in \{1, \ldots, n\}$, then $\frac{m_1}{m_2}$ denotes the power-product $x_1^{i_1-j_1} \ldots x_n^{i_n-j_n}$, and the *least common multiple* $\mathsf{lcm}(m_1, m_2)$ of m_1 and m_2 is the power product $x_1^{max(i_1,j_1)} \ldots x_n^{max(i_n,j_n)}$. We say a polynomial p *contains the power-product* m if p contains the monomial cm for some coefficient $c \neq 0$. Given a polynomial $p = c_1 m_1 + \ldots + c_n m_n$ and a monomial cm, we use cmp to denote the polynomial $(c_1 c) m_1 m + \ldots + (c_n c) m_n m$. Similarly, given a polynomal $p = c_1 m_1 + \ldots + c_n m_n$ and a polynomal q, we use pq to denote the polynomal $c_1 m_1 q + \ldots + c_n m_n q$. In the work that follows, all polynomials are assumed to be in a sum-of-monomials normal form (e.g., a polynomial will never contain two distinct monomials formed from the same power-product).

Given two monic monomials p_1 and p_2 of the form $\underline{m_1} + q_1$ and $\underline{m_2} + q_2$, let $\tau_{1,2}$ be the $\mathsf{lcm}(m_1, m_2)$, then we use $\mathsf{spol}(p_1, p_2)$ to denote the polynomial

$$(\frac{\tau_{1,2}}{m_1}) q_1 - (\frac{\tau_{1,2}}{m_2}) q_2.$$

Given a set of polynomials S, it is easy to see that if $\{p_1, p_2\} \subseteq \mathcal{I}(S)$, then $\mathsf{spol}(p_1, p_2) \in \mathcal{I}(S)$.

An order relation $\prec$ on the set $\mathbb{M}$ is *admissible* if $m_1 \prec m_2$ implies that $m_1 m \prec m_2 m$, for all m_1, m_2 and m in $\mathbb{M}$. A *monomial order* is a total order on $\mathbb{M}$ which is admissible and a well ordering. Given two polynomials p_1 and p_2, we say $p_1 \prec p_2$ if there is a monomial cm in p_2 such that for all monomials $c_i m_i$ in p_1, $m_i \prec m$.

We first recall Buchberger's algorithm and observe that it is but one of many possible strategies for computing Gröbner bases. Then, we introduce *abstract Gröbner bases* and formalize notions of *fairness* and *correctness* for basis construction strategies.

A. Buchberger's Algorithm and Strategy

Let us examine Buchberger's algorithm (Fig. 1) and reflect upon the basis construction strategy underlying it. But what is a strategy? Perhaps the best way to approach this question is to examine *what might be changed* in the algorithm while still preserving its correctness. Two absolutely crucial ideas underlying the algorithm which seem to be a requirement of all Gröbner basis procedures are (i) the use of polynomials as rewrite rules, and (ii) the iterative recovery of confluence (that is, *completion*) of the rewrite system induced by the polynomials through the computation of critical pairs (S-polynomials).

Input: $\langle F = \{p_1, \ldots, p_k\} \subset \mathbb{Q}[\vec{x}], \prec \rangle$
Output: G s.t. G is a GBasis of F w.r.t. $\prec$
G := F; S := $\{\langle p_i, p_j \rangle \mid 1 \leq i < j \leq k\}$
while $S \neq \emptyset$ **do**
  Let $\langle p_i, p_j \rangle \in S$
  For some q s.t. *S-polynomial*$(p_i, p_j) \xrightarrow{G} q$
  if $q \neq 0$ **then**
    $S := S \cup \{\langle p, q \rangle \mid p \in G\}$
    $G := G \cup \{q\}$
  end if
  $S := S \setminus \{\langle p_i, p_j \rangle\}$
end while

Fig. 1. Buchberger's Algorithm

If, for the sake of motivation, we assume that these are the *only* two requirements of a Gröbner basis procedure, then it is easy to see much that might be changed. For instance, one might allow members of G to simplify other members of G. Or one might simplify multiple S-polynomials simultaneously, as done in F4. Or one might allow specially selected members of $G \setminus \{p_i, p_j\}$ to simplify the individual components of pairs $\langle p_i, p_j \rangle \in S$ just before considering $\mathsf{spol}(p_i, p_j)$. Or one might use $\mathsf{spol}(p_i, p_j)$ to simplify members of G before using members of G to compute a normal form for $\mathsf{spol}(p_i, p_j)$. When one attempts to construct Gröbner basis procedures using different strategies such as these, it can become difficult to (i) prove the correctness of the resulting procedure, and (ii) prove that desirable optimizations developed in the context of well-studied procedures, such as a reduction to zero criteria known to be admissible in Buchberger's algorithm, are in fact admissible under the strategy being used in the new procedure. This is especially true of reduction to zero criteria that have temporal requirements (e.g., by requiring that certain S-polynomials were "processed" *before* others). We introduce *abstract Gröbner bases* to address precisely these problems.

B. Abstract Gröbner Bases

Given a monomial order $\prec$, the key idea in Buchberger's algorithm is to use a polynomial $cm + q$, where $q_i \prec m$, as a rewrite rule $cm \rightarrow -q$. For clarity, we will write polynomials used as rewrite rules in a form in which the head monomial has been underlined. For instance, when using $\underline{cm} + q$ as a rewrite rule we will mean $cm \rightarrow -q$. We say a polynomial used as a rewrite rule $\underline{cm} + q$ is *monic* if $c = 1$. To simplify the presentation that follows, we will assume all polynomials used as rewrite rules are monic. The monic

$$\textsf{Orient} \quad \frac{S \cup \{\underline{cm} + q\}, G}{S, G \cup \{\underline{m} + (\frac{1}{c})q\}}$$

$$\textsf{Superpose} \quad \frac{S, G \cup \{p_1, p_2\}}{S \cup \{\mathsf{spol}(p_1, p_2)\}, G \cup \{p_1, p_2\}}$$

$$\textsf{Delete} \quad \frac{S \cup \{0\},\ G}{S, G}$$

$$\textsf{Simplify-S} \quad \frac{S \cup \{c_1 m_1 m_2 + q_1\}, G \cup \{\underline{m_2} + q_2\}}{S \cup \{q_1 - c_1 m_1 q_2\}, G \cup \{\underline{m_2} + q_2\}}$$

$$\textsf{Simplify-H} \quad \frac{S, G \cup \{\underline{m_1 m_2} + q_1, \underline{m_2} + q_2\}}{S \cup \{q_1 - m_1 q_2\}, G \cup \{\underline{m_2} + q_2\}} \quad \textbf{if } m_1 \neq 1$$

$$\textsf{Simplify-T} \quad \frac{S, G \cup \{\underline{m} + c_1 m_1 m_2 + q_1, \underline{m_2} + q_2\}}{S, G \cup \{\underline{m} - c_1 m_1 q_2 + q_1, \underline{m_2} + q_2\}}$$

