Article Submitted to Journal of Symbolic Computation

Factoring Polynomials over Large Finite Fields and Stable Coloring of Tournaments *

Qi Cheng[1] and Ming-Deh Huang[2]

[1] *School of Computer Science, University of Oklahoma, Norman, OK, USA*
[2] *Department of Computer Science, University of Southern California Los Angeles, CA, USA*

Abstract

We develop a new algorithm for factoring polynomials over finite fields by exploring an interesting connection between the algebraic factoring problem and the combinatorial problem of stable coloring of tournaments. Assuming GRH, we present an algorithm which can be viewed as a recursive refinement scheme through which most cases of polynomials are completely factored in deterministic polynomial time within the first level of refinement; most of the remaining cases are factored completely within the second level refinement, and so on. All cases are completely factored after no more than $\log n/1.5$ levels of refinement. In the worst case, the algorithm will perform polynomial amount of ring operations on rings of dimensions less than $n^{\frac{\log n+O(1)}{3}}$ over $\mathbf{F}_p$ in order to factor a polynomial of degree n over $\mathbf{F}_p$, while the best previously known algorithm requires ring operations on a ring of dimension $n^{\frac{\log n+O(1)}{2}}$. We also show that under a purely combinatorial conjecture concerning tournaments, our algorithm has polynomial time complexity.

1. Introduction

Factoring polynomials over finite fields is an important primitive operation in computational number theory. It has extensive applications in mathematics, engineering and information science. Although there are efficient randomized algorithms to solve this problem, it remains open whether there exists a deterministic polynomial time algorithm for it, even assuming standard number theoretical conjectures. The deterministic complexity of this fundamental problem is the main focus of this paper.

*Part of this paper, in its preliminary form, appeared in the proceeding of Algorithmic Number Theory Symposium (ANTS) IV

Let q be a prime power p^n and $f \in \mathbf{F}_q[x]$ be of degree n given in the dense representation which requires $O(n \log q)$ bits. It is well-known that factoring f over $\mathbf{F}_q$ can be reduced in deterministic polynomial time to factoring a completely splitting polynomial $f' \in \mathbf{F}_p[x]$ in polynomial time [4]. We remark in passing that when p is small, the reduction yields an efficient deterministic algorithm to factor f. In light of the above-mentioned reduction we will assume without loss of generality that the input polynomial f is completely splitting and separable over $\mathbf{F}_p$ in the sense that all roots of f are distinct and in $\mathbf{F}_p$.

The approach which underlies the existing efficient randomized algorithms for solving the problem can be regarded as factoring by external asymmetry. Indeed suppose the roots of the input polynomial f are $\alpha_1, \alpha_2, \cdots, \alpha_n \in \mathbf{F}_p$. Given $h \in \mathbf{F}_p$, the polynomial

$$g = (x - (\alpha_1 - h))(x - (\alpha_2 - h)) \cdots (x - (\alpha_n - h)) \in \mathbf{F}_p[x]$$

can be computed efficiently from f without knowing $\alpha_1, \alpha_2, \cdots, \alpha_n$. Suppose in the set $\{\alpha_1 - h, \cdots, \alpha_n - h\}$, some of elements are quadratic residue and the others are not. Then computing

$$GCD(g, x^{\frac{p-1}{2}} - 1)$$

will produce a nontrivial factor of g, and consequently a nontrivial factor of f. This is the main idea behind the random polynomial time algorithm [6, 12] which relies on an asymmetry among the roots of the polynomial in relation to an external element h. This algorithm is very effective in practice. However all attempts to derandomize it have failed.

Various researchers then started to explore the internal structure of roots for factoring. Ronyai [14] first observed that the combinatorial property of roots can play a role in factoring. Assuming the *Generalized Riemann Hypothesis* (GRH), he gave an algorithm which factors polynomials in time $O((n \log p)^{cr})$, where c is an absolute constant and r is an arbitrary factor of n which is greater than 1. In particular we can factor polynomials with bounded degrees in polynomial time, and we can efficiently split even degree polynomials. By generalizing his idea to higher extension algebra, Evdokimov [8] proposed a deterministic algorithm with time complexity $(n^{\log n} \log p)^{O(1)}$, assuming GRH. His algorithm requires a polynomial amount of ring operations in rings of dimension $n^{O(1)+\frac{\log n}{2}}$ in the worst case. Gao [10] pointed out that when it is hard to factor a polynomial using Evdokimov's algorithm, the roots of the polynomial possess an impressive symmetric property which he called *square balanced*.

In this paper we continue to investigate the approach of factoring polynomials by the internal structure of the roots. We develop new algorithms for factoring polynomials over finite fields by exploring an interesting connection between the algebraic factoring problem and the combinatorial problem of stable coloring of tournaments. In this approach we associate a polynomial to be factored with a tournament on its roots. We design algebraic procedures that explore symmetry

in the associated tournament and cause the polynomial to split as asymmetry in the tournament is detected. Further splitting, if necessary, is effected as deeper levels of symmetry are explored through algebraic means. The resulting algorithm can be viewed as a recursive refinement scheme through which most cases of polynomials, regardless of their degrees, are split completely at the first level within polynomial time, most of the remaining cases are split completely before the end of the second level refinement, and so on.

The first level of our refinement scheme is a procedure which uses Ronyai's method [14] as the building block. The basic observation is that Ronyai's method, when applied to a polynomial, groups the roots of the polynomial into factors according to scores in the associated tournament. By refining the method we obtain a procedure which implicitly performs stable coloring on the tournament. As combinatorial theory shows that most graphs decompose into singletons under a stable coloring, we can similarly show that most polynomials decompose into linear factors under this procedure. Should a non-linear factor survive the first level of refinement, it will be passed to higher levels of refinement where algebraic procedures are employed to explore higher levels of stable coloring on the tournament.

The resulting deterministic algorithm has $(n^{\log n} \log p)^{O(1)}$ worst case complexity. Moreover, all but at most $2^{-n/5}$ fraction of cases exit at the first level of the refinement scheme being completely factored. Then most of the remaining cases are split at the next level of refinement, and so on. The amount of time in going through the i-th level of refinement is bounded by $(n^i \log p)^{O(1)}$. All cases are completely factored after no more than $\log n/1.5$ levels of refinement. This is an improvement over Evdokimov's algorithm where $\log n$ levels of ring extensions are necessary in the worst case. It is worth noting that our improvement depends on a combinatorial result concerning the nonexistence of the so-called triply-regular tournaments.

Our result assumes GRH. In bounding the fraction of cases that may need to go on to higher level of refinement we need the additional assumption that $n \leq \log p/2$.

Under a purely combinatorial conjecture concerning tournaments, we can show that the maximum number of levels of refinement in our algorithm is bounded by a constant, which implies that our algorithm has worst case polynomial complexity. There are strong evidences for the conjecture which we discuss in Section 7.

