Efficient multivariate low-degree tests via interactive oracle proofs of proximity for polynomial codes

Daniel Augot[1,2], Sarah Bordage[2,1], and Jade Nardi[1,2]

[1] Inria
[2] LIX, CNRS UMR 7161, École polytechnique, Institut polytechnique de Paris
{daniel.augot,sarah.bordage,jade.nardi}@inria.fr

Abstract. We consider the proximity testing problem for error-correcting codes which consist in evaluations of multivariate polynomials either of bounded individual degree or bounded total degree. Namely, given an oracle function $f : L^m \to \mathbb{F}_q$, where $L \subset \mathbb{F}_q$, a verifier distinguishes whether f is the evaluation of a low-degree polynomial or is far (in relative Hamming distance) from being one, by making only a few queries to f. This topic has been studied in the context of locally testable codes, interactive proofs, probabilistically checkable proofs, and interactive oracle proofs. We present the first interactive oracle proofs of proximity (IOPP) for tensor products of Reed-Solomon codes (evaluation of polynomials with bounds on individual degrees) and for Reed-Muller codes (evaluation of polynomials with a bound on the total degree).

Such low-degree polynomials play a central role in constructions of probabilistic proof systems and succinct non-interactive arguments of knowledge with zero-knowledge. For these applications, highly-efficient multivariate low-degree tests are desired, but prior probabilistic proofs of proximity required super-linear proving time. In contrast, for multivariate codes of length N, our constructions admit a prover running in time linear in N and a verifier which is logarithmic in N.

For fixed constant number of variables m, the efficiency parameters of our IOPPs for multivariate codes compare well, all things equal, with those of the IOPP for Reed-Solomon codes of [Ben-Sasson *et al.*, ICALP 2018] from which they are directly inspired.

Keywords: Algebraic coding theory · Reed-Solomon codes · Product codes · Reed-Muller codes · Low degree testing · Interactive proof systems.

This is an extended abstract of an article submitted to the journal Designs, Codes and Cryptography *in August 2021. The full version of the paper is available online [1].*

1 Introduction

Let $\mathbb{F}_q$ be a finite field of size q. Any function $f : \mathbb{F}_q^m \to \mathbb{F}_q$ can be written as a polynomial of individual degrees at most $q-1$, hence a polynomial of total degree

$\leqslant m(q-1)$. The problem of *low-degree testing* can be formulated as follows. Given a proximity parameter $\delta \in (0,1)$ and oracle access to a function $f : \mathbb{F}_q^m \to \mathbb{F}_q$ (as a table of values), check with a few queries whether f is is a polynomial function of low degree compared to q, or δ-far in relative Hamming distance from being low-degree. The main focus of this paper is the problem of low-degree testing applied to a function $f : L^m \to \mathbb{F}_q$ with $L \subset \mathbb{F}_q$. Multivariate low-degree tests fall into two flavors. Depending on whether one requires a bound on the total degree or the individual degree, a multivariate low-degree test is either a proximity test for a Reed-Muller code or the m-wise tensor product of a Reed-Solomon code.

Low-degree tests have been the subject of a substantial body of research during the past four decades. Indeed, design and better analysis of low-degree tests have gone hand in hand with the construction of efficient probabilistically checkable proofs (PCPs), interactive proofs (IPs) and locally testable codes (LTCs). One motivation for designing probabilistic proof systems with low communication complexity, fast generation and sublinear verification is the application to verifiable computation. In [2], the authors point out that a subsequent bottleneck of PCP-based proof systems is that of computing solutions to the low-degree testing problem for multivariate polynomials. A few years ago, [6,15] introduced interactive oracle proofs (IOPs), which generalize both PCPs, IPs and interactive PCPs [13] and open a new large design space. On the contrary of known PCPs constructions, it turns out that the IOP model enable the design of proofs systems that are efficient enough for practical applications of zero-knowledge proofs and schemes for delegated computation. Interactive oracle proofs of proximity (IOPP) are the natural generalization of probabilistically checkable proofs of proximity (PCPP) [12,8] to the IOP model (see Definition 1). Several of constructions, including [3,5,14,11], crucially rely on a *prover-efficient* IOPP for Reed-Solomon codes which the authors of [2] named FRI protocol. Improved soundness analysis of the FRI protocal appear in subsequent works [9,7,4]. While multivariate low degree tests have been extensively studied in the PCPP model, they have not been the subject of any specific constructions in the IOPP model.

1.1 Our contributions

We propose two constructions: the first is an IOPP for the tensor product of Reed-Solomon codes, the second an IOPP for Reed-Muller codes. The alphabets $\mathbb{F}_q$ which we consider admit either smooth multiplicative subgroups or smooth affine subspaces, where smooth means that the size of the set is a power of a small fixed integer. Our two IOPPs are generalizations of the FRI protocol [2] to the multivariate case. If m is a constant, they have strictly linear-time prover and strictly logarithmic-time verifier (with respect to the blocklength $|L|^m$ of the code). In particular, query complexity is logarithmic in the degree bound d.