Fig. 2. Inference rules.

polynomial $p = \underline{m} + q$ induces a *reduction relation* $\mapsto_p$ on polynomials. It is defined as $q_1 + c_1 m_1 m \mapsto_p q_1 - c_1 m_1 q$ for arbitrary monomials $c_1 m_1$ and polynomials q_1. Given a set of monic polynomials $G = \{p_1, \ldots, p_k\}$, the reduction relation induced by G is defined as: $\mapsto_G = \bigcup_{i=1}^{k} \mapsto_{p_i}$.

Definition 1 (Gröbner bases) A finite set of monic polynomials G is a Gröbner basis of the ideal $\mathcal{I}(F)$ iff $\mathcal{I}(G) = \mathcal{I}(F)$ and $\mapsto_G$ is confluent.

The inference rules in Figure 2 work on pairs of sets of polynomials (S, G). In all rules, the coefficients c and c_1 are assumed to be non-zero. We use $(S_1, G_1) \vdash (S_2, G_2)$ to indicate that (S_1, G_1) can be transformed to (S_2, G_2) by applying one of the inference rules in Figure 2.

Theorem 1: $(S_1, G_1) \vdash (S_2, G_2)$ implies $\mathcal{I}(S_1 \cup G_1)) = \mathcal{I}(S_2 \cup G_2))$.

Proof: Easy by observing (i) every rule that extends (S_1, G_1) does so by adding polynomials already in $\mathcal{I}(S_1 \cup G_1)$, (ii) reducing a polynomial p using q when p and q are in (S_1, G_1) does not change $\mathcal{I}(S_1 \cup G_1)$, and (iii) a polynomial p is removed from $(S_1 \cup G_1)$ only when $p = 0$. ■

Definition 2 (Procedure) A Gröbner basis procedure $\mathfrak{G}$ is a program that accepts a set of polynomials $\{p_1, \ldots, p_k\}$, a monomial order $\prec$, and uses the rules in Figure 2 to generate a (finite or infinite) sequence $(S_1 = \{p_1, \ldots, p_k\},\ G_1 = \emptyset) \vdash (S_2, G_2) \vdash (S_3, G_3) \vdash \ldots$. This sequence is called a *run* of $\mathfrak{G}$.

Given a set of monic polynomials G, the set of S-polynomials $\mathsf{SP}(G)$ is defined as the set

$$\{\mathsf{spol}(p_1, p_2) \mid p_1, p_2 \in G\}.$$

Definition 3 (Correct Procedure) A Gröbner basis procedure $\mathfrak{G}$ is said to be correct iff it produces only finite runs $(S_1,\ G_1 = \emptyset) \vdash \ldots \vdash (S_n = \emptyset,\ G_n)$, and

$$\mathsf{SP}(G_n) \subseteq (S_1 \cup S_2 \cup \ldots \cup S_{n-1}).$$

Theorem 2: Let $\mathfrak{G}$ be a correct Gröbner basis procedure, then for any run $(S_1,\ G_1 = \emptyset) \vdash \ldots \vdash (S_n = \emptyset,\ G_n)$, G_n is a Gröbner basis for $\mathcal{I}(S_1)$.

The proof of Theorem 2, which follows from Theorem 6 below, uses a technique called *proof orders*. We will study this in detail in the next section.

Definition 4 (Eager S-simplification) Given a Gröbner basis procedure $\mathfrak{G}$, we say $\mathfrak{G}$ implements eager S-simplification iff $\mathfrak{G}$ only applies Orient to $p \in S_i$ when Simplify-S cannot be applied to p.

Proposition 3: Given a Gröbner basis procedure $\mathfrak{G}$ using eager S-simplification, then for any run $(S_1, G_1) \vdash (S_2, G_2) \vdash \ldots$, for all $j \geq 1$, there is no $\underline{m_1} + q_1$ and $\underline{m_2} + q_2$ in G_j such that $m_1 = m_2$ and $q_1 \neq q_2$. Moreover, in this case, the condition $m_1 \neq 1$ in the rule Simplify-H is only restricting self simplifications.

Definition 5 (Fairness) A Gröbner basis procedure $\mathfrak{G}$ is said to be fair iff for any run $(S_1, G_1) \vdash (S_2, G_2) \vdash \ldots$

$$\mathsf{SP}(\bigcup_{i \geq 1} \bigcap_{j \geq i} G_j) \subseteq \bigcup_{i \geq 1} S_i.$$

Theorem 4: If a Gröbner basis procedure $\mathfrak{G}$ implements eager S-simplification, is fair, and Superpose is applied at most once for any pair of polynomials in $\bigcup_{i \geq 1} G_i$, then $\mathfrak{G}$ is correct.

Proof: We just need to show that every run of $\mathfrak{G}$ is finite. This follows from Dickson's lemma, and the fact that any infinite run will contain an infinite number of Superpose steps. ■

Example 1: Let F be the set of polynomials:

$$\{x^2 y - 1,\ x y^2 - y\}.$$

Then, using the inference rules in Figure 2, we can generate the run in Figure 3. A reduced Gröbner basis for F is contained in the final state $(\emptyset,\ \{y - 1,\ x - 1\})$.

As an exercise in gaining familiarity with the inference rules, we illustrate how they can be used to simulate Buchberger's algorithm in Figure 4.