We remark that the method in this paper can be used to prove that polynomials of degree n over $\mathbf{F}_p$ can be factored completely in time $(n^{\delta(p)} \log p)^{O(1)}$, where $\delta(p)$ is the size of largest transitive subgraph in the *multicolor* cyclotomic tournament over $\mathbf{F}_p$ (see definition in section 2). An interesting open problem is to derive a sharp upper bound for $\delta(p)$.

2. Tournaments and finite fields

A tournament is a complete simple digraph. Let u and v be two vertices of the tournament. We say that v dominates u if there is an arc from v to u. The score of a vertex v in a tournament is the number of vertices dominated by v. Denote by $O(v)$ all the nodes in the tournament dominated by v and by $I(v)$ all the nodes dominating v. A tournament is *regular* if every vertex has the same score. It is easy to see that if n is the number of vertices in a regular tournament, then n must be an odd number and the score of any vertex is $\frac{n-1}{2}$.

Let p be prime. We construct tournaments on $\mathbf{F}_p$ as follows. First assume that $p \equiv 3 \pmod 4$. For $a, b \in \mathbf{F}_p$, there is an arc from a to b (a dominates b) iff $a-b$ is a quadratic non-residue. This tournament is well studied in graph theory and is called *quadratic residue tournament* or *paley tournament* [13]. It is proved [5] that the paley tournaments are the *most* symmetric tournaments, because they are the only tournaments which are arc symmetric.

In case of $p \equiv 1 \pmod 4$, we will need a more general notion of tournament - *multicolor tournament*. A multicolor tournament $T = (V, E)$ is a complete digraph where each arc is associated with a unique color that satisfies the following conditions.

1. For any two vertices $i, j \in V$, The color of (i, j) is different from the color of (j, i).
2. For any four vertices $i, j, s, t \in V$, (i, j) has the same color as (s, t) iff (j, i) has the same color as (t, s).

From the definition, we see that the set of colors in a multicolor tournament is partitioned into pairs. In particular, the number of colors must be even. We can think that tournaments in the ordinary sense are two-color tournaments: one color is "dominating" and the other is "being dominated".

Suppose $p - 1 = mr$, where m is even and relatively prime to r. We can construct a multicolor tournament over $\mathbf{F}_p$ with the set of the m-th roots of unity as colors by labeling an arc (i, j) with the color $(i-j)^r$. Note that an m-th root of unity ζ and its additive inverse $-\zeta$ form a complementary pair of colors. If (i, j) has color α, we say that i α-dominates j. We can define the score of a vertex with respect to a color in a straight-forward way. The resulting multicolor tournament is called the *m-th cyclotomic tournament* over $\mathbf{F}_p$.

In this paper, for odd primes p, we use "the cyclotomic tournament over $\mathbf{F}_p$" to refer the unique 2^k-th cyclotomic tournament where $p - 1 = 2^k r$ with r odd, unless otherwise specified. Note that the quadratic residue tournament for $\mathbf{F}_p$ with $p \equiv 3 \pmod 4$ is essentially the (second) cyclotomic tournament over $\mathbf{F}_p$. We remark that although the number of colors can be very large, there are at most $O(n^2)$ arc colors in any induced subtournament of n vertices, and the colors come in pairs.

3. The first level of refinement

Let $f \in \mathbf{F}_p[x]$ be a polynomial with all roots distinct and in $\mathbf{F}_p$ with

$$f = (x - a_1)(x - a_2) \cdots (x - a_n).$$

We associate with f the subtournament induced by a_1, ..., a_n in the cyclotomic tournament and denote it by $\mathbf{T}(f)$. Figuare 1 shows the subtournament associated with $x^7 + 318x^6 + 340x^5 + 2352x^4 + 2151x^3 + 1774x^2 + 3116x + 3648 \equiv (x-2178)(x-1811)(x-846)(x-1032)(x-1111)(x-3749)(x-3784) \bmod 4943$, for $m = 2$.

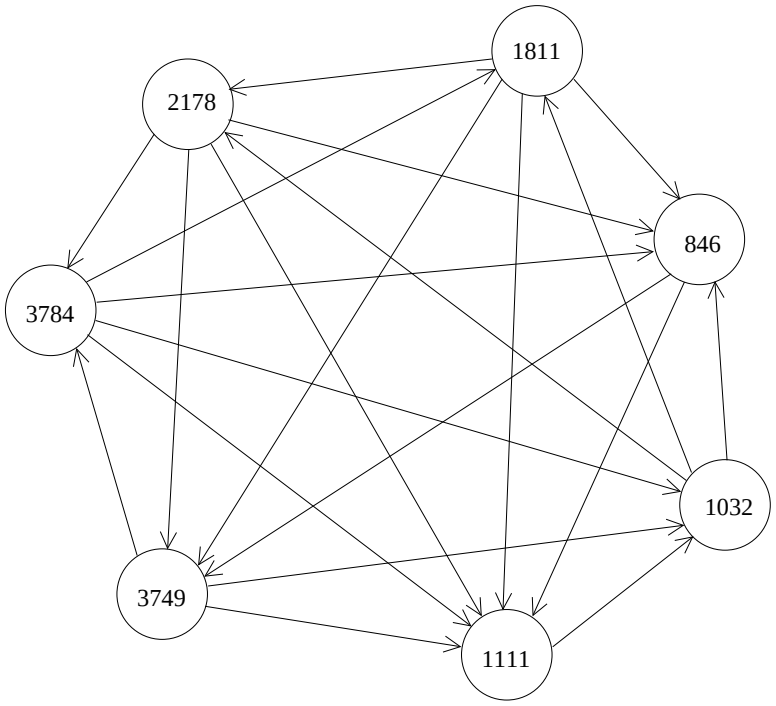

Figure 1: The subtournament associated with $x^7+318x^6+340x^5+2352x^4+2151x^3+1774x^2+3116x+3648 \equiv (x-2178)(x-1811)(x-846)(x-1032)(x-1111)(x-3749)(x-3784) \bmod 4943$.

Define the *score polynomial* of f with respect to a color α, denoted by $S_\alpha(f)$, as

$$\prod_{i=1}^{n} (x - a_i)^{b_i} \tag{1}$$

where b_i is the score of a_i with respect to a color α.

For example, the score polynomial with respect to color -1 of $x^7 + 318x^6 + 340x^5 + 2352x^4 + 2151x^3 + 1774x^2 + 3116x + 3648 \bmod 4943$ is

$$x^{21} + 1192x^{20} + 4361x^{19} + 2458x^{18} + 3080x^{17} + 4871x^{16} + 3771x^{15} +$$

$$1508x^{14} + 2004x^{13} + 4279x^{12} + 4636x^{11} + 3173x^{10} + 1347x^9 + 4791x^8 + 135x^7 + 1722x^6 + 3865x^5 + 880x^4 + 2268x^3 + 3093x^2 + 1044x + 517,$$

which is congruent to $(x-2178)^4(x-1811)^4(x-846)^2(x-1032)^3(x-1111)(x-3749)^3(x-3784)^4 \bmod 4943$.