Previous low-degree tests required the verifier to query a number of field elements linear in d. Since our constructions are explicit, all efficiency measures of the two IOPPs are explicitly presented. These parameters match the IOPP for Reed-Solomon codes of [2], from which they are inspired. Concerning applications to IOP constructions, having a constant number of variables m can be relevant. Indeed, linear-size IOPs have already been constructed from m-wise tensor product codes [10] and m were a fixed integer there.

In the case of tensor product of Reed-Solomon codes, our construction can be generalized to distinct degree bounds and different evaluation domain. For Reed-Muller codes and unlike previous works, we are able to consider a support L^m where $L \subset \mathbb{F}_q$ can can be much smaller than $\mathbb{F}_q$. We think that allowing smaller support might give more flexibility in the design of proof systems.

2 Notations and definitions

2.1 Notations

We denote by $\mathbb{F}_q$ the finite field of size q and by $\mathbb{F}_q^\times$ the multiplicative group of $\mathbb{F}_q$. We use the notation $[a \mathinner{..} b]$ for the set of integers $\{a, a+1, \dots, b\}$. Let $m \geqslant 1$ be an integer. Vectors are written in bold, and for two tuples $\boldsymbol{x} = (x_1, \dots, x_m)$ and $\boldsymbol{u} = (u_1, \dots, u_m)$, $\boldsymbol{x}^{\boldsymbol{u}}$ refers to $\boldsymbol{x}^{\boldsymbol{u}} := x_1^{u_1} \cdots x_k^{u_k}$. Writing $\boldsymbol{X} = (X_1, \dots, X_m)$, $\mathbb{F}_q[\boldsymbol{X}]$ refers to the ring of polynomials in the indeterminates $X_1, \dots, X_m$. For a polynomial $P \in \mathbb{F}_q[\boldsymbol{X}]$, its total degree is denoted by $\deg P$, and $\deg_{X_j} P$ is the degree of P with respect to X_j. The individual degree of P is the maximum of the degrees in each variable.

The Hamming weight of a vector $\boldsymbol{u} \in \mathbb{F}_q^n$ will be denoted by $w_H(\boldsymbol{u})$ and the relative Hamming distance between two vectors $\boldsymbol{u}, \boldsymbol{u'} \in \mathbb{F}_q^n$ by $\Delta(\boldsymbol{u}, \boldsymbol{u'})$. Given $\boldsymbol{u} \in \mathbb{F}_q^n$ and a code $C \subseteq \mathbb{F}_q^n$, we define $\Delta(\boldsymbol{u}, C)$ to be the minimal distance between $\boldsymbol{u}$ and any codeword of C. If $\Delta(\boldsymbol{u}, C) > \delta$, we say that $\boldsymbol{u}$ is δ-far from C, otherwise $\boldsymbol{u}$ is δ-close to C. We will consider *evaluation codes*. In this setting, we view codewords as functions in $\mathbb{F}_q^D$ for a certain domain D, and for $f \in C$ and $x \in D$, $f(x)$ naturally denotes the x-entry of the codeword f.

2.2 Definitions

An IOPP $(\mathcal{P}, \mathcal{V})$ for a code $C \subset \mathbb{F}^D$ is a pair of probabilistic algorithms, $\mathcal{P}$ is designated as *prover* and $\mathcal{V}$ as *verifier*. The IOPP $(\mathcal{P}, \mathcal{V})$ has round complexity $r(n)$ if the prover and the verifier interact over at most $r(n)$ rounds. At each round, the verifier sends a message to the prover, and the prover answers with an oracle. We denote by $\langle \mathcal{P} \leftrightarrow \mathcal{V} \rangle \in \{\mathsf{accept}, \mathsf{reject}\}$ the output of $\mathcal{V}$ after interacting with $\mathcal{P}$. The notation $\mathcal{V}^f(C)$ means that f is given as an oracle input to $\mathcal{V}$, while $\mathcal{P}(C, f)$ means that the prover has excess to full codeword. Both know the code C.

Definition 1 (IOPP for a code C). *We say that a pair of probabilistic algorithms $(\mathcal{P}, \mathcal{V})$ is an IOPP system for a code C with* soundness error $s : (0, 1] \to [0, 1]$ *if the following two conditions hold:*

Perfect completeness: *If $f \in C$, then* $\Pr[\langle \mathcal{P}(C, f) \leftrightarrow \mathcal{V}^f(C) \rangle = \mathsf{accept}] = 1$.

Soundness: *For any function $f \in \mathbb{F}_q^D$ such that $\delta := \Delta(f, C) > 0$ and any unbounded malicious prover $\mathcal{P}^*$,* $\Pr[\langle \mathcal{P}^* \leftrightarrow \mathcal{V}^f(C) \rangle = \mathsf{accept}] \leqslant s(\delta)$.