III. Proof Orders

In the following, we assume that

$$(F = S_1, G_1 = \emptyset) \vdash \ldots \vdash (S_n = \emptyset, G_n)$$

is an arbitrary run of a correct Gröbner basis procedure $\mathfrak{G}$. We use S_* to denote the set $S_1 \cup \ldots \cup S_n$ and G_* to denote the set $G_1 \cup \ldots \cup G_n$.

An *equational step* in (S_*, G_*) is a tuple $\langle s, p, cm, t \rangle$, where s, p and t are polynomials, cm is a monomial, $p \in S_* \cup G_*$, and $t = s - cmp$. We use

$$s \xleftrightarrow{\langle p,\, cm \rangle} t$$

to denote the equational step $\langle s, p, cm, t \rangle$.

Proposition 5: Let $\langle s, p, cm, t \rangle$ be an equational step, then for any monomial $c'm'$ in p, s or t contains the power-product $m'm$.

$\{x^2y-1,\ xy^2-y\},\ \emptyset$
$\vdash$ Orient: x^2y-1
$\{xy^2-y\},\ \{\underline{x^2y}-1\}$
$\vdash$ Orient: xy^2-y
$\emptyset,\ \{\underline{x^2y}-1,\ \underline{xy^2}-y\}$
$\vdash$ Superpose: $\mathsf{spol}(\underline{x^2y}-1,\ \underline{xy^2}-y) = xy-y$
$\{xy-y\},\ \{\underline{x^2y}-1,\ \underline{xy^2}-y\}$
$\vdash$ Orient: $xy-y$
$\emptyset,\ \{\underline{x^2y}-1,\ \underline{xy^2}-y,\ \underline{xy}-y\}$
$\vdash$ Simplify-H: $\underline{xy}-y$ over $\underline{x^2y}-1$
$\{xy-1\},\ \{\underline{xy^2}-y,\ \underline{xy}-y\}$
$\vdash$ Simplify-S: $\underline{xy}-y$ over $xy-1$
$\{y-1\},\ \{\underline{xy^2}-y,\ \underline{xy}-y\}$
$\vdash$ Orient: $y-1$
$\emptyset,\ \{\underline{xy^2}-y,\ \underline{xy}-y,\ \underline{y}-1\}$
$\vdash$ Simplify-H: $\underline{y}-1$ over $\underline{xy^2}-y$
$\{xy-y\},\ \{\underline{xy}-y,\ \underline{y}-1\}$
$\vdash$ Simplify-S: $\underline{xy}-y$ over $xy-y$
$\{0\},\ \{\underline{xy}-y,\ \underline{y}-1\}$
$\vdash$ Delete
$\emptyset,\ \{\underline{xy}-y,\ \underline{y}-1\}$
$\vdash$ Simplify-H: $\underline{y}-1$ over $\underline{xy}-y$
$\{x-y\},\ \{\underline{y}-1\}$
$\vdash$ Simplify-S: $\underline{y}-1$ over $x-y$
$\{x-1\},\ \{\underline{y}-1\}$
$\vdash$ Orient: $x-1$
$\emptyset,\ \{\underline{y}-1,\ \underline{x}-1\}$
$\vdash$ Superpose: $\mathsf{spol}(\underline{y}-1,\underline{x}-1) = x-y$
$\{x-y\},\ \{\underline{y}-1,\ \underline{x}-1\}$
$\vdash$ Simplify-S: $\underline{y}-1$ over $x-y$
$\{x-1\},\ \{\underline{y}-1,\ \underline{x}-1\}$
$\vdash$ Simplify-S: $\underline{x}-1$ over $x-1$
$\{0\},\ \{\underline{y}-1,\ \underline{x}-1\}$
$\vdash$ Delete:
$\emptyset,\ \{\underline{y}-1,\ \underline{x}-1\}$

Fig. 3. A *run* for $\{x^2y-1,\ xy^2-y\}$ w.r.t. DegLex with $x \prec y$.

Input: $\langle S = \{p_1,\ldots,p_k\} \subset \mathbb{Q}[\vec{x}], \prec\rangle$
Output: G s.t. G is a GBasis of S w.r.t. $\prec$
Apply Orient to every member of S
Apply Superpose between every $p_i, p_j \in G$ $(p_i \neq p_j)$
while $S \neq \emptyset$ **do**
  Choose $\mathsf{spol}(p_i,p_j) \in S$
  Apply Simplify-S to $\mathsf{spol}(p_i,p_j) \in S$ as long as possible
  Call the resulting simplified polynomial (in S) q
  if $q \neq 0$ **then**
    Apply Orient to q
    Apply Superpose to all pairs $\langle p,q\rangle$ $(p \neq q \in G)$
      for which Superpose has not been previously
      applied
  else
    Apply Delete to q
  end if
end while

Fig. 4. Rule-based Simulation of Buchberger's Algorithm

A *right rewrite step* in (S_*, G_*) is a tuple $\langle s,p,m,t\rangle$, where s, p and t are polynomials, m is a monic monomial, $p \in G_*$. Let s be of the form $c_s m m_p + q_s$ and p be of the form $\underline{m_p} + q_p$, then $t = s - c_s m p = q_s - c_s m q_p$. Intuitively, p is a polynomial being used as a rewrite rule, and m specifies that the monomial $c_s m m_p$ of s will be "rewritten" to $-c_s m q_p$. We use

$$s \xrightarrow{\langle p,m\rangle} t$$

to denote the right rewrite step $\langle s,p,m,t\rangle$.

Similarly, a *left rewrite step* in (S_*, G_*) is a tuple $\langle t,p,m,s\rangle$, where s, p, t and m are defined as in the right rewrite step case. We use

$$t \xleftarrow{\langle p,m\rangle} s$$

to denote the left rewrite step $\langle s,p,m,t\rangle$. A *rewrite step* is a left or right rewrite step. For every rewrite step, we say s is the *source* and t is the *target*. Note that $t \prec s$.

A *proof step* is an equational step or a rewrite step. We use $s \simeq_F t$ to denote that $s \in \mathcal{I}(F)$ iff $t \in \mathcal{I}(F)$. Recall that $\mathcal{I}(F) = \mathcal{I}(S_* \cup G_*)$, hence for all proof steps $p \in \mathcal{I}(F)$, and $s \simeq_F t$.