An interesting observation is that the score polynomial $S(f)$ can be computed in polynomial time using Ronyai's method.

THEOREM 3.1: *Assuming GRH, then there is an algorithm that given a prime p and a polynomial f in $\mathbf{F}_p[x]$ of degree n, (1) finds all the arc colors in $\mathbf{T}(f)$; (2) for any arc color α in $\mathbf{T}(f)$, computes the score polynomial $S_\alpha(f)$ in time $(n \log p)^{O(1)}$.*

This is a special case of Theorem 4.1, which we will introduce and prove later.

From f and $S_\alpha(f)$ we can split f into factors each having roots of the same score with the following procedure.

ALGORITHM 1: Input f with distinct roots in $\mathbf{F}_p$.

```
Compute the arc colors of T(f);
For every arc color pair (α, ᾱ) in T(f) do
    Calculate S_α(f);
    Let f_1 = S_α(f), f_2 = f;
    while f_1 ≠ 1 do
          While f_2|f_1 do f_1 = f_1/f_2 endwhile;
          put f_2/gcd(f_1, f_2) into output set;
          let f_2 = gcd(f_1, f_2);
    endwhile;
endfor;
```

Figure 2 shows that Algorithm 1 classifies the roots by score and split the polynomial $x^7 + 318x^6 + 340x^5 + 2352x^4 + 2151x^3 + 1774x^2 + 3116x + 3648$ into $(x-1111)(x-846)(x^2+162x+3542)(x^3+2113x^2+3241x+2087)$, where $x-1111$, $x-846$, $x^2+162x+3542$ and $x^3+2113x^2+3241x+2087$ correspond to the vertex sets with score 1, 2, 3 and 4 respectively.

We call a tournament *regular* if every vertex dominates the same number of vertices. We call a polynomial *regular* if it induces a regular tournament. For an irregular polynomial (tournament), after we apply the algorithm, we get several factors corresponding to the scores. However, we need not stop here. Suppose a factor is not regular (i.e. the roots of the factor do not induce a regular subtournament). Then it will be split when the algorithm is applied to it. Applying the algorithm to the product of two factors may also cause further splitting. These ideas lead to the following refinement procedure on a set of factors with disjoint sets of roots.

ALGORITHM 2: Input a set of relatively prime polynomials $\{f_1, f_2, \cdots, f_n\}$, each with distinct roots in $\mathbf{F}_p$.

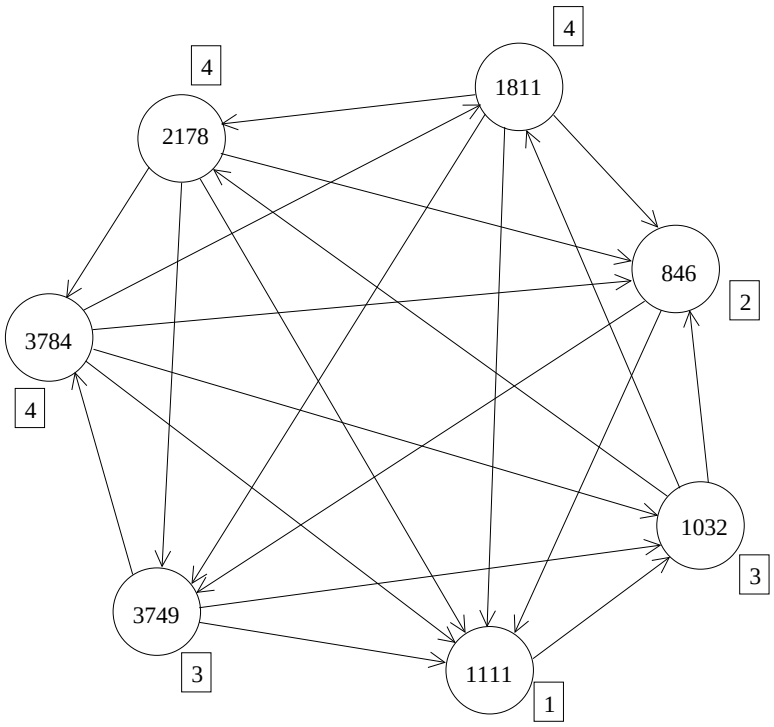

Figure 2: Applying Algorithm 1 on the polynomial $x^7 + 318x^6 + 340x^5 + 2352x^4 + 2151x^3 + 1774x^2 + 3116x + 3648 \bmod 4943$.

1. For $1 \leq i \leq n$, apply the algorithm (1) on f_i, let the set of output polynomials be S_i;
2. For $1 \leq i < j \leq n$, apply algorithm (1) on $f_i f_j$, for every output factor g, put $gcd(g, f_i)$ into S_i, $gcd(g, f_j)$ into S_j;
3. For every S_i, if there are any two polynomial $g, h \in S_i$, such that $gcd(g, h) \neq 1$, then remove g, h from S_i and add $gcd(g, h)$, $g/gcd(g, h)$, $h/gcd(g, h)$ into S_i.

We apply Algorithm (1) to f and then apply Algorithm (2) to the set of factors output by Algorithm (1). As we observe what is happening to the underlying tournament, we find that the process is very similar to elementary refinement for tournaments [3]. The first procedure partitions the roots by score. Suppose $C_1, C_2, \cdots, C_h$ form the partition. For all roots x, let $N_i(x)$ denote the number of neighbors of x in C_i. In applying Algorithm (2) to the corresponding set of factors, we first apply Algorithm (1) on C_i. This amounts to comparing $N_i(x)$ for all $x \in C_i$. Then we apply Algorithm (1) on $C_i C_j$. This amounts to comparing $N_i(x) + N_j(x)$ for all $x \in C_i \cup C_j$. When we exit Algorithm (2), we have refined the partition in the following manner. Two roots x, y are now in the same class iff they are in same class before the refinement, and

$$(N_1(x), N_2(x), \cdots, N_h(x)) = (N_1(y), N_2(y), \cdots, N_h(y)).$$

After we repeat Algorithm (2) at most n times, we will reach a point where the partition remains unchanged. At this point the partition of the roots is a *stable coloring* in the following sense.

Definition: A partition of the vertex set of a tournament into subsets, $C_1, \ldots, C_m$, is a **level-one stable coloring** if

1. C_i, $1 \leq i \leq m$, induces a regular subtournament,
2. For $1 \leq i, j \leq m$, for all $u, v \in C_i$, u dominates the same number of vertices in C_j as v does.

Figure 3 shows how the algorithm acts on the polynomial $x^7 + 318x^6 + 340x^5 + 2352x^4 + 2151x^3 + 1774x^2 + 3116x + 3648$, resulting a complete factorization of the polynomial.