The IOPP is *public-coin* if verifier's messages are generated by public randomness and queries are performed after the end of the interaction with the prover. Relevant measures for an IOPP system are the following. The alphabet of the IOPP we consider will be a finite field $\mathbb{F}_q$. The total number of field elements of all the oracles built by the prover during the interaction is the proof length $l(n)$ of the IOPP. The query complexity $q(n)$ is the total number of symbols queried by the verifier to both the purported codeword f and the oracles sent by the prover during the interaction. We consider arithmetic complexities, and we assume each arithmetic operation performed in $\mathbb{F}_q$ takes constant time. The prover complexity $t_p(n)$ is the time needed to generate prover messages. The verifier complexity $t_v(n)$ is the time spent by the verifier to make her decision when queries and query-answers are given as inputs.

Definition 2 (Reed-Solomon code). *Given $L \subseteq \mathbb{F}_q$ and $k \leqslant |L|$, we denote by $\mathsf{RS}[\mathbb{F}_q, L, k]$ the Reed-Solomon (RS) code over alphabet $\mathbb{F}_q$ defined by*

$$\mathsf{RS}[\mathbb{F}_q, L, k] := \left\{ f \in \mathbb{F}_q^L \mid \exists P \in \mathbb{F}_q[X], \deg P < k \textit{ s.t. } \forall x \in L,\, f(x) = P(x) \right\}.$$

Definition 3 (Tensor product of Reed-Solomon code). *Given $L \subset \mathbb{F}_q$, and $m, k \geqslant 1$, such that $k \leqslant |L|$, we denote by $(\mathsf{RS}[\mathbb{F}_q, L, k])^{\otimes m}$ the m-wise tensor product of the code $\mathsf{RS}[\mathbb{F}_q, L, k]$. Equivalently,*

$$(\mathsf{RS}[\mathbb{F}_q, L, k])^{\otimes m} := \Big\{ f \in \mathbb{F}_q^{L^m} \mid \exists P \in \mathbb{F}_q[\boldsymbol{X}], \deg_{X_i} P < k, i \in [1 \,..\, m], \textit{ such that } \forall x \in L,\, f(\boldsymbol{x}) = P(\boldsymbol{x}) \Big\}. \quad (1)$$

The tensor product code $(\mathsf{RS}[\mathbb{F}_q, L, k])^{\otimes m}$ has length $|L|^m$, dimension k^m, rate $\left(\frac{k}{|L|}\right)^m$ and relative distance $\left(1 - \frac{k-1}{|L|}\right)^m$.

Reed-Muller codes consist of evaluation of multivariate polynomials with coefficients in $\mathbb{F}_q$ of bounded total degree. We are interested of (punctured) Reed-Muller codes with support $L^m \subset \mathbb{F}_q^m$, where L may be much smaller than $\mathbb{F}_q$. Since this setting is not commonly encountered in coding theory, we introduce the non-standard term *short Reed-Muller codes* to emphasize this fact.

Definition 4 (Short Reed-Muller code). *A* short Reed-Muller code *with support $L^m \subset \mathbb{F}_q^m$ is defined as follows*

$$\mathsf{SRM}[\mathbb{F}_q, L, m, k] := \left\{ f \in \mathbb{F}_q^{L^m} \mid \exists P \in \mathbb{F}_q[\boldsymbol{X}], \deg P < k \textit{ s.t. } \forall \boldsymbol{x} \in L^m,\, f(\boldsymbol{x}) = P(\boldsymbol{x}) \right\}.$$

If $k \leqslant |L|$, the evaluation map from the space of multivariate polynomials of total degree less than k to the space of functions $\mathbb{F}_q^{L^m}$ is injective, thus the dimension of $\mathsf{SRM}[\mathbb{F}_q, L, m, k]$ is $\binom{m+k-1}{m}$. A bound on the minimum distance of $\mathsf{SRM}[\mathbb{F}_q, L, m, k]$ follows from the Schwartz-Zippel lemma [17,16], which states that any non-zero multivariate polynomial $P \in \mathbb{F}_q[\boldsymbol{X}]$ of total degree less than q cannot vanish in more than $\frac{\deg P}{|L|}$ fraction of L^m. The code $\mathsf{SRM}[\mathbb{F}_q, L, m, k]$ has length $|L^m|$, rate $\binom{m+k-1}{m} |L|^{-m}$ and relative distance at least $1 - \frac{k-1}{|L|}$.

Proposition 1 (Low-degree extension). *Let $H_1, \ldots, H_m \subseteq \mathbb{F}_q$ and let $f : H_1 \times \cdots \times H_m \to \mathbb{F}_q$ be a function. Then there exists a unique polynomial $\widehat{f}$ in m variables over $\mathbb{F}_q$ such that :*

1. *$\widehat{f}$ has degree $\deg_{X_i} \widehat{f} < |H_i|$ in its i-th variable,*
2. *$\widehat{f}$ agrees with f on $H_1 \times \cdots \times H_m$.*

The polynomial $\widehat{f}$ is referred to as the low-degree extension *of the function f (with respect to $\mathbb{F}_q$ and $H_1, \ldots, H_m$).*