A *proof* Pr for $p \simeq_F q$ in (S_*, G_*) is a sequence of proof steps

$$\langle s_1, p_1, c_1 m_1, t_1\rangle \ldots \langle s_k, p_k, c_k m_k, t_k\rangle$$

such that, $s_1 = p$, $t_k = q$, $t_i = s_{i+1}$ for $i \in \{1,\ldots,k-1\}$. We use *lhs*(Pr) to denote s_1 and *rhs*(Pr) to denote t_k.

For example, let F be the set $\{xy-y, x^2y-1\}$. Hence, for any run, $xy-y \in S_0$. Now, assume $\underline{x^2y}-1 \in G_*$. Then,

$$y \xleftrightarrow{\langle xy-y,\,-x\rangle} y + x^2y - xy \xleftrightarrow{\langle xy-y,\,-1\rangle} x^2y \xrightarrow{\langle x^2y-1,\,1\rangle} 1$$

is a proof for $y \simeq_F 1$.

A *rewrite proof* Pr is a proof containing k rewrite steps such that p_i is in G_n for $i \in \{1,\ldots,k\}$, and there is a $j \in \{0,\ldots,k\}$, where the first j steps are right rewrite steps, and the others are left rewrite steps.

For example, assume G_n contains the polynomials $\{\underline{x}+1,\ \underline{y}+z,\ \underline{w^2}-1\}$. Then, the following proof is a rewriting proof for $xy+2 \simeq_F w^2z+2$.

$$xy+2 \xrightarrow{\langle \underline{x}+1,\, y\rangle} -y+2 \xrightarrow{\langle \underline{y}+z,\, 1\rangle} z+2 \xleftarrow{\langle \underline{w^2}-1,\, z\rangle} w^2z+2$$

We say two proofs Pr_1 and Pr_2 in (S_*, G_*) are *equivalent* if *lhs*(Pr_1) = *lhs*(Pr_2) and *rhs*(Pr_1) = *rhs*(Pr_2).

The *cost* of a proof step is a pair where the first component is a multi-set of polynomials and the other a polynomial, and is defined as:

1. For $s \xleftrightarrow{\langle p,cm\rangle} t$, the cost is $(\{s,t\},0)$.
2. For $s \xrightarrow{\langle p,m\rangle} t$ and $t \xleftarrow{\langle p,m\rangle} s$, the cost is $(\{s\},p)$.

Two different cost pairs are compared using the lexicographic product order $\ll$ of $(\prec_M, \prec)$, where $\prec_M$ is the multi-set extension of the order $\prec$ on polynomials. Proof steps are compared by comparing their costs. The overall cost of a proof Pr is the multi-set of the costs of all its proof steps, and two different multi-sets of costs are compared using the multi-set extension $\ll_M$ of $\ll$. Finally, proofs are compared by comparing their costs, and we use $\mathrm{Pr}' \sqsubset \mathrm{Pr}$ to denote that proof Pr' is smaller than proof Pr.

Lemma 1: The order $\sqsubset$ is well-founded.

Proof: This is an immediate consequence of the following facts: the order $\prec$ is well-founded, the multi-set extension of a well-founded order is well-founded, and the lexicographic product order of well-founded orders is well-founded. ■

Lemma 2: Let Pr be a proof in (S_*, G_*) that is not a rewrite proof. Then, there exists a proof Pr' in (S_*, G_*) such that Pr' is equivalent to Pr and $\mathrm{Pr}' \sqsubset \mathrm{Pr}$.

Proof: If Pr is not a rewrite proof, then there are three possible reasons:

1. Pr contains an equational step.
2. Pr contains a rewrite step $\langle s_i, p_i, m_i, s_{i+1}\rangle$, and p_i is not in G_n.
3. Pr contains a *peak* of the form

$$t_1 \xleftarrow{\langle p_1, m_1\rangle} s \xrightarrow{\langle p_2, m_2\rangle} t_2$$

for p_1 and p_2 in G_n.

In the following, we consider each of these three cases separately.

1. Assume Pr contains an equational step

$$s \xleftrightarrow{\langle p, cm\rangle} t$$

By definition of equational step, $t = s - (cm)p$. First, assume $p \in S_*$, then since $S_n = \emptyset$, p is removed from some $S_{j<n}$ using Orient, Delete or Simplify-S. The case where $p \in G_*$ is similar to the case where p is removed from some $S_{j<n}$ using Orient.

(a) Assume Orient was used to remove p. Let p be of the form $\underline{c_p m_p} + q_p$, then $p' = (\frac{1}{c_p})p$ is in G_{j+1}. By Proposition 5, s or t must contain the power-product $m_p m$. First, let us assume that s contains $c_s m_p m$ and t does not. Then, $c_s = c_p c$ because t does not contain the power-product $m_p m$, and by simple algebraic manipulation:

$$t = s - (cm)p = s - (\frac{c_s}{c_p}m)p = s - (c_s m)((\frac{1}{c_p})p) = s - (c_s m)p'.$$

Let Pr' be the proof that is obtained by replacing the equational step with:

$$s \xrightarrow{\langle p', m\rangle} t$$

Similarly, if t contains the power-product $m_p m$ and s does not, we replace the the equational step with the rewrite step:

$$s \xleftarrow{\langle p', m\rangle} t$$

Finally, if both of them contain the power-product $m_p m$, let c_t be the coefficient of $m_p m$ in t. Then, by the definition of equational step, $c_t = c_s - c_p c$. Let s' be the polynomial $s - (c_s m)p'$. By algebraic manipulation, we have:

$$\begin{aligned} s' &= s - (c_s m)p' = s - ((c_p c + c_t)m)p' \\ &= s - (cm)(c_p p') - (c_t m)p' = s - (cm)p - (c_t m)p' \\ &= t - (c_t m)p'. \end{aligned}$$

In this case, let Pr' be the proof that is obtained by replacing the equational step with:

$$s \xrightarrow{\langle p', m\rangle} s' \xleftarrow{\langle p', m\rangle} t$$

In all three cases, the rewrite steps are smaller than the equational step, because $\{s\} \prec_M \{s,t\}$ and $\{t\} \prec_M \{s,t\}$. This shows that the new proof $\mathrm{Pr}' \sqsubset \mathrm{Pr}$.

