Power Decoding of Reed–Solomon Codes Revisited

Nielsen, Johan Sebastian Rosenkilde

Published in:
Proceedings of the 4th International Castle Meeting on Coding Theory and Applications (4ICMCTA 2014)

Link to article, DOI:
10.1007/978-3-319-17296-5_32

Publication date:
2014

Document Version
Peer reviewed version

Power Decoding of Reed–Solomon Codes Revisited

Johan S. R. Nielsen

Abstract Power decoding, or "decoding by virtual interleaving", of Reed–Solomon codes is a method for unique decoding beyond half the minimum distance. We give a new variant of the Power decoding scheme, building upon the key equation of Gao. We show various interesting properties such as behavioural equivalence to the classical scheme using syndromes, as well as a new bound on the failure probability when the powering degree is 3.

Key words: Reed-Solomon code, Algebraic decoding, Power decoding

1 Introduction

Power decoding was originally developed by Schmidt, Sidorenko and Bossert for low-rate Reed–Solomon codes (RS) [6], and is usually capable of decoding almost as many errors as the Sudan decoder [9] though it is a unique decoder. If an answer is found, this is always the closest codeword, but in some cases the method will fail; in particular, this happens if two codewords are equally close to the received. With random errors this seems to happen exceedingly rarely, though a bound for the probability has only been shown for the simplest case of powering degree 2 [6, 10].

The algorithm rests on the surprising fact that a received word coming from a low-rate RS code can be "powered" to give received words of higher-rate RS codes having the same error positions. For each of these received words, one constructs a classical key equation by calculating the corresponding syndromes and solves them simultaneously for the same error locator polynomial.

Gao gave a variant of unique decoding up to half the minimum distance [1]: in essence, his algorithm uses a different key equation and with this finds the information polynomial directly. We here show how to easily derive a variant of Power

Johan S. R. Nielsen
Ulm University, Institute of Communications Engineering, Ulm, Germany, e-mail: jsrn@jsrn.dk

decoding for Generalised RS (GRS) codes, Power Gao, where we obtain multiple of Gao's type of key equation, and we solve these simultaneously.

We then show that Power Gao is *equivalent* to Power syndromes in the sense that they will either both fail or both succeed for a given received word. Power Gao has some "practical" advantages, though: it extends Power decoding to the case of using 0 as an evaluation point (which Power syndromes does not support); and the information is obtained directly when solving the key equations, so finding roots of the error locator and Forney's formula is not necessary.

The main theoretical advantage is that Power Gao seems easier to analyse: in particular, we show two new properties of Power decoding: 1) that whether Power decoding fails or not depends only on the error and not on the sent codeword; and 2) a new bound on the failure probability when the powering degree is 3.

We briefly sketched Power Gao already in [3], but its behaviour was not well analysed and its relation to Power syndromes not examined. In Section 2 we derive the powered Gao key equations, and in Section 3 we describe the complete algorithm and discuss computational complexity issues. In Section 4 we show the behavioural equivalence to Power syndromes as well as the new properties on Power decoding. Section 5 describes an explicit family of errors for which Power decoding will fail.

2 The Key Equations

Consider some finite field $\mathbb{F}$. The $[n,k,d]$ Generalised Reed-Solomon (GRS) code is the set

$$\mathscr{C} = \left\{ \left(\beta_1 f(\alpha_1), \ldots, \beta_n f(\alpha_n)\right) \mid f \in \mathbb{F}[x] \wedge \deg f < k \right\}$$

where $\alpha_1, \ldots, \alpha_n \in \mathbb{F}$ are distinct, and the $\beta_1, \ldots, \beta_n \in \mathbb{F}$ are non-zero (not necessarily distinct). The α_i are called *evaluation points* and the β_i *column multipliers*. $\mathscr{C}$ has minimum distance $d = n-k+1$ and the code is therefore MDS.