3 Technical overview

In this section, we present a technical overview of our constructions. For the sake of brevity, we do not provide proofs of the presented results. We refer the interested reader to the full version of the paper for detailed proofs. Moreover, we restrict ourselves to the following setting. Assume that $\mathbb{F}_q$ is a prime field and $q-1$ is divisible by a power of two, i.e. $q = a \cdot 2^n + 1$ for some positive integers a and n. We will consider $L_0 \subset \mathbb{F}_q$ a cyclic multiplicative group of order 2^n. For any integer r, we define a sequence of evaluation sets $(L_i)_{0 \leqslant i \leqslant r}$ as: $L_{i+1} := q(L_i)$ where $q(X) = X^2$. Our target degree bound for low-degree testing will be k_0 a power of 2, $k_0 < |L_0|$. For $r = \log k_0$, denote by RS_i^m the code $\mathsf{RS}\left[\mathbb{F}_q, L_i, k_i\right]^{\otimes m}$ with $k_i = k_0/2^i$. Similarly, set $\mathsf{SRM}_i := \mathsf{SRM}\left[\mathbb{F}_q, L_i, m, k_i\right]$.

Our IOPP have r rounds of interactions, where in each round, the problem of proximity testing to a code RS_i^m (resp. SRM_i) is reduced to a problem of size 2^m times smaller, namely proximity testing to RS_{i+1}^m (resp. SRM_{i+1}).

The observation which forms the basis of our constructions is the following decomposition of multivariate polynomials.

Proposition 2 (Multivariate decomposition). *For every $\widehat{f} \in \mathbb{F}_q[\boldsymbol{X}]$ there exists a unique sequence $(\widehat{g}_{\boldsymbol{e}})_{\boldsymbol{e} \in [0\,..\,l-1]^m}$ of polynomials in $\mathbb{F}_q[\boldsymbol{X}]$ such that*

$$\widehat{f}(\boldsymbol{X}) = \sum_{\boldsymbol{e} \in [0\,..\,l-1]^m} \boldsymbol{X}^{\boldsymbol{e}} \widehat{g}_{\boldsymbol{e}}\left(X_1^2, \ldots, X_m^2\right), \tag{2}$$

and

- *for all $\boldsymbol{e} \in [0\,..\,l-1]^m$ and $j \in [1\,..\,m]$, $\deg_{X_j} \widehat{g}_{\boldsymbol{e}} \leqslant \left\lfloor \frac{\deg_{X_j} \widehat{f}}{l} \right\rfloor$,*
- *for all $\boldsymbol{e} \in [0\,..\,l-1]^m$, $\deg \widehat{g}_{\boldsymbol{e}} \leqslant \left\lfloor \frac{\deg \widehat{f} - w_H(\boldsymbol{e})}{l} \right\rfloor$.*

3.1 IOPP for tensor product of Reed-Solomon codes

For each code RS_i^m, $0 \leqslant i < r$, we define a family of folding operators satisfying three key properties: *completeness*, *local computability* and *distance preservation.*

Definition 5 (Folding operators). *Let $i \in [0, r-1]$. Let $f : L_i^m \to \mathbb{F}_q$ be an arbitrary function and let $\widehat{f}$ be its low-degree extension. Let $(\widehat{g}_{\boldsymbol{e}})_{\boldsymbol{e} \in \{0,1\}^m}$ be the 2^m m-variate polynomials provided by Proposition 2 applied to $\widehat{f}$. We consider their evaluations on L_{i+1}^m, respectively denoted by $g_{\boldsymbol{e}}$. For any $\boldsymbol{p} \in \mathbb{F}_q^m$, we define the* folding of f ***Fold***$[f, \boldsymbol{p}]$ *as the following function:*

$$\textbf{\textsf{Fold}}[f, \boldsymbol{p}] : \begin{cases} L_{i+1}^m \to & \mathbb{F}_q, \\ \boldsymbol{y} \ \mapsto & \sum\limits_{\boldsymbol{e} \in \{0,1\}^m} \boldsymbol{p}^{\boldsymbol{e}} g_{\boldsymbol{e}}(\boldsymbol{y}). \end{cases} \tag{3}$$

Proposition 3 (Key properties of folding operators). *Definition 5 satisfy the following properties.*

1. Completeness: *For any $\boldsymbol{p} \in \mathbb{F}_q^m$, if $f \in \mathsf{RS}_i^m$, then* ***Fold***$[f, \boldsymbol{p}] \in \mathsf{RS}_{i+1}^m$.
2. Local computability: *Let $f : L_i^m \to \mathbb{F}_q$ be an arbitrary function and let $\boldsymbol{p} \in \mathbb{F}_q^m$. The value of* ***Fold***$[f, \boldsymbol{p}]$ *at any $\boldsymbol{y} \in L_{i+1}^m$ can be computed with exactly 2^m queries to f.*
3. Distance preservation: *Let $f_i : L_i^m \to \mathbb{F}_q$ be an arbitrary function. Let $\varepsilon \in \left(0, \frac{1}{3}\right)$ and $\delta < 1 - (1 - \lambda + \varepsilon)^{\frac{1}{3}}$. If $\Delta(f, \mathsf{RS}_i^m) > \delta$, then*

$$\Pr_{\boldsymbol{p} \in \mathbb{F}_q^m} \left[\Delta(\textbf{\textsf{Fold}}[f, \boldsymbol{p}], \mathsf{RS}_{i+1}^m) < \delta - m\varepsilon \right] < \frac{2m}{\varepsilon^2 q}.$$