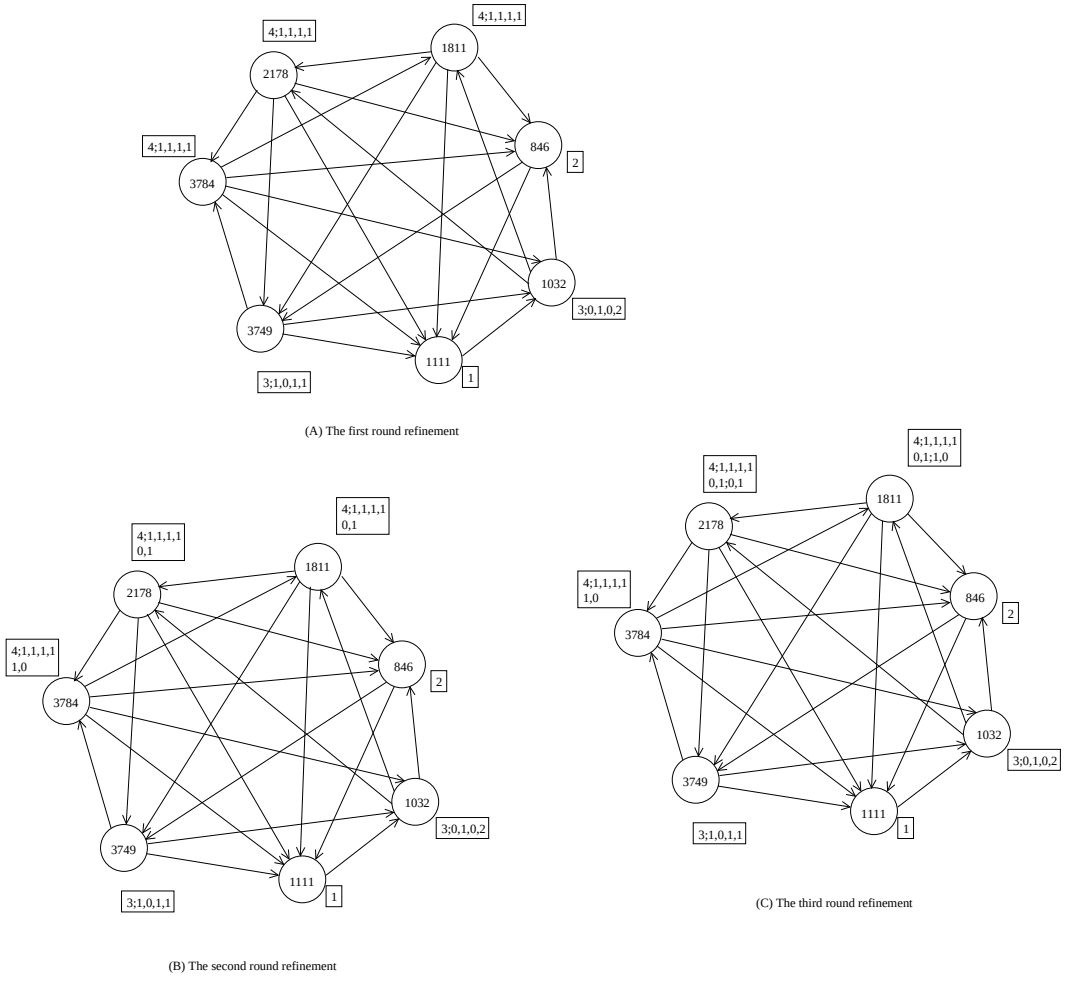

Figure 3: Applying Algorithm 2 on the polynomial $x^7 + 318x^6 + 340x^5 + 2352x^4 + 2151x^3 + 1774x^2 + 3116x + 3648$.

At this point we have completed the description of the first level of refinement in our algorithm for factoring polynomials over $\mathbf{F}_p$. It is interesting to observe that after the level-one refinement, each factor is the union of some vertex orbits under the automorphism group of the tournament.

THEOREM 3.2: *The fraction of completely splitting separable polynomials over* $\mathbf{F}_p$ *with degree* $n < \log p/2$ *that cannot be split completely by the first level of refinement is less than* $2^{-n/5}$.

Proof: The following proposition was proved in [7]:

PROPOSITION 3.1: *Let* T *be a random tournament on* n *vertices selected from the uniform distribution over the set of labeled* n*-tournaments. The probability that* T *cannot be factored into singletons by the refinement is less than* $2^{-n/5}$.

If a separable polynomial f has all roots on $\mathbf{F}_p$, its root set will induce a subtournament in paley tournament. On the other hand, every induced subtournament in paley tournament corresponds to a completely splitting separable polynomial over $\mathbf{F}_p$. It was proved in [9] that every labeled tournament (graph) of order n occurs roughly as frequently as it should as induced subtournament (subgraph) in paley tournament (graph), namely, with probability $(1 + o(1))/2^{\binom{n}{2}}$, when $n < (\log p)/2$. Hence the theorem follows from Proposition 3.1. □

4. The main theorem

A factor which remains after the first level of refinement has an underlying tournament which is regular. To refine it further we look for coherent stable colorings on all the subtournaments obtained by removing one root (vertex) from the regular tournament.

Definition: Let $C_1, C_2, \cdots, C_n$ be a level-one stable coloring for a tournament T, C'_1, C'_2, $\cdots$, C'_m be a level-one stable coloring for a tournament T', we say the two coloring are coherent if (reordering the lists if necessary)

- $n = m$ and $|C_i| = |C'_i|$, for all $1 \leq i \leq n$.
- For any arc color α and $i \neq j$, if every vertex in C_i has k α-arcs to C_j, then every vertex in C'_i has k α-arcs to C'_j.

Definition: Suppose T is a regular tournament with vertices v_1, ..., v_n. Suppose $C_1^{v_i}, C_2^{v_i}, \cdots, C_{m_i}^{v_i}$ is a level-one stable coloring of $T - v_i$. We say that the collection of these level-one stable colorings constitutes a level-two stable coloring for T, if they are coherent with one another.

Below we describe how a regular polynomial can be manipulated algebraically so that either a level-two stable coloring on the underlying tournament is identified, or the polynomial is split.

In general let f be a polynomial of degree n with distinct roots a_1, ..., a_n in $\mathbf{F}_p$ as before. Let $R = \mathbf{F}_p[x]/(f) = \mathbf{F}_p[A]$, where $A = x \bmod f$. Let $f^* \in R[x]$ so that

$$f(x) = (x - A)f^*.$$

There exist uniquely determined primitive idempotents $e_i \in R$, $1 \leq i \leq n$, such that $\sum_{i=1}^n e_i = 1, e_i e_j = e_i \delta_{ij}$. In fact, $e_i = \prod_{j \neq i}(A - a_j) / \prod_{j \neq i}(a_i - a_j)$. For every element $c \in R$, these exist unique elements $c_1, \cdots, c_n \in \mathbf{F}_p$ such that $c = \sum_{i=1}^n c_i e_i$. We call c_i the ith canonical projection of c on $\mathbf{F}_p$. The canonical projections of a polynomial in $R[x]$ can be similarly defined. In particular,

$$f^* = \sum_{i=1}^{n} f_i e_i \text{ where } f_i = \prod_{j \neq i}(x - a_j).$$

We remark that since f_i represents the subtournament obtained from the tournament of f by removing the root a_i, f^* succinctly represents all these subtournaments simultaneously.

