A Variant of the F4 Algorithm

Antoine Joux\textsuperscript{1,2} and Vanessa Vitse\textsuperscript{2}

\textsuperscript{1} Direction Générale de l’Armement (DGA)
\textsuperscript{2} Université de Versailles Saint-Quentin, Laboratoire PRISM, 45 av. des États-Unis, 78035 Versailles cedex, France
antoine.joux@m4x.org, vanessa.vitse@prism.uvsq.fr

Abstract. Algebraic cryptanalysis usually requires to find solutions of several similar polynomial systems. A standard tool to solve this problem consists of computing the Gröbner bases of the corresponding ideals, and Faugère’s F4 and F5 are two well-known algorithms for this task. In this paper, we adapt the “Gröbner trace” method of Traverso to the context of F4. The resulting variant is a heuristic algorithm, well suited to algebraic attacks of cryptosystems since it is designed to compute with high probability Gröbner bases of a set of polynomial systems having the same shape. It is faster than F4 as it avoids all reductions to zero, but preserves its simplicity and its efficiency, thus competing with F5.

Keywords: Gröbner basis, Gröbner trace, F4, F5, multivariate cryptography, algebraic cryptanalysis.

1 Introduction

The goal of algebraic cryptanalysis is to break cryptosystems by using mathematical tools coming from symbolic computation and modern algebra. More precisely, an algebraic attack can be decomposed in two steps: first the cryptosystem and its specifics have to be converted into a set of multivariate polynomial equations, then the solutions of the obtained polynomial system have to be computed. The security of a cryptographic primitive thus strongly relies on the difficulty of solving the associated polynomial system. These attacks have been proven to be very efficient for both public key or symmetric cryptosystems and stream ciphers (see [2] for a thorough introduction to the subject).

In this article, we focus on the polynomial system solving part. It is well known that this problem is very difficult (NP-hard in general). However, for many instances coming from algebraic attacks, the resolution is easier than in the worst-case scenario. Gröbner bases, first introduced in [6], are a fundamental tool for tackling this problem. Historically, one can distinguish two families of Gröbner basis computation algorithms: the first one consists of developments of Buchberger’s original algorithm [8,14,15,19], while the second can be traced back to the theory of elimination and resultants and relies on Gaussian elimination of Macaulay matrices [10,24,25,26]. Which algorithm to use depends on the shape and properties of the cryptosystem and its underlying polynomial system (base field, degrees of the polynomials, number of variables, symmetries...).
Faugère’s F4 algorithm [14] combines ideas from both families. It is probably the most efficient installation of Buchberger’s original algorithm, and uses Gaussian elimination to speed up the time-consuming step of “critical pair” reductions. It set new records in Gröbner basis computation when it was published a decade ago, and its implementation in Magma [5] is still considered as a major reference today. However, F4 shares the main drawback of Buchberger’s algorithm: it spends a lot of time computing useless reductions. This issue was addressed by Faugère’s next algorithm, F5 [15], which first rose to fame with the cryptanalysis of the HFE Challenge [16]. Since then, it has been successfully used to break several other cryptosystems (e.g. [4,17]), increasing considerably the popularity of algebraic attacks. It is often considered as the most efficient algorithm for computing Gröbner bases over finite fields and its performances are for the main part attributable to the use of an elaborate criterion. Indeed, the F5 criterion allows to skip much more unnecessary critical pairs than the classical Buchberger’s criteria [7]; actually it eliminates a priori all reductions to zero under the mild assumption that the system forms a semi-regular sequence [3]. Nevertheless, this comes at the price of degraded performances in the reduction step: during the course of the F5 algorithm, many reductions are forbidden for “signature” compatibility conditions, giving rise to polynomials that are either redundant (not “top-reduced” [13]), or whose tails are left almost unreduced.

In many instances of algebraic attacks, one has to compute Gröbner bases for numerous polynomial systems that have the same shape, and whose coefficients are either random or depend on a relatively small number of parameters. In this context, one should use specifically-devised algorithms that take this information into account. A first idea would be to compute a parametric or comprehensive Gröbner basis [30]; its specializations yield the Gröbner bases of all the ideals in a parametric polynomial system. However, for the instances arising in cryptanalysis, the computational cost of a comprehensive Gröbner basis is prohibitive. Another method was proposed by Traverso in the context of modular computations of rational Gröbner bases [29]: by storing the trace of an initial execution of the Buchberger algorithm, one can greatly increase the speed of almost all subsequent computations. Surprisingly, it seems that this approach was never applied to cryptanalysis until now.

We present in this paper how a similar method allows to avoid all reductions to zero in the F4 algorithm after an initial precomputation. A list of relevant critical pairs is extracted from a first F4 execution, and is used for all following computations; the precomputation overhead is largely compensated by the efficiency of the F4 reduction step, yielding theoretically better performances than F5. This algorithm is by nature probabilistic: the precomputed list is in general not valid for all the subsequent systems. One of the main contribution of this article is to give a complete analysis of this F4 variant and to estimate its probability of failure, which is usually very small.

The paper is organized as follows. After recalling the basic structure of Buchberger-type algorithms, we explain in section 2 how to adapt it to the