The completeness property follows directly from the decomposition of bounded degree multivariate polynomials. The local computability is obtained by proving that for any $\boldsymbol{y} \in L_{i+1}^m$, the value **Fold** $[f, \boldsymbol{p}]$ $(\boldsymbol{y})$ corresponds to the evaluation at $\boldsymbol{z} \in \mathbb{F}^m$ of the polynomial of degree 1 in each variable which interpolates the set of points $\left\{(\boldsymbol{x}, f(\boldsymbol{x})) \mid (x_1^2, \ldots, x_m^2) = \boldsymbol{y}\right\}$. The distance preservation property is a consequence of Proposition 4, which states the following. Given $(u_{\boldsymbol{e}})_{\boldsymbol{e} \in \{0,1\}^m}$ where $u_{\boldsymbol{e}} \in \mathbb{F}^D$, it suffices that enough elements in the set of multilinear combinations of $\boldsymbol{u}$ are δ-close to a code $C \subset \mathbb{F}^D$ to deduce that each $u_{\boldsymbol{e}}$ is δ'-close to C with $\delta' = \delta + o(m)$. In other words, there exists a non trivial subset of locations $T \subset D$ such that the restriction of each function $u_{\boldsymbol{e}}$ to T is a codeword of the code $C_{|T} = \{c_{|T} : T \to \mathbb{F}; c \in C\}$. Proposition 4 can be proved by induction on the number of variables m, where the base case $m = 1$ is dealt with in [7, Lemma 3.2].

Proposition 4 (Correlated agreement). *Let m be a positive integer. Let $C \subset \mathbb{F}_q^D$ be a linear code of relative distance $\lambda = \Delta(C)$. Let $\varepsilon, \delta > 0$ such that $\varepsilon < 1/3$ and*

$$\delta < 1 - (1 - \lambda + \varepsilon)^{1/3}. \tag{4}$$

Let $\boldsymbol{u} = (u_{\boldsymbol{e}})_{\boldsymbol{e} \in \{0,1\}^m}$ such that

$$\Pr_{\boldsymbol{p} \in \mathbb{F}_q^m} \left[\Delta\left(\sum_{\boldsymbol{e} \in \{0,1\}^m} \boldsymbol{p}^{\boldsymbol{e}} u_{\boldsymbol{e}}, C \right) < \delta \right] \geqslant \frac{2m}{\varepsilon^2 q}. \tag{5}$$

Then there exist $T \subset D$ and a family $\boldsymbol{v} = (v_{\boldsymbol{e}})_{\boldsymbol{e} \in \{0,1\}^m} \in C^{2^m}$ such that

– $|T| \geqslant (1-\delta-m\varepsilon)\,|D|$,
– *for each* $\boldsymbol{e} \in \{0,1\}^m$, $u_{\boldsymbol{e}|T} = v_{\boldsymbol{e}|T}$.

Given a sequence of codes $(\mathsf{RS}_i^m)_{0\leqslant i\leqslant r}$ and a family of folding operators for each code RS_i^m, we present the construction of a public-coin IOPP $(\mathcal{P}_{\mathsf{RS}^m}, \mathcal{V}_{\mathsf{RS}^m})$ for the code RS_0^m in Figure 1. As in the FRI protocol [2], our protocol is divided into two phases. The COMMIT phase is an interaction over r rounds between a prover $\mathcal{P}$ and a verifier $\mathcal{V}$. At each round i, the verifier samples a random element $\boldsymbol{p}_i \in \left(\mathbb{F}_q^m\right)^t$. The prover answers with an oracle function $f_{i+1} : L_i^m \to \mathbb{F}_q$, which is expected to coincide with $\mathbf{Fold}\left[f_i, \boldsymbol{p}_i\right]$. The second phase is run only by the verifier $\mathcal{V}$, and is called QUERY phase. The task of $\mathcal{V}$ is to check whether each pair of oracle functions (f_i, f_{i+1}) is consistent and to perform a membership test to the last code. The local computability property of the folding operators enables both the prover to run in linear with respect to $|L_0^m|$ and the verifier to perform the consistency checks with two queries to each oracle.

```
COMMIT Phase: (interactive)
 1. For each round $i$ from 0 to $r-1$ :
    (a) Verifier $\mathcal{V}$ picks uniformly at random an element $\boldsymbol{p}_i \in (\mathbb{F}_q^m)^t$;
    (b) Verifier $\mathcal{V}$ sends $\boldsymbol{p}_i$ to Prover $\mathcal{P}$;
    (c) An honest Prover $\mathcal{P}$ computes $\mathbf{Fold}\left[f_i, \boldsymbol{p}_i\right] : L_{i+1}^m \to \mathbb{F}_q$
QUERY Phase: (run by $\mathcal{V}$ only)
 1. Repeat $\alpha$ times the following query test:
    (a) Sample $\boldsymbol{y}_0 \in L_0^m$ uniformly at random;
    (b) For $i = 0$ to $r-1$:
         i. Define $\boldsymbol{y}_{i+1} \in L_{i+1}^m$ as $\boldsymbol{y}_{i+1} = \pi_i(\boldsymbol{y}_i)$;
        ii. Query $f_i$ on $S_{\boldsymbol{y}_{i+1}}$ of size $l_i$ to compute $\mathbf{Fold}\left[f_i, \boldsymbol{p}_i\right](\boldsymbol{y}_{i+1})$;
       iii. Query $f_{i+1}(\boldsymbol{y}_{i+1})$;
        iv. If $f_{i+1}(\boldsymbol{y}_{i+1}) \neq \mathbf{Fold}\left[f_i, \boldsymbol{p}_i\right](\boldsymbol{y}_{i+1})$, outputs reject
 2. Outputs acccept if and only if $f_r \in C_r$
```