Consider now that some $\boldsymbol{c} = (c_1, \ldots, c_n)$ was sent, resulting from evaluating some $f \in \mathbb{F}[x]$, and that $\boldsymbol{r} = (\beta_1 r_1, \ldots, \beta_n r_n) = \boldsymbol{c} + (\beta_1 e_1, \ldots, \beta_n e_n)$ was the received word with (normalised) error $\boldsymbol{e} = (e_1, \ldots, e_n)$. Let $\mathscr{E} = \{i \mid e_i \neq 0\}$ and $\varepsilon = |\mathscr{E}|$. In failure probability considerations, we consider the $|\mathbb{F}|$-ary symmetric channel.

Introduce $G \triangleq \prod_{i=1}^{n}(x-\alpha_i)$, and for any integer $t \geq 1$, let $R^{(t)}$ be the Lagrangian polynomial through the "powered" $\boldsymbol{r}$, i.e. the minimal degree polynomial satisfying $R^{(t)}(\alpha_i) = r_i^t$ for $i = 1, \ldots, n$. Naturally, we have $\deg R^{(t)} \leq n-1$ and $R^{(t)}$ can be directly calculated by the receiver. As usual for key equation decoders, the algorithm will revolve around the notion of error locator: $\Lambda = \prod_{j \in \mathscr{E}}(x-\alpha_j)$. Choose now some $\ell \in \mathbb{N}$ subject to $\ell(k-1) < n$. Then we easily derive the powered Gao key equations:

Proposition 1. $\Lambda R^{(t)} \equiv \Lambda f^t \mod G$

Proof. Polynomials are equivalent modulo G if and only if they have the same evaluation at $\alpha_1, \ldots, \alpha_n$. For α_i where $e_i \neq 0$, both sides of the above evaluate to zero, while for the remaining α_i they give $\Lambda(\alpha_i) {r_i}^t = \Lambda(\alpha_i) f(\alpha_i)^t$. □

3 The Decoding Algorithm

The key equations of Proposition 1 are non-linear in Λ and f, so the approach for solving them is to relax the equations into a linear system, similarly to classical key equation decoding. We will ignore the structure of the right hand-sides and therefore seek polynomials λ and $\psi^{(1)},\dots,\psi^{(\ell)}$ such that $\lambda R^{(t)} \equiv \psi^{(t)} \mod G$ as well as $\deg\lambda + t(k-1) \geq \deg\psi^{(t)}$ for $t = 1,\dots,\ell$. We will call such $(\lambda,\psi^{(1)},\dots,\psi^{(\ell)})$ *a solution* to the key equations.

Clearly $(\Lambda,\Lambda f,\dots,\Lambda f^{\ell})$ is a solution. There are, however, infinitely many more, so the strategy is to find a solution such that $\deg\lambda$ is minimal; we will call this the *minimal solution*. Thus decoding can only succeed when Λ has minimal degree of all solutions. The probability of this occurring will be discussed in Section 4.

Conceptually, Power Gao decoding is then straightforward: pre-calculate G and from the received word, calculate $R^{(1)},\dots,R^{(\ell)}$. Find then a minimal solution $(\lambda,\psi_1,\dots,\psi_\ell)$ with λ monic. If this has the valid structure of $(\Lambda,\Lambda f,\dots,\Lambda f^{\ell})$, then return f. Otherwise, declare decoding failure.

For Power syndromes, the key equations are similar to ours except that the modulo polynomials are just powers of x. In this case, finding a minimal solution is known as multi-sequence shift-register synthesis, and the fastest known algorithm is an extension of the Berlekamp–Massey algorithm [6] or the Divide-&-Conquer variant of this [7]. These can not handle the modulus G that we need, however.