For example, it can be verified that

$$F(x) = x^7 + 6406x^6 + 2344x^5 + 7583x^4 + 8118x^3 + 4906x^2 + 3187x + 829$$

is a regular polynomial. Hence it can not be splitted using Algorithm 1. We can compute $F^*(x)$:

$$x^6 + (A + 6406)x^5 + (A^2 + 6406A + 2344)x^4 + (A^3 + 6406A^2 + 2344A + 7583)x^3 + (A^4 + 6406A^3 + 2344A^2 + 7583A + 8114)x^2 (A^5 + 6406A^4 + 2344A^3 + 7583A^2 + 8114A + 4906)x + (A^6 + 6406A^5 + 2344A^4 + 7583A^3 + 8114A^2 + 4906A + 3187)$$

An element of R has the form $h(A)$ where h is a polynomial over $\mathbf{F}_p$ of degree less than n. It is a zero-divisor in R iff the GCD of $h(x)$ and f is not 1. In other word as we attempt to find an inverse of $h(A)$ in R by computing the GCD of $h(x)$ and $f(x)$, we either succeed or find a nontrivial factor of f.

More generally, a *completely splitting semisimple algebra* [8] of dimension m over $\mathbf{F}_p$ is of the form

$$R = \bigoplus_{1 \leq i \leq m} \mathbf{F}_p e_i$$

where $e_i \in R$, $1 \leq i \leq m$, and $\sum_{i=1}^m e_i = 1, e_i e_j = e_i \delta_{ij}$. The elements e_i are called *primitive idempotents*. For any $g(x) \in R[x]$, we have $g(x) = \sum_{i=1}^m g_i(x) e_i$, where $g_i(x) \in \mathbf{F}_p[x]$. We say that g is a *completely splitting polynomial* over R if for all i the roots of g_i are all distinct and in $\mathbf{F}_p$. Define the *score polynomial* of $g(x)$ with respect to a color α by

$$S_\alpha(g(x)) = \sum_{i=1}^{m} S_\alpha(g_i(x)) e_i.$$

Thus $S_\alpha(g)$ succinctly represents the set of score polynomials $S_\alpha(g_i)$ for $i = 1, \cdots, m$.

In the following theorem we extend Ronyai's algorithm to polynomials over general completely splitting semisimple algebras over $\mathbf{F}_p$. In the theorem we assume: (1) Given $a \in R$, we can determine whether a is a zero divisor, and if not, find its inverse in polynomial amount of ring operations. (2) If $a^{(p-1)/l}$ is an idempotent of algebra R, at least l distinct l-th roots of a can be found in $(l \log p)^{O(1)}$ ring operations. Note that we do not assume that these idempotents are explicitly given.

THEOREM 4.1: *Suppose R is a ring with the above properties. Let $f \in R[x]$ be a completely splitting monic polynomial with degree n. There is a deterministic algorithm which either finds a nontrivial zero-divisor in R (hence a nontrivial factor of f), or*

1. *finds all the colors in $\mathbf{T}(f_1)$, where f_1 is the first canonical projection of f, and*
2. *for any color α in $\mathbf{T}(f_1)$, computes $S_\alpha(f)$ in $(n \log p)^{O(1)}$ ring operations.*

Proof: Let $f = \sum_{i=0}^{n} a_i x^i \in R[x], a_n = 1$, be a monic polynomial, suppose

$$f = \sum_{1 \le i \le m} f_i(x) e_i,$$

where for any $1 \le i \le n$, $f_i(x) \in \mathbf{F}_p[x]$ splits completely over F_p into n distinct linear factors.

Let A be the companion matrix of f,

$$A = \sum_{1 \le i \le m} A_i e_i,$$

where A_i's are $n \times n$ matrices over $\mathbf{F}_p$. Let $a_1^{(i)}, a_2^{(i)}, \cdots, a_n^{(i)}$ be the eigenvalues of A_i and $\mu_1^{(i)}, \mu_2^{(i)}, \cdots, \mu_n^{(i)}$ be the corresponding characteristic vectors.

Consider the following linear transformation

$$G = I \bigotimes A - A \bigotimes I = \sum_{1 \le i \le m} e_i (I \bigotimes A_i) - e_i (A_i \bigotimes I),$$

which acts on R^{n^2}. The vectors in $L = G(R^{n^2})$ must have the form

$$\sum_{j,k,j \ne k} \sum_i c_{i,j,k} \mu_j^{(i)} \bigotimes \mu_k^{(i)} e_i = \sum_{j,k,j \ne k} (\sum_i c_{j,k,i} e_i)(\sum_i \mu_j^{(i)} \bigotimes \mu_k^{(i)} e_i).$$

It is a free module, having a basis $\{\sum_{1 \le i \le m} \mu_j^{(i)} \bigotimes \mu_k^{(i)} e_i | j \ne k, 1 \le j, k \le n\}$. The dimension of L is $n(n-1)$.

Let H be the transformation of G on L. Let the characteristic polynomial

of $C = H^r$ be $c(x) = \sum_i c_i(x)e_i$. We consider the case when $c(x) \in \mathbf{F}_p[x]$. If $c(x) \notin \mathbf{F}_p[x]$, we will show later in the proof that a zero-divisor in R can be found. All the roots of $c(x)$ are the colors in f_1

Let α be one of roots of $c(x)$. Let T be the kernel of $C - \alpha I$ as a linear map on L. The vector in T must have form

$$\sum_i \sum_{j,k,(a_j^{(i)} - a_k^{(i)})^r = \alpha,} c_{i,j,k} \mu_j^{(i)} \bigotimes \mu_k^{(i)} e_i.$$

Let U be the transformation $A \bigotimes I$ on T. The characteristic polynomial of U is the score polynomial (with respect to α).