Before we consider the next case, note that the case where $p \in G_*$ can be handled as above. The only difference is that $p' = p$ when $p \in G_*$.

(b) Assume that Delete was used to remove p, then $p = 0$ and $s = t$, and the equational step can be removed from the proof. Therefore, $\mathrm{Pr}' \sqsubset \mathrm{Pr}$.

(c) Assume p is of the form $c_p m_p m_r + q_p$ and Simplify-S was applied to p using a polynomial $r \in G_j$ of the form $\underline{m_r} + q_r$. Let p' be $-c_p m_p q_r + q_p$, then p' is in S_{j+1}. By Proposition 5, s or t must contain the power-product $m_p m_r m$. Let us assume both of them contain $m_p m_r m$, and c_s and c_t are the coefficients of $m_p m_r m$ in s and t respectively. Recall that c_t must be $c_s - c_p c$. Now, let s' be the polynomial $s-(c_s m_p m)r$ and t' be the polynomial $t-(c_t m_p m)r$. Note that $s' \prec s$ and $t' \prec t$. By simple algebraic manipulation we can show that $t' = s' - (cm)p'$. Now, let Pr' be the proof that is obtained by replacing the equational step with:

$$s \xrightarrow{\langle r, m_p m\rangle} s' \xleftrightarrow{\langle p', cm\rangle} t' \xleftarrow{\langle r, m_p m\rangle} t$$

All three new proof steps are smaller than the original equational step because $\{s\} \prec_M \{s,t\}$, $\{t\} \prec_M \{s,t\}$, and $\{s',t'\} \prec_M \{s,t\}$. This shows that the new proof $\mathrm{Pr}' \sqsubset \mathrm{Pr}$. If s does not contain the power-product $m_p m_r m$, then the first rewrite step is not needed. Similarly, if t does not contain the power-product $m_p m_r m$ the last rewrite step is not needed.

2. Assume Pr contains a rewrite step $\langle s, p, m, t\rangle$, and p is not in G_n. Without loss of generality, assume it is a right rewrite step

$$s \xrightarrow{\langle p, m\rangle} t$$

Since p is not in G_n, it was removed from some $G_{j<n}$ using Simplify-H or Simplify-T and a polynomial $r \in G_j$ of the form $\underline{m_r} + q_r$.

(a) Assume Simplify-H was applied to p using r, and p is of the form $\underline{m_p m_r} + q_p$. Note that $m_p \neq 1$ because of the side condition of Simplify-H, therefore $r \prec p$. Let p' be the polynomial $-m_p q_r + q_p$, then p' is in

S_{j+1}. Since $\langle s,p,m,t\rangle$ is a right rewrite step, s must contain the monomial $c_s m_p m_r m$. By the definition of right rewrite rule, $t = s-(c_s m)p$. Now, let s' be the polynomial $s-(c_s m_p m)r$. Thus, by algebraic manipulation, we can show that $t = s' - (cm)p'$. Let Pr$'$ be the proof that is obtained by replacing the rewrite step with:

$$s \xrightarrow{\langle r,\, m_p m\rangle} s' \xleftrightarrow{\langle p',\, cm\rangle} t$$

The new equational step is smaller than the original step because $s' \prec s$ and $t \prec s$, and consequently $\{s',t\} \prec_M \{s\}$. The cost of the original rewrite step is $(\{s\},p)$. The cost of the new rewrite step is $(\{s\},r)$, and is smaller than $(\{s\},p)$ because $r \prec p$.

(b) Assume Simplify-T was applied to p using r, and p is of the form $\underline{m'_p} + c_p m_p m_r + q_p$. Let p' be the polynomial $\underline{m'_p} - c_p m_p q_r + q_p$, then p' is in G_{j+1}. This case is similar to the case 1c for Simplify-S. Let s' and t' be polynomials defined as in case 1c. Now, let Pr$'$ be the proof that is obtained by replacing the rewrite step with:

$$s \xrightarrow{\langle r,\, m_p m\rangle} s' \xrightarrow{\langle p',\, cm\rangle} t' \xleftarrow{\langle r,\, m_p m\rangle} t$$

The cost of the original rewrite rule is $(\{s\},p)$, and the costs of the new rewrite rules are $(\{s\},r)$, $(\{s'\},p')$ and $(\{t\},r)$. They are smaller than $(\{s\},p)$ because $r \prec p$, $s' \prec s$ and $t \prec s$. If s does not contain the power-product $m_p m_r m$, then the first rewrite step is not needed. In this case $s' = s$, and the cost $(\{s'\},p')$ is smaller than $(\{s\},p)$ because $p' \prec p$. Similarly, if t does not contain the power-product $m_p m_r m$ the last rewrite step is not needed.

3. Assume Pr contains a peak of the form

$$t_1 \xleftarrow{\langle p_1,\, m'_1\rangle} s \xrightarrow{\langle p_2,\, m'_2\rangle} t_2$$

for p_1 and p_2 in G_n. Assume p_1 and p_2 are of the form $\underline{m_1} + q_1$ and $\underline{m_2} + q_2$ respectively. Now, we consider two cases: $m'_1 m_1 \neq m'_2 m_2$ and $m'_1 m_1 = m'_2 m_2$.

(a) Assume $m'_1 m_1 \neq m'_2 m_2$, then s must be of the form $q_s + c_1 m'_1 m_1 + c_2 m'_2 m_2$. Moreover, we must have

$$t_1 = q_s - c_1 m'_1 q_1 + c_2 m'_2 m_2$$
$$t_2 = q_s + c_1 m'_1 m_1 - c_2 m'_2 q_2$$

Let s' be the polynomial $q_s - c_1 m'_1 q_1 - c_2 m'_2 q_2$. Let Pr$'$ be the proof that is obtained by replacing the peak with:

$$t_1 \xleftrightarrow{\langle p_2,\, c_2 m'_2\rangle} s' \xleftrightarrow{\langle p_1,\, c_1 m'_1\rangle} t_2$$

The polynomials t_1, t_2 and s' are smaller than s, hence $\{t_1,s'\} \prec_M \{s\}$, and $\{s',t_2\} \prec_M \{s\}$. Therefore both equational steps are smaller than the rewrite steps in the peak.