Fig. 1. Construction of an IOPP based on folding operators

The properties of the resulting IOPP system $(\mathcal{P}_{\mathsf{RS}^m}, \mathcal{V}_{\mathsf{RS}^m})$ are displayed in Theorem 1.

Theorem 1. *Let k, m be positive integers. Assume k is a power of two and $L \subset \mathbb{F}_q^\times$ is a 2-smooth coset of a multiplicative subgroup or additive subgroup of $\mathbb{F}_q$. Then, there exists a public-coin IOPP system $(\mathcal{P}_{\mathsf{RS}^m}, \mathcal{V}_{\mathsf{RS}^m})$ for the tensor product code $(\mathsf{RS}\left[\mathbb{F}_q, L, k\right])^{\otimes m}$ of blocklength n^m with the following properties.*

1. **Round complexity** *is* $r(n^m) < \log n$.
2. **Query complexity** *is* $q(n^m) < \alpha 2^m \log n + 1$ *for α repetitions of the QUERY phase.*

3. **Proof length** *is* $l(n^m) < \frac{n^m}{2^m-1}$.
4. **Prover complexity** *is* $t_p(n^m) < 4(m+2)n^m$.
5. **Verifier decision complexity** *is* $t_v(n^m) < 4\alpha(2^m+m)\log n$.
6. **Perfect completeness:** *If* $f \in (\mathsf{RS}[\mathbb{F}_q, L, k])^{\otimes m}$ *and if the oracles* $f_1, \ldots f_r$ *are computed by an honest prover* $\mathcal{P}_{\mathsf{RS}^m}$*, then* $\mathcal{V}_{\mathsf{RS}^m}$ *outputs* ***accept*** *with probability 1.*
7. **Soundness:** *Assume that* $f : L^m \to \mathbb{F}_q$ *is* δ*-far from* $(\mathsf{RS}[\mathbb{F}_q, L, k])^{\otimes m}$*. Denote* λ *the relative minimum distance of* $(\mathsf{RS}[\mathbb{F}_q, L, k])^{\otimes m}$ *and, for any* $\varepsilon \in \left(0, \frac{1}{3}\right)$*, set* $\gamma(\lambda, \varepsilon) := 1 - (1 - \lambda + \varepsilon)^{1/3}$*. Then, for any unbounded prover* $\mathcal{P}^*$*, the verifier* $\mathcal{V}_{\mathsf{RS}^m}$ *outputs* **accept** *after* α *repetitions of the* QUERY *phase with probability at most*

$$\frac{2m\log n}{\varepsilon^2 q} + (1 - \min(\delta, \gamma(\varepsilon, \lambda)) + \varepsilon m \log n)^{\alpha}.$$

3.2 IOPP for Short Reed-Muller codes

The IOPP for Short Reed-Muller codes is obtained following the blueprint we used for the individual degree case. However, the total degrees of the polynomials appearing in the decomposition of Proposition 2 range from $\left\lfloor \frac{\deg f - m}{2} \right\rfloor$ to $\left\lfloor \frac{\deg f}{2} \right\rfloor$. Contrasting with foldings operators for bounded individual degree polynomials (and univariate polynomials), that area of variation needs to be taken into account to ensure both completeness and distance preservation properties.

Definition 6 (Balancing functions). *Let* $i \in [0 \mathinner{..} r-1]$*. For any* $\boldsymbol{e} \in \{0,1\}^m$*, we call a* balancing function *any map* $h_{\boldsymbol{e}} : L_{i+1}^m \to \mathbb{F}_q$ *which corresponds to the evaluation of a* m*-variate multilinear monic monomial* $\widehat{h}_{\boldsymbol{e}}$ *of total degree exactly* $\left\lfloor \frac{w_H(\boldsymbol{e})}{2} \right\rfloor$*. We call* $(h_{\boldsymbol{e}})_{\boldsymbol{e}\in\{0,1\}^m}$ *a* balancing tuple for the code SRM_{i+1}.