Now we describe the algorithm to compute the score polynomial. The companion matrix of f is

$$A = \begin{pmatrix} 0 & 1 & 0 & \cdots & 0 & 0 \\ 0 & 0 & 1 & \cdots & 0 & 0 \\ 0 & 0 & 0 & \cdots & 0 & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & 0 & 0 & \cdots & 0 & 1 \\ -a_0 & -a_1 & -a_2 & \cdots & -a_{n-2} & -a_{n-1} \end{pmatrix}$$

We compute $I \bigotimes A$ and $A \bigotimes I$, using the Kronecker product of matrices. Then we construct the linear space $L = G(R^{n^2})$ by doing the Gauss elimination on $G = I \bigotimes A - A \bigotimes I$. In the process, we either encounter a zero divisor on R, or end up with a basis for this linear space. Notice that R is a commutative ring with identity, hence it has the invariant dimension property. This implies that the number of independent generators we get should be $n(n-1)$. Notice that $C = H^r$ can be computed by the repeated squaring technique. Consider the list

$$C, C^2, C^4, \cdots, C^{2^k}.$$

Certainly that $C^{2^k} = I$. (Note that $p - 1 = 2^k r$ where r is odd.) If C^{2^i} is defined over $\mathbf{F}_p$ and its eigenvalues are known, we can compute the square roots of all its eigenvalues. Let those square roots be $\theta_1, \cdots, \theta_b$. If $C^{2^{i-1}}$ is also defined over $\mathbf{F}_p$, then we compute its eigenvalues by checking the rank of $C^{2^{i-1}} - \theta_i I$ for $1 \leq i \leq b$. If $C^{2^{i-1}}$ is not defined over $\mathbf{F}_p$, then there must exist a θ_i such that the non-zero coefficient of the lowest non-zero term of the characteristic polynomial of $C^{2^{i-1}} - \theta_i I$ is an zero-divisor. Hence we will find all the roots of $c(x)$ if $c(x) \in \mathbf{F}_p[x]$, or find a zero-divisor in R if $c(x) \notin \mathbf{F}_p[x]$.

We then need to construct basis for T, which is the kernel of $C - \alpha I$, and compute U, which is the matrix for the linear transformation $A \bigotimes I$ on T. These tasks can be done using the standard linear algebra methods. In the last step, we calculate the characteristic polynomial of U. We either get a zero divisor in the process, or obtain the score polynomial. The whole process takes polynomial amount of ring operations. The theorem follows. □

5. The second level refinement

With the above theorem we are in a position to extend Algorithms (1) and (2) in a natural way to completely splitting polynomials in $R[x]$, where R is a completely splitting algebra over $\mathbf{F}_p$ satisfying the conditions in Theorem 4.1, assuming GRH. The steps in the algorithms which involve polynomial division or GCD need some modification due to the presence of zero divisors in R. To divide a polynomial g by a polynomial h over R, we need to check whether the leading coefficient of h is a zero divisor, and if not, to find its inverse in R. Encountering a zero divisor causes an early exit in the computation.

Following [10], the GCD of two polynomials over R can be defined: Let $f(x), g(x) \in R[x]$ with $f(x) = \sum_{i=1}^{n} f_i(x)e_i, g(x) = \sum_{i=1}^{n} g_i(x)e_i$, where $f_i(x), g_i(x) \in \mathbf{F}_p[x], 1 \leq i \leq n$. Define $GCD(f,g) = \sum_{i=1}^{n} GCD(f_i, g_i)e_i$. For $f, g \in R[x]$, $GCD(f,g)$ can be computed in polynomial amount of ring operations [10], or a zero-divisor will be found.

The above discussion shows that we can extend the level-one refinement in a natural way to the polynomial f^* over a ring R satisfying the conditions in Theorem 4.1. We call this the level-two refinement on f.

THEOREM 5.1: *Assuming GRH, then there exists a deterministic polynomial time algorithm which on input a regular polynomial $f \in \mathbf{F}_p[x]$ of degree n that induces a two-color subtournament in the cyclotomic tournament over $\mathbf{F}_p$, either finds a nontrivial factor of f, or succinctly finds a level-two stable coloring for the tournament of f in the sense that it factors f^* as*

$$f^* = \prod_{i=1}^{m} g_i \text{ where } g_i = \sum_{j=1}^{n} C_i^{v_j} e_j$$

where $C_1^{v_j}$, ..., $C_m^{v_j}$ constitute a level-one stable coloring $\mathcal{C}_j$ for the subtournament obtained by removing the j-th root from f and $\mathcal{C}_1$, ..., $\mathcal{C}_n$ form a level-two stable coloring for the tournament of f. Furthermore, $m \geq 2$, (after reordering if necessary) there is $1 \leq r < m$, $\sum_{1 \leq i \leq r} deg(g_i) = \sum_{r+1 \leq i \leq m} deg(g_i) = \frac{n-1}{2}$.

Proof: $R = \mathbf{F}_p[x]/(f)$ is a completely splitting semisimple algebra, satisfying the conditions in Theorem 4.1 [8]. Given a non-trivial zero divisor in R, we can find a nontrivial factor of f efficiently. By Theorem 4.1, we can compute $S(f^*)$ in polynomial time.

Let T be the regular tournament associated with f. For a subset S of $V(T)$, we denote polynomial $\prod_{s \in S}(x - s)$ simply by S. Thus

$$f^*(x) = \sum_{i=1}^{n} (T - a_i)e_i.$$

If the tournament with n vertices is regular, then for any vertex x, every vertex

in $O(x)$ has score $(n-1)/2$ in $T-x$, while every vertex in $I(x)$ has score $(n-3)/2$. So we get polynomial

$$S(f^*) = \sum_{i=1}^{n} O(a_i)^{(n-1)/2} I(a_i)^{(n-3)/2} e_i.$$

Let's examine what happens as we apply the level-two refinement on f^*. Without early exit, the polynomial f^* is factored as $f_O^* f_I^*$ when we apply Algorithm (1), where

$$f_O^* = \sum_{i=1}^{n} O(a_i) e_i \quad \text{and } f_I^* = \sum_{i=1}^{n} I(a_i) e_i$$

As we apply Algorithm (2) on f_O^* and f_I^*, we may encounter a zero-divisor in $\mathbf{F}_p[A]$ and exit the refinement early with f split as result. If we successfully run through the refinement without an early exit, then a level-one stable coloring has been found for each T_i simultaneously. Moreover, not encountering a zero divisor means that these level-one stable coloring are coherent(notice that during the computation if we obtain a polynomial $g = \sum_i g_i(x) e_i$, such that two of the component polynomials have different degrees, then the coefficient of the highest order term of g is a zero divisor.), thus a level-two stable coloring has been succinctly constructed in the factors of f^* over R. □

Again we consider the tournaments with two colors. Define the *score vector* of a tournament as the sorted list of all the scores in the tournament. If the score vector is the same for the subtournaments induced on $O(v)$ (respectively $I(v)$) for every vertex v, it is known as *pseudo-vertex-symmetric* [1]. As a special case, a tournament is called *doubly-regular* if it is regular and for every vertex v, the subtournaments induced on $O(v)$ and $I(v)$ are regular. See the figure 4 for an example.