A generalised form of multi-sequence shift-register synthesis was considered in [3], and several algorithms for finding a minimal solution were presented. The key equations for our case fit into this framework. We refer the reader to [3] for the details on these algorithms, but the asymptotic complexities when applied to Power Gao decoding are given in Table 1 on the following page. The same complexities would apply to Power syndromes and also match the algorithms [6, 7] mentioned before. The other steps of the decoding are easily seen to be cheaper than this; e.g. the calculation of $R^{(1)},\dots,R^{(\ell)}$ by Lagrangian interpolation can be done trivially in $O(\ell n^2)$ or using fast Fourier techniques in $O(\ell n\log^2 n)$ [2, p. 231]. Thus Power Gao decoding is asymptotically as fast as Power syndromes.

4 Properties of the Algorithm

Power Gao will fail if $(\Lambda,\Lambda f,\dots,\Lambda f^{\ell})$ is not the found minimal solution, so the question is when one can expect this to occur. Since the algorithm returns at most one codeword, it *must* fail for some received words whenever $\varepsilon \geq d/2$. Whenever an answer is found, however, this must correspond to a closest codeword: any closer codeword would have its own corresponding error locator and information polynomial, and these would yield a smaller solution to the key equations.

We first show that Power syndromes is behaviourally equivalent to Power Gao. We will need to assume that the evaluation points $\alpha_i \neq 0$ for all i, which is a condi-

Table 1 Complexities of solving the key equations for the three approaches discussed in [3].

Algorithm	O-complexity
Mulders–Storjohann	$\ell^2 n^2$
Alekhnovich	$\ell^3 n \log^2 n \log\log n$
Demand–Driven*	$\ell n^2 [\log n \log\log n]$

*: If $\mathscr{C}$ is cyclic, then $G = x^n - 1$ since the α_i form a multiplicative group, and in this case the log-factors in square brackets can be removed.

tion for Power syndromes decoding. This implies $x \nmid G$. We will use a "coefficient reversal" operator defined for any $p \in \mathbb{F}[x]$ as $\overline{p} = x^{\deg p} p(x^{-1})$.

In Power syndromes decoding, one considers $\boldsymbol{r}^{(t)} = (\beta_1 r_1^t, \ldots, \beta_n r_n^t)$ for $t = 1, \ldots, \ell$ as received words of GRS codes with parameters $[n, t(k-1)+1, n-t(k-1)]$, resulting from evaluating f^t; these "virtual" codes have the same evaluation points and column multipliers as $\mathscr{C}$. The $\boldsymbol{r}^{(t)}$ will therefore have the same error positions as $\boldsymbol{r}$, so the same error locator applies. For each t, we can calculate the syndrome $S^{(t)}$ corresponding to $\boldsymbol{r}^{(t)}$, which can be written as

$$S^{(t)} = \Big(\sum_{i=1}^{n} \frac{r_i^t \zeta_i}{1 - x\alpha_i} \bmod x^{n-t(k-1)+1} \Big)$$

where $\zeta_i = \prod_{j \neq i} (\alpha_i - \alpha_j)^{-1}$; see e.g. [5, p. 185]. By insertion one sees that

$$\overline{\Lambda} S^{(t)} \equiv \Omega^{(t)} \mod x^{n-t(k-1)+1}, \quad t = 1, \ldots, \ell$$

where $\Omega^{(t)}$ is a certain polynomial satisfying $\deg \Omega^{(t)} < \deg \Lambda$. Note that we are using Λ reversed; indeed, one often defines error-locator as $\prod_{i \in \mathscr{E}} (1 - x\alpha_i) = \overline{\Lambda}$ when considering the syndrome key equation. The decoding algorithm follows simply from finding a minimal degree polynomial $\overline{\lambda}$ such that $\omega^{(t)} = (\overline{\lambda} S^{(t)} \bmod x^{n-t(k-1)+1})$ satisfies $\deg \lambda > \deg \omega^{(t)}$ for all t. The decoding method fails if $\overline{\lambda} \neq \gamma \overline{\Lambda}, \forall \gamma \in \mathbb{F}$. We now have:

Proposition 2. *Decoding using Power Gao fails if and only if decoding using Power syndromes fails.*