Definition 7 (Folding operator). *Let* $i \in [0, r-1]$*. Let* $(h_{\boldsymbol{e}})_{\boldsymbol{e}\in\{0,1\}^m}$ *be a balancing tuple for* SRM_{i+1} *and let* $f : L_i^m \to \mathbb{F}_q$ *be an arbitrary function. Let* $(\widehat{g}_{\boldsymbol{e}})_{\boldsymbol{e}\in\{0,1\}^m}$ *be the* 2^m m*-variate polynomials provided by Proposition 2 applied to* $\widehat{f}$*. We consider their evaluations on* L_{i+1}^m*, respectively denoted by* $g_{\boldsymbol{e}}$*. For any* $(\boldsymbol{p}, \boldsymbol{p}') \in \left(\mathbb{F}_q^m\right)^2$*, we define the* folding of f *as the function* $\textbf{\textit{Fold}}[f, (\boldsymbol{p}, \boldsymbol{p}')] : L_{i+1}^m \to \mathbb{F}_q$ *such that*

$$\textbf{\textit{Fold}}\left[f, (\boldsymbol{p}, \boldsymbol{p}')\right](\boldsymbol{y}) = \sum_{\boldsymbol{e}\in\{0,1\}^m} \boldsymbol{p}^{\boldsymbol{e}} g_{\boldsymbol{e}}(\boldsymbol{y}) + \sum_{\substack{\boldsymbol{e}\in\{0,1\}^m \\ \boldsymbol{e}\neq\boldsymbol{0}}} \boldsymbol{p}'^{\boldsymbol{e}} h_{\boldsymbol{e}}(\boldsymbol{y}) g_{\boldsymbol{e}}(\boldsymbol{y}). \tag{6}$$

With folding operators defined as per Definition 7, we are able to prove the analogous of Proposition 3 for Short Reed-Muller codes. Plugging them into the protocol depicted in Figure 1 leads to the following theorem.

Theorem 2. *Let* k, m *be positive integers. Assume* k *is a power of two and* $L \subset \mathbb{F}_q$ *is a* 2*-smooth coset of a multiplicative subgroup or additive subgroup of*

$\mathbb{F}_q$. There exists a public-coin IOPP *system* $(\mathcal{P}_{\mathsf{RM}}, \mathcal{V}_{\mathsf{RM}})$ *testing proximity of a function* $f : L^m \to \mathbb{F}_q$ *to the short Reed-Muller code* $\mathsf{SRM}\left[\mathbb{F}_q, L, m, k\right]$ *with the following properties:*

1. **Round complexity** *is* $r(n^m) < \log n$.
2. **Query complexity** *is* $q(n^m) < \alpha(2^m \log n + 1)$ *for a* QUERY *phase with repetition parameter* α.
3. **Proof length** *is* $l(n^m) < \frac{n^m}{(2^m-1)}$.
4. **Prover complexity** *is* $t_p(n^m) < \left(\frac{5}{2}m + 14\right) n^m$.
5. **Verifier decision complexity** *is* $t_v(n^m) < \alpha 2^m \left(\frac{5}{4}m + 7\right) \log n$.
6. **Perfect completeness:** *If* $f \in \mathsf{SRM}\left[\mathbb{F}_q, L, m, k\right]$ *and if the oracles* $f_1, \ldots f_r$ *are computed by an honest prover, then* $\mathcal{V}_{\mathsf{RM}}$ *outputs* ***accept*** *with probability 1.*
7. **Soundness:** *Assume that* $f : L^m \to \mathbb{F}_q$ *is* δ*-far from* $\mathsf{SRM}\left[\mathbb{F}_q, L, m, k\right]$. *Denote* $\lambda = 1-2\frac{k}{|L|}$. *For any* $\varepsilon \in (0, \frac{2}{3})$, *set* $\gamma(\varepsilon, \lambda) := \min\left(1 - (1 - \lambda + \varepsilon)^{1/3}, \frac{1}{2}(\lambda + m\frac{\varepsilon}{2})\right)$. *Then, for any unbounded prover* $\mathcal{P}^*$, *the verifier* $\mathcal{V}$ *outputs* accept *after* α *repetitions of the* QUERY *phase with probability at most*

$$r\frac{16m}{\varepsilon^2 q} + (1 - \min(\delta, \gamma(\varepsilon, \lambda)) + rm\varepsilon)^{\alpha}.$$