For a doubly regular polynomial f, f^* may go through level-two refinement being factored only into f_O^* and f_I^*. Intuitively speaking, pseudo-vertex-symmetric tournaments are highly symmetrical and rare. From the proof of Theorem 5.1, we can conclude

COROLLARY 5.1: *If f has level-two stable coloring, it corresponds to a pseudo-vertex-symmetric tournament. If f^* has only two factors, f corresponds to doubly-regular tournament.*

For tournament with more than two colors, we have the following theorem by the similar argument in proof of Theorem 5.1

THEOREM 5.2: *Assuming GRH, then there is a deterministic polynomial time algorithm which on input a regular polynomial $f \in \mathbf{F}_p[x]$ of degree n associated with m colors (m is even), either finds a nontrivial factor of f, or finds a nontrivial factor of f^* which has degree at most n/m.*

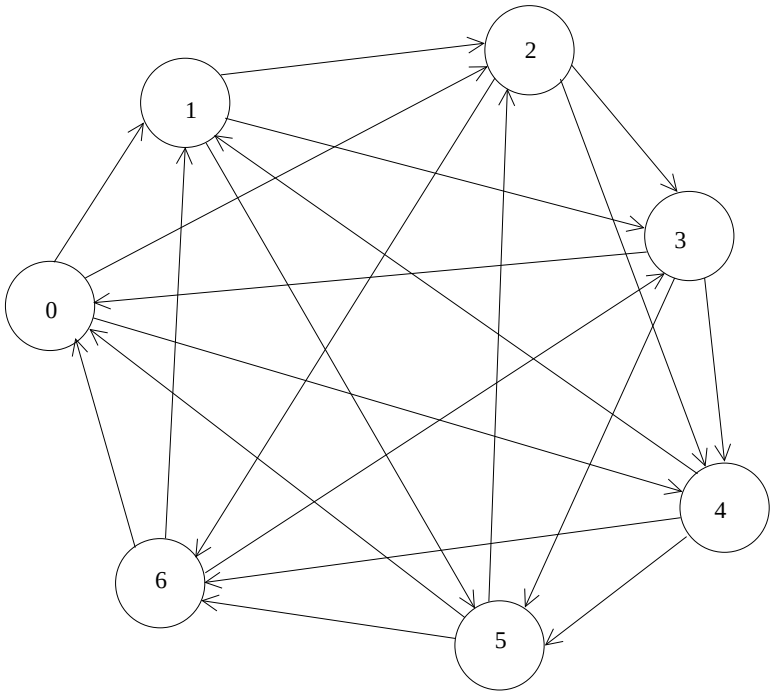

Figure 4: Example of a doubly-regular tournament.

6. Higher level refinement

We call a polynomial $g \in R[x]$ *regular* if $g(x) = \sum_{i=1}^{n} g_i(x)e_i$ where $g_i \in \mathbf{F}_p[x]$ is regular for all i. The proof of Theorem 5.1 and 5.2 can be extended in a natural way to show that there is a deterministic polynomial time algorithm which on input a regular $f \in R[x]$ of degree n, where R is a ring satisfying the conditions in Theorem 4.1, either finds a nontrivial zero divisor of R, or succinctly and simultaneously finds a level-two stable coloring for each canonical projection of f.

Suppose $f(x) \in \mathbf{F}_p[x]$ is regular with degree n. Let $R_0 = \mathbf{F}_p$, $R_1 = R_0[x]/(f(x))$. Applying Theorem 5.1 to f^*, we either find a nontrivial factor of f or split f^* over R_1. Suppose f^* is split over R_1. Let f_1 be the factor with least degree n_1. If $n_1 = 1$, we can factor f [8, Lemma 9]. Otherwise f_1 is a polynomial whose canonical projections are regular tournaments with order n_1. Let $R_2 = R_1[x]/(f_1(x))$. Then R_2 is a completely splitting algebra over $\mathbf{F}_p$ satisfying the conditions in Theorem 4.1, assuming GRH.

Let $f_1 = \sum_{i=1}^{n} T_i e_i$ where $T_i = \prod_{1 \le j \le n_1}(x - a_j^{(i)})$. Note that $f_1 = \prod_{1 \le j \le n_1}(x - \sum_{1 \le i \le n} a_j^{(i)} e_i)$. Let $B = x \pmod{f_1(x)}$ and $f_1 = (x - B)f_1^*$. The idempotents over R_2 are

$$e_j^{(2)} = \prod_{k,k \ne j} (B - \sum_{1 \le i \le n} a_k^{(i)} e_i) / \prod_{k,k \ne j} (\sum_{1 \le i \le n} a_j^{(i)} e_i - \sum_{1 \le i \le n} a_k^{(i)} e_i) \qquad (2)$$

$$= \sum_{1\le i\le n} (\prod_{k,k\neq j} (B - a_k^{(i)}) / \prod_{k,k\neq j} (a_j^{(i)} - a_k^{(i)}))e_i \quad (3)$$

Let $\epsilon_j^{(i)} = \prod_{k,k\neq j}(B - a_k^{(i)}) / \prod_{k,k\neq j}(a_j^{(i)} - a_k^{(i)})$. Then R_2's canonical primitive idempotents are $\{\epsilon_j^{(i)} e_i | 1 \le i \le n, 1 \le j \le n_1\}$, and

$$f_1^* = \sum_{i=1}^{n}\sum_{j=1}^{n_1}(T_i - a_j^{(i)})\epsilon_j^{(i)} e_i.$$

We apply Algorithm (1) and (2) on f_1^* over R_2. Either an early exit leads to the splitting of f_1 over R_1 or even the splitting f over $\mathbf{F}_p$; or a level-two stable coloring is simultaneously found on every canonical projection of f_1. Inductively, let f_i be the polynomial resulting from the i-th level of refinement with degree n_i over a completely splitting algebra R_i. Now we sketch the algorithm:

ALGORITHM 3: Input a regular $f \in \mathbf{F}_p[x]$ with degree n.

$i = 1$; $R_1 = \mathbf{F}_p[x]/(f(x))$; $n_1 = n - 1$;
$f_i = f^*$; $S_i = \{f_1\}$;
$upper - level - zero - divisor = empty$;
$upper - level - root = empty$;
while $i >= 1$ do
    If $upper - level - zero - divisor$ is not empty
        find new factor of polynomials in S_i and update S_i endif;
    If $upper - level - root$ is not empty
        find new factor of polynomials in S_i and update S_i endif;
    Set $upper - level - zero - divisor$ = empty;
    Set $upper - level - root$ = empty;
    Apply generalized Algorithm (1) and (2) on S_i until
    we cannot find new factors;
    If a zero-divisor is found then
        put the zero-divisor into the $upper - level - zero - divisor$;
        i = i-1;
    else
        Let f_i be the polynomial with the smallest degree in S_i;
        if $n_i = 1$
            put the root into $upper - level - root$;
            i=i-1;
        else
            $R_{i+1} = R_i[x]/(f_i); n_{i+1} = n_i - 1;$
            $f_{i+1} = f_i^*; S_{i+1} = \{f_{i+1}\}$;
            $i = i + 1$;
        endif
    endif

`endwhile`
`Split` f `use the zero-divisor in` R_1 `or the root of` f^*`;`

A two-color tournament is called *triply-regular* if it is regular and for every vertex v, the subtournaments induced on $O(v)$ and $I(v)$ are doubly-regular. It is a remarkable fact that there is no triply-regular tournament with $n \geq 4$ vertices[11].