(b) Assume $m'_1 m_1 = m'_2 m_2$, then s must be of the form $q_s + cm\tau_{1,2}$ where $\tau_{1,2} = \mathsf{lcm}(m_1,m_2)$. Then, we must have

$$t_1 = q_s - cm(\frac{\tau_{1,2}}{m_1})q_1$$
$$t_2 = q_s - cm(\frac{\tau_{1,2}}{m_2})q_2$$

Moreover, $\mathsf{spol}(p_1,p_2) = \frac{\tau_{1,2}}{m_1}q_1 - \frac{\tau_{1,2}}{m_2}q_2$ must be in S_*. Let Pr$'$ be the proof that is obtained by replacing the peak with:

$$t_1 \xleftrightarrow{\langle \mathsf{spol}(p_1,p_2),\, -cm\rangle} t_2$$

Since $\{t_1,t_2\} \prec_M \{s\}$, the new equational step is smaller than the rewrite steps in the peak. ■

Lemma 3: Every proof Pr in (S_*,G_*) is equivalent to a rewrite proof.

Proof: By well-founded induction on the well-founded order $\sqsubset$. Let Pr be a proof in (S_*,G_*). If Pr is itself a rewrite proof, then we are done. Otherwise, by Lemma 2, there is a proof Pr$'$ such that Pr$' \sqsubset$ Pr. By induction, Pr$'$, and thus also Pr, is equivalent to a rewrite proof. ■

Given a polynomial q of the form $c_1 m_1 + c_2 m_2 + \ldots + c_k m_k$, we use

$$s \overset{\langle p,\, q\rangle}{\twoheadleftarrow\!\!\twoheadrightarrow} t$$

to denote a *multi-equational step*, that is, the sequence of equational steps:

$$s \xleftrightarrow{\langle p,\, c_1 m_1\rangle} s_1 \xleftrightarrow{\langle p,\, c_2 m_2\rangle} s_2 \ldots s_{k-1} \xleftrightarrow{\langle p,\, c_k m_k\rangle} t$$

It is easy to see that $t = s - pq$.

Theorem 6: Given a set of polynomials $F = \{p_1,\ldots,p_k\}$, an arbitrary run

$$(F = S_1, G_1 = \emptyset) \vdash \ldots \vdash (S_n = \emptyset, G_n)$$

of a correct Gröbner basis procedure $\mathfrak{G}$, and a polynomial p, the following holds: If $p \in \mathcal{I}(F)$, then there exists a rewrite proof for $p \simeq_F 0$ using $\mapsto_{G_n}$. Moreover, G_n is confluent.

Proof: If $p \in \mathcal{I}(F)$, then we must have $p = p_1 q_1 + \ldots + p_k q_k$ for some $q_1, \ldots, q_k \in \mathbb{Q}[\vec{x}]$. Let Pr be the following proof in (S_*,G_*) for $p \simeq_F 0$

$$p \overset{\langle p_1,\, q_1\rangle}{\twoheadleftarrow\!\!\twoheadrightarrow} \ldots \overset{\langle p_k,\, q_k\rangle}{\twoheadleftarrow\!\!\twoheadrightarrow} 0$$

By Lemma 3, Pr is equivalent to a rewrite proof.

Now, we show that G_n is confluent. Suppose not. Let $\mapsto_{G_n}$ be the reduction relation induced by G_n. Since G_n is not confluent, there are polynomials s, t_1 and t_2 such that

$$s \mapsto_{G_n} \ldots \mapsto_{G_n} t_1$$
$$s \mapsto_{G_n} \ldots \mapsto_{G_n} t_2$$

where t_1 and t_2 cannot be reduced by G_n. The reductions above induce a proof Pr in (S_*,G_*) for $t_1 \simeq_F t_2$. Actually, this proof only uses polynomials in G_n, but it has a peak at s. By Lemma 3, there is an equivalent rewrite proof Pr$'$, contradicting the assumption that t_1 and t_2 cannot be reduced by G_n. ■

IV. Criteria for Discarding S-polynomials

Buchberger introduced two criteria for discarding superfluous S-polynomials [6]. We now examine how these classical criteria can be accommodated in the general setting of abstract Gröbner bases. Inspecting the proof of Lemma 2, we see that S-polynomials are only used in case 3b, where a non-rewrite proof Pr contains a *peak*. This observation suggests a methodology for proving the strategy-independent admissibility of criteria for discarding redundant S-polynomials.

Observation 1: An S-polynomial $\mathsf{spol}(p_1,p_2)$ can be discarded if it is not needed to obtain a smaller proof Pr' in case 3b of Lemma 2.

In the following, we assume p_1, p_2 and p_k are polynomials in G_* of the form $\underline{m_1}+q_1$, $\underline{m_2}+q_2$ and $\underline{m_k}+q_k$ respectively.

Criterion 1: If $\mathsf{lcm}(m_1,m_2) = m_1m_2$, then $\mathsf{spol}(p_1,p_2)$ is superfluous.

Criterion 2: If there exists some $p_k \in G_*$ s.t. $\mathsf{lcm}(m_1,m_2)$ is a multiple of m_k and $\mathsf{spol}(p_1,p_k)$ and $\mathsf{spol}(p_2,p_k)$ are in S_*, then $\mathsf{spol}(p_1,p_2)$ is superfluous.

Proposition 7: If $\mathsf{lcm}(m_1,m_2) = mm_k$, then

$$\mathsf{lcm}(m_1,m_2) = (m_{k_1})\mathsf{lcm}(m_1,m_k)$$
$$\mathsf{lcm}(m_1,m_2) = (m_{k_2})\mathsf{lcm}(m_2,m_k)$$

for some m_{k_1} and m_{k_2}. Actually,

$$m_{k_1} = \frac{\mathsf{lcm}(m_1,m_2)}{\mathsf{lcm}(m_1,m_k)}$$
$$m_{k_2} = \frac{\mathsf{lcm}(m_1,m_2)}{\mathsf{lcm}(m_2,m_k)}$$

Note that m_{k_1} and m_{k_2} are well defined monomials because $\mathsf{lcm}(m_1,m_2) = \mathsf{lcm}(m_1,m_2,m_k)$.