References

1. Augot, D., Bordage, S., Nardi, J.: Efficient multivariate low-degree tests via interactive oracle proofs of proximity for polynomial codes. Electron. Colloquium Comput. Complex. p. 118 (2021), `https://eccc.weizmann.ac.il/report/2021/118`
2. Ben-Sasson, E., Bentov, I., Horesh, Y., Riabzev, M.: Fast reed-solomon interactive oracle proofs of proximity. In: 45th International Colloquium on Automata, Languages, and Programming, ICALP 2018, July 9-13, 2018, Prague, Czech Republic. pp. 14:1–14:17 (2018)
3. Ben-Sasson, E., Bentov, I., Horesh, Y., Riabzev, M.: Scalable zero knowledge with no trusted setup. In: Boldyreva, A., Micciancio, D. (eds.) Advances in Cryptology - CRYPTO 2019 - 39th Annual International Cryptology Conference, Santa Barbara, CA, USA, August 18-22, 2019, Proceedings, Part III. Lecture Notes in Computer Science, vol. 11694, pp. 701–732. Springer (2019)
4. Ben-Sasson, E., Carmon, D., Ishai, Y., Kopparty, S., Saraf, S.: Proximity gaps for reed-solomon codes. In: 61st IEEE Annual Symposium on Foundations of Computer Science, FOCS 2020, Durham, NC, USA, November 16-19, 2020. pp. 900–909. IEEE (2020)
5. Ben-Sasson, E., Chiesa, A., Riabzev, M., Spooner, N., Virza, M., Ward, N.P.: Aurora: Transparent succinct arguments for R1CS. In: Ishai, Y., Rijmen, V. (eds.) Advances in Cryptology - EUROCRYPT 2019 - 38th Annual International Conference on the Theory and Applications of Cryptographic Techniques, Darmstadt, Germany, May 19-23, 2019, Proceedings, Part I. Lecture Notes in Computer Science, vol. 11476, pp. 103–128. Springer (2019)
6. Ben-Sasson, E., Chiesa, A., Spooner, N.: Interactive oracle proofs. In: Theory of Cryptography - 14th International Conference, TCC 2016-B, Beijing, China, October 31 - November 3, 2016, Proceedings, Part II. pp. 31–60 (2016)

7. Ben-Sasson, E., Goldberg, L., Kopparty, S., Saraf, S.: DEEP-FRI: sampling outside the box improves soundness. In: 11th Innovations in Theoretical Computer Science Conference, ITCS 2020, January 12-14, 2020, Seattle, Washington, USA. pp. 5:1–5:32 (2020)
8. Ben-Sasson, E., Goldreich, O., Harsha, P., Sudan, M., Vadhan, S.P.: Robust PCPs of proximity, shorter PCPs and applications to coding. In: Babai, L. (ed.) Proceedings of the 36th Annual ACM Symposium on Theory of Computing, Chicago, IL, USA, June 13-16, 2004. pp. 1–10. ACM (2004)
9. Ben-Sasson, E., Kopparty, S., Saraf, S.: Worst-case to average case reductions for the distance to a code. In: 33rd Computational Complexity Conference, CCC 2018, June 22-24, 2018, San Diego, CA, USA. pp. 24:1–24:23 (2018)
10. Bootle, J., Chiesa, A., Groth, J.: Linear-time arguments with sublinear verification from tensor codes. In: Pass, R., Pietrzak, K. (eds.) Theory of Cryptography - 18th International Conference, TCC 2020, Durham, NC, USA, November 16-19, 2020, Proceedings, Part II. Lecture Notes in Computer Science, vol. 12551, pp. 19–46. Springer (2020)
11. Chiesa, A., Ojha, D., Spooner, N.: Fractal: Post-quantum and transparent recursive proofs from holography. In: Canteaut, A., Ishai, Y. (eds.) Advances in Cryptology - EUROCRYPT 2020 - 39th Annual International Conference on the Theory and Applications of Cryptographic Techniques, Zagreb, Croatia, May 10-14, 2020, Proceedings, Part I. Lecture Notes in Computer Science, vol. 12105, pp. 769–793. Springer (2020)
12. Dinur, I., Reingold, O.: Assignment testers: Towards a combinatorial proof of the pcp-theorem. In: 45th Symposium on Foundations of Computer Science (FOCS 2004), 17-19 October 2004, Rome, Italy, Proceedings. pp. 155–164. IEEE Computer Society (2004)
13. Kalai, Y.T., Raz, R.: Interactive PCP. In: Aceto, L., Damgård, I., Goldberg, L.A., Halldórsson, M.M., Ingólfsdóttir, A., Walukiewicz, I. (eds.) Automata, Languages and Programming, 35th International Colloquium, ICALP 2008, Reykjavik, Iceland, July 7-11, 2008, Proceedings, Part II - Track B: Logic, Semantics, and Theory of Programming & Track C: Security and Cryptography Foundations. Lecture Notes in Computer Science, vol. 5126, pp. 536–547. Springer (2008)
14. Kattis, A., Panarin, K., Vlasov, A.: Redshift: Transparent snarks from list polynomial commitment iops. Cryptology ePrint Archive, Report 2019/1400 (2019), `https://ia.cr/2019/1400`
15. Reingold, O., Rothblum, G.N., Rothblum, R.D.: Constant-round interactive proofs for delegating computation. In: Wichs, D., Mansour, Y. (eds.) Proceedings of the 48th Annual ACM SIGACT Symposium on Theory of Computing, STOC 2016, Cambridge, MA, USA, June 18-21, 2016. pp. 49–62. ACM (2016)
16. Schwartz, J.T.: Fast probabilistic algorithms for verification of polynomial identities. J. ACM **27**(4), 701–717 (1980)
17. Zippel, R.: Probabilistic algorithms for sparse polynomials. In: Ng, E.W. (ed.) Symbolic and Algebraic Computation, EUROSAM '79, An International Symposiumon Symbolic and Algebraic Computation, Marseille, France, June 1979, Proceedings. Lecture Notes in Computer Science, vol. 72, pp. 216–226. Springer (1979)