THEOREM 6.1: *For any polynomial $f \in \mathbf{F}_p[x]$ of degree n, the number of levels that our algorithm go through in order to factor f completely is at most $\frac{\log n}{1.5}$.*

Proof: We know that $n_{i+1} \leq n_i/2$. If we can show that $n_{i+2} \leq n_i/8$, then $n_t \leq 1$ if $t > \frac{\log n}{1.5}$.

If the number of arc colors in one of the canonical projects is greater than 2, then $n_{i+1} \leq n_i/4$. Hence $n_{i+2} \leq n_i/8$.

Suppose that every canonical project of f_i is a two-color tournament. If any of canonical projections of f_i are not doubly-regular, then $n_{i+1} \leq n_i/4$, and $n_{i+2} \leq n_i/8$. Otherwise, if the projections of f_i are doubly-regular, it is possible that $n_{i+1} = n_i/2$, but then the projections of f_{i+1} will not be doubly-regular, since there is no nontrivial triply-regular tournament, hence $n_{i+2} \leq n_{i+1}/4$, thus we have $n_{i+2} \leq n_i/8$. □

In our algorithm, the maximum dimension of rings is:

$$n \times \frac{n}{2} \times \frac{n}{8} \times \cdots \times 1 = \frac{n^{O(1)+\frac{\log n}{2}}}{\frac{n}{4} \times \frac{n}{32} \times \cdots \times 1} = n^{O(1)+\frac{\log n}{3}}$$

7. Discussion

In general suppose $f(x)$ is a regular polynomial over $\mathbf{F}_p$ and $T = \mathbf{T}(f)$ is a tournament that admits a level-two stable coloring. Put two arcs uv and xy in the same class iff in the stable coloring of $T - u$ and $T - x$, v and y are in corresponding classes (that is $v \in C_j^u$ and $y \in C_j^x$ for some j). For any arc class G, we call graph $B_G = (V, G)$ a *base graph* for T with respect to the level-two stable coloring of T, where V is the set of vertices in T. Suppose that the level-two stable coloring is represented by the factoring of f^* into the product of g_i as in Theorem 5.1. Then the base graphs are in one-one correspondence with the factors g_i. Each base graph is a regular digraph and the set of arcs in a base graph is the union of some arc orbits in the tournament under the automorphism group of the tournament.

The bound on the number of levels in the above theorem seems to be far from being tight. Define a function β from set of tournaments to the set of natural numbers as follows. If a tournament T is not regular or doesn't have second level stable coloring, $\beta(T) = 0$. If T has second level stable coloring, $\beta(T)$ is the maximum of $\beta(\mathcal{C}, T)$ among all the level-two stable colorings $\mathcal{C}$ of T, where $\beta(\mathcal{C}, T)$ is the number of arcs in a minimum base graph with respect to $\mathcal{C}$.

CONJECTURE 7.1: *Suppose T is a regular tournament on n vertices that admits a level-two stable coloring. Then for any level-two stable coloring of T, $C_1^{v_1}, \cdots, C_m^{v_n}$, there is a C_j^i, such that $\beta(C_j^i) = O(n^c)$, where $c < 2$ is a constant independent of T.*

Intuitively, the coherence requirement should already make it difficult for all the C_j^i to have large minimum base graphs, if they could all have level-two stable colorings at all. The conjecture implies that our deterministic algorithm factor a polynomial of degree n over $\mathbf{F}_p$ completely within polynomial time, since

$$n \times \frac{n}{2} \times n^d \times \frac{n^d}{2} \times n^{d^2} \times \cdots \leq n^{2(1+d+d^2+\cdots)} = n^{O(1)},$$

where $d = \frac{c}{2}$.

The fact that there is no triply-regular tournament with more than three vertices and the following observation by Babai [2] provide strong evidences for the conjecture.

PROPOSITION 7.1: *Let T be a vertex-transitive tournament with $n > 1$ vertices. Let v_0 be a vertex of T. Then for every vertex $v_1 \neq v_0$ there exists a vertex $v_2 \neq v_0, v_1$ such that the size of the orbit of the pair (v_1, v_2) in the stabilizer of v_0 is at most $(n-1)/2$.*

Another way to improve our results is to look at the case when we have a lot of arc colors.

Definition: We call a tournament with n vertices transitive if there is a linear order of its vertices, $v_1, v_2, \cdots, v_n$, such that for any i and color α, if v_i α-dominates v_{i+1}, then v_i α-dominates v_j for any $j > i$.

Denote $\delta(p)$ be the size of largest transitive subgraph in a cyclotomic tournament. Heuristically when the number of colors gets bigger, $\delta(p)$ should become smaller, even down to a constant. One can for example proves that a random tournament with n vertices and n^c $(c < 1)$ colors has only constant size transitive subtournament. It follows easily from our algorithm that the polynomial in $\mathbf{F}_p$ can be factored completely in time $P(n^{\delta(p)}, \log p)$, where P is a polynomial function.

References

1. Annie Astie-Vida and Vincent Dugat. Autonomous parts and decomposition of regular tournaments. *Discrete Mathematics*, 111:27–36, 1993.

2. L. Babai. personal communication, 1998.

3. L. Babai and L. Kucera. Canonical labeling of graphs in linear average time. In *Proc. 20th IEEE Symp. on Foundations of Comp. Science*, pages 39–46, 1979.

4. Eric Bach and Jeffrey Shallit. *Algorithmic Number theory*, volume I. The MIT Press, 1996.

5. J.L. Berggren. An algebraic characterization of finite symmetric tournaments. *Bull. Austral. Math. Soc.*, 6:53–59, 1972.

6. E. R. Berlekamp. Factoring polynomials over large finite fields. *Math. Comp.*, 24:713–735, 1970.

7. Qi Cheng and Fang Fang. Kolmogorov random graphs only have trivial stable colorings. *Information Processing Letters*, 81(3):133–136, 2002.

8. Sergei Evdokimov. Factorization of polynomials over finite field in subexponential time under erh. In *ANTS*, volume 877 of *Lecture Notes in Computer Science*, pages 209–219, 1994.

9. P. Frankl, V. Rodl, and R.M. Wilson. The number of submatrics of a given type in a hadamard matrix and related results. *J. of Combinatorical Theory, Series B*, 44, 1988.

10. Shuhong Gao. On the deterministic complexity of polynomial factoring. *J. Symbolic Computation*, 31:19–36, 2001.

11. Vladimir Muller and Jan Pelant. On strongly homogeneous tournaments. *Czechoslovak Mathematical Journal*, 24(99):379–391, 1974.

12. M. O. Rabin. Probabilistic algorithms in finite fields. *Siam J. Computing*, 9:128–138, 1980.

13. K.B. Reid and L. W. Beineke. *Selected Topics in Graph Theory*, chapter Tournaments. Academic Press., 1978.

14. Lajos Ronyai. Factoring polynomials over finite fields. In *Proc. 19th ACM Symp. on Theory of Computing*, pages 132–137, 1987.