We first adjust our notion of a correct procedure to take into account the fact that the $\mathsf{Superpose}$ rule may be enhanced to carry a side-condition, φ, barring its application.

Definition 6 (Conditionally Correct Procedure) A Gröbner basis procedure $\mathfrak{G}$ is said to be conditionally φ-correct iff it produces only finite runs $(S_1,\ G_1=\emptyset) \vdash \ldots \vdash (S_n=\emptyset,\ G_n)$, and

$$\mathsf{SP}_\varphi(G_n) \subseteq (S_1 \cup S_2 \cup \ldots \cup S_{n-1}),$$

where $\mathsf{SP}_\varphi(G_n) = \{spol(p_1,p_2) \mid p_1,p_2 \in G_n \wedge \neg\varphi(p_1,p_2)\}$.

Theorem 8: Let φ_1,φ_2 be the natural side-conditions barring applications of $\mathsf{Superpose}$ corresponding to Criteria 1 and 2 respectively. Let $\mathfrak{G}$ be a Gröbner basis procedure that is conditionally $(\varphi_1 \vee \varphi_2)$-correct. Then, Lemma 2 still holds for $\mathfrak{G}$.

Proof: Inspecting the proof of Lemma 2, it is easy to see that case 3b is the only one affected by the restricted $\mathsf{Superpose}$ rule. That is, Pr has a peak of the form:

$$t_1 \xleftarrow{\langle p_1,\, m_1' \rangle} s \xrightarrow{\langle p_2,\, m_2' \rangle} t_2$$

for p_1 and p_2 in G_n, p_1 and p_2 are of the form $\underline{m_1}+q_1$ and $\underline{m_2}+q_2$ respectively, and $m_1'm_1 = m_2'm_2$. Then, s must be of the form $q_s + cm\tau_{1,2}$, where $\tau_{1,2} = \mathsf{lcm}(m_1,m_2)$. Moreover, we must have:

$$t_1 = q_s - cm\frac{\tau_{1,2}}{m_1}q_1$$
$$t_2 = q_s - cm\frac{\tau_{1,2}}{m_2}q_2$$

Now, assume $\mathsf{spol}(p_1,p_2)$ is not in S_* because one of the criteria above was used.

1. Assume $\mathsf{spol}(p_1,p_2)$ is not in S_* because of Criterion 1. Then, $\tau_{1,2} = m_1m_2$, and consequently

$$s = q_s + cmm_1m_2$$
$$t_1 = q_s - cmm_2q_1$$
$$t_2 = q_s - cmm_1q_2$$

   Now, let s' be the polynomial $q_s + (cm)q_1q_2$, and Pr' be the proof that is obtained by replacing the peak with:

$$t_1 \overset{\langle p_2,\, -cmq_1 \rangle}{\twoheadleftarrow\!\!\twoheadrightarrow} s' \overset{\langle p_1,\, -cmq_2 \rangle}{\twoheadleftarrow\!\!\twoheadrightarrow} t_2$$

   Since, t_1, t_2, s' and every intermediate polynomial in the multi-equational steps above is smaller than s, the new equational steps in Pr' are smaller than the two rewrite rules in the peak in Pr. Therefore, $\mathrm{Pr}' \sqsubset \mathrm{Pr}$.
2. Assume $\mathsf{spol}(p_1,p_2)$ is not in S_* because of Criterion 2. Then, there is a p_k of the form $\underline{m_k}+q_k$ in G_* such that $\mathsf{spol}(p_1,p_k)$ and $\mathsf{spol}(p_2,p_k)$ are in S_*, and $\tau_{1,2} = m'm_k$ for some m'. Let $\tau_{1,k} = \mathsf{lcm}(m_1,m_k)$ and $\tau_{2,k} = \mathsf{lcm}(m_2,m_k)$. Then, by Proposition 7, we have $\tau_{1,2} = m_{k_1}\tau_{1,k}$ and $\tau_{1,2} = m_{k_2}\tau_{2,k}$.

$$\begin{aligned} t_1 &= q_s - cm\frac{\tau_{1,2}}{m_1}q_1 \\ &= q_s - cm\frac{m_{k_1}\tau_{1,k}}{m_1}q_1 \\ &= q_s - cmm_{k_1}\frac{\tau_{1,k}}{m_1}q_1 \end{aligned}$$

   Similarly, $t_2 = q_s - cmm_{k_2}\frac{\tau_{2,k}}{m_2}q_2$. Recall that,

$$\mathsf{spol}(p_1,p_k) = (\frac{\tau_{1,k}}{m_1})q_1 - (\frac{\tau_{1,k}}{m_k})q_k$$
$$\mathsf{spol}(p_2,p_k) = (\frac{\tau_{2,k}}{m_2})q_2 - (\frac{\tau_{2,k}}{m_k})q_k$$

   Now, let s' be the polynomial $q_s - cm\frac{\tau_{1,2}}{m_k}q_k$. By algebraic manipulation, we have:

$$\begin{aligned} t_1 + cmm_{k_1}\mathsf{spol}(p_1,p_k) &= q_s - cmm_{k_1}\frac{\tau_{1,k}}{m_k}q_k \\ &= q_s - cm\frac{m_{k_1}\tau_{1,k}}{m_k}q_k \\ &= q_s - cm\frac{\tau_{1,2}}{m_k}q_k = s' \\ &= q_s - cm\frac{m_{k_2}\tau_{2,k}}{m_k}q_k \\ &= q_s - cmm_{k_2}\frac{\tau_{2,k}}{m_k}q_k \\ &= t_2 + cmm_{k_2}\mathsf{spol}(p_2,p_k) \end{aligned}$$

Note that in the equations above, all "fractions" of the form $\frac{m_i}{m_j}$ are actual monomials because in all cases m_j divides m_i. For instance, $\frac{\tau_{1,k}}{m_k}$ is a monomial because m_k always divides $\mathsf{lcm}(m_1, m_k) = \tau_{1,k}$. Now, let Pr′ be the proof that is obtained by replacing the peak in Pr with:

$$t_1 \xleftrightarrow{\langle \mathsf{spol}(p_1,p_k),\, -cmm_{k_1} \rangle} s' \xleftrightarrow{\langle \mathsf{spol}(p_2,p_k),\, -cmm_{k_2} \rangle} t_2$$

Since t_1, t_2 and s' are smaller than s, we have $\mathrm{Pr}' \sqsubset \mathrm{Pr}$. ■

Definition 7 (Eager SH-simplification) We say a Gröbner basis procedure $\mathfrak{G}$ implements eager SH-simplification iff $\mathfrak{G}$ only applies Orient to $p \in S_i$ when Simplify-S cannot be applied to p, and $\mathfrak{G}$ only attempts[1] to apply Superpose to $p_1, p_2 \in G_i$ when Simplify-H cannot be applied to p_1, p_2.

Criterion 3: Assume p_1 and p_2 are polynomials in G_* of the form $\underline{m_1} + q_1$, $\underline{m_2} + q_2$ respectively. If m_1 divides m_2 or m_2 divides m_1, then $\mathsf{spol}(p_1, p_2)$ is superfluous[2].

Theorem 9: Let φ be the natural side-condition for Superpose corresponding to Criteria 3. Let $\mathfrak{G}$ be a conditionally φ-correct Gröbner basis procedure using eager SH-simplification. Let $\mathfrak{G}$ have the property that it has attempted to apply Superpose to every $p_1, p_2 \in G_n$. Then, Lemma 2 still holds.

Proof: As in the proof of Theorem 8, we only need to consider case 3b. That is, Pr has a peak of the form:

$$t_1 \xleftarrow{\langle p_1, m_1' \rangle} s \xrightarrow{\langle p_2, m_2' \rangle} t_2$$

for p_1 and p_2 in G_n, and p_1 and p_2 are of the form $\underline{m_1} + q_1$ and $\underline{m_2} + q_2$. Now, assume $\mathsf{spol}(p_1, p_2)$ is not in S_* because of Criterion 3, then m_1 divides m_2 or m_2 divides m_1. Since $\mathfrak{G}$ uses eager SH-simplification, by Proposition 3, $m_1 \neq m_2$. Therefore, m_1 properly divides m_2 or m_2 properly divides m_1. Without loss of generality, assume m_1 properly divides m_2, then p_2 cannot be in G_n because rule Simplify-H would simplify it using p_1. ■

V. Conclusion

In conclusion, we have developed a general method for proving the strategy-independent correctness of superfluous S-polynomial critera which seems to be quite powerful. We then used this methodology to prove the strategy-independent correctness of three criteria. We began by introducing the general setting of abstract Gröbner bases, where different Gröbner basis procedures correspond to different strategies for applying a small set of inference rules. Then, we used the machinery of proof orders and formal equational proofs to prove the correctness of arbitrary strategies meeting some simple requirements. We observed that in proving the correctness of a Gröbner basis procedure $\mathfrak{G}$, S-polynomials are only needed to eliminate *peaks* in the formal proofs constructed by $\mathfrak{G}$. This suggested a methodology for proving the correctness of superfluous S-polynomial criteria. The key idea was to reduce the strategy-independent admissibility of superfluous S-polynomial criteria to the existence of "smaller" proofs in the absence of S-polynomials deemed superfluous by the criteria under investigation.

VI. Acknowledgements

We are very grateful to a number of anonymous referees for their helpful suggestions which have improved our paper.

References

[1] L. Bachmair and N. Dershowitz. Equational inference, canonical proofs, and proof orderings. *Journal of ACM*, 42(2), 1994.

[2] L. Bachmair and H. Ganzinger. Buchberger's algorithm: A constraint-based completion procedure. In *Constraints in Computational Logics, First International Conference, CCL'94*, volume 845 of *LNCS*, 1994.

[3] L. Bachmair and A. Tiwari. D-bases for polynomial ideals over commutative noetherian rings. In *Rewriting Techniques and Applications, RTA'97*, volume 1103 of *LNCS*, 1997.

[4] B. Buchberger. Ein algorithmus zum auffinden der basiselemente des restklassenringes nach einem nulldimensionalen polynomideal. Technical report, Mathematical Institute, University of Innsbruck, Austria, 1965.

[5] B. Buchberger. Ein algorithmisches Kriterium für die Lösbarkeit eines algebraischen Gleichungssystems. *Aequationes mathematicea*, 4(3), 1970.

[6] B. Buchberger. A criterion for detecting unnecessary reductions in the construction of groebner bases. In *Symposium on Symbolic and Algebraic Manipulation (EUROSAM)* , volume 72, 1979.

[7] M. Caboara, M. Kreuzer, and L. Robbiano. Efficiently computing minimal sets of critical pairs. *J. Symb. Comput.*, 38(4):1169–1190, 2004.

[8] L. de Moura and N. Bjørner. Z3: An efficient smt solver. In *Tools and Algorithms for the Construction and Analysis of Systems, 14th International Conference, TACAS'08*, volume 4963 of *LNCS*, 2008.

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

[10] J.-C. Faugère. A new efficient algorithm for computing Grbner bases without reduction to zero (F5) . In *International Symposium on Symbolic and Algebraic Computation (ISSAC)*, 2002.

[11] D. E. Knuth and P. B. Bendix. Simple word problems in universal algebras. *Computational Problems in Abstract Algebra*, 1970.

[12] R. Loos. Term reduction systems and algebraic algorithms. In *5th GI Workshop on Artificial Intelligence*, 1981.

[13] A. Riazanov and A. Voronkov. Limited resource strategy in resolution theorem. *Journal of Symbolic Computation*, 36(1–2), 2003.

[14] F. Winkler. Reducing the Complexity of the Knuth-Bendix Completion Algorithm: A "Unification" of Different Approaches. In *European Conference on Computer Algebra (EUROCAL'85)*, volume 204 of *LNCS*, 1985.

[1] By "attempts to apply" we mean that Superpose is either applied as usual, or it is tried but is ultimately skipped because of an active side-condition φ barring its application.

[2] As a very helpful referee pointed out, it is perhaps unlikely that this criteria will be very effective in practice, especially when the Gebauer-Möller criteria are used [7]. Nevertheless, we find it to be an interesting example of the usefulness of Observation 1 as the basis of a methodology for proving the strategy-independent correctness of superfluous S-polynomial criteria.