Proof. Note first that $R^{(t)} = \sum_{i=1}^{n} r_i^t \zeta_i \prod_{j \neq i} (x - \alpha_j)$. By insertion we get $S^{(t)} \equiv \overline{R}^{(t)} \overline{G}^{-1} \mod x^{n-t(k-1)+1}$ (since $x \nmid G$). Power Gao fails if there is some $\lambda \in \mathbb{F}[x]$ which is not a constant times Λ and such that $\deg \lambda \leq \deg \Lambda$ and $\psi^{(t)} = (\lambda R^{(t)} \bmod G)$ has $\deg \psi^{(t)} < \deg \lambda + t(k-1) + 1$ for each $t = 1, \ldots, \ell$. This means there must be some $\omega^{(t)}$ with $\deg \omega^{(t)} \leq \deg \lambda - 1$ such that

$$\begin{aligned} \lambda R^{(t)} - \omega^{(t)} G &= \psi && \iff \\ \overline{\lambda}\,\overline{R}^{(t)} - \overline{\omega}^{(t)} \overline{G} &= \overline{\psi}^{(t)} x^{\deg G + \deg \lambda - 1 - (\deg \lambda + t(k-1))} && \implies \\ \overline{\lambda}\,\overline{R}^{(t)} &\equiv \overline{\omega}^{(t)} \overline{G} \mod x^{n-t(k-1)-1} \end{aligned}$$

Dividing by $\overline{G}$, we see that $\overline{\lambda}$ and the $\overline{\omega}^{(t)}$ satisfy the congruences necessary to form a solution to the Power syndromes key equation, and they also satisfy the degree bounds. Showing the proposition in the other direction runs analogously. □

Corollary 1 (Combining [6] and Proposition 2). *Power Gao decoding succeeds if $\varepsilon < d/2$. Let*

$$\tau(\ell) = \tfrac{\ell}{\ell+1} n - \tfrac{1}{2}\ell(k-1) - \tfrac{\ell}{\ell+1}$$

Then decoding will fail with high probability if $\varepsilon > \tau(\hat{\ell})$, where $1 \le \hat{\ell} \le \ell$ is chosen to maximise $\tau(\ell)$. [1]

Between the above two bounds, Power decoding will sometimes succeed and sometimes fail. Simulations indicate that failure occurs with quite small probability. The only proven bound so far is for $\ell = 2$ where for exactly ε errors occurring, we have $P_f(\varepsilon) < (q/q-1)^{\varepsilon} q^{3(\varepsilon-\tau(2))}/(q-1)$, [6, 10].

We will give a new bound for $P_f(\varepsilon)$ when $\ell = 3$, but we will first show a property which allows a major simplification in all subsequent analyses.

Proposition 3. *Power Gao decoding fails for some received word $\boldsymbol{r}$ if and only if it fails for $\boldsymbol{r} + \hat{\boldsymbol{c}}$ where $\hat{\boldsymbol{c}}$ is any codeword.*

Proof. We will show that Power Gao decoding fails for $\boldsymbol{r} = \boldsymbol{c} + \boldsymbol{e}$ if and only if it fails for $\boldsymbol{e}$ as received word; since $\boldsymbol{c}$ was arbitrary, that implies the proposition.

Let $R_e^{(t)}$ be the power Lagrangians for $\boldsymbol{e}$ as received word, i.e. $R_e^{(t)}(\alpha_i) = e_i^t$ for each i and t, and let $R_e = R_e^{(1)}$. Consider a solution to the corresponding key equations $(\lambda, \psi_1, \ldots, \psi_\ell)$; i.e. $\lambda R_e^{(t)} \equiv \psi_t \mod G$ and $\deg\lambda + t(k-1) + 1 > \deg\psi_t$. Let as usual $R^{(t)}$ be the power Lagrangians for $\boldsymbol{r}$ as received word and $R = R^{(1)}$. Note now that $R^{(t)} \equiv R^t \mod G$ since both sides of the congruence evaluate to the same at all α_i; similarly $R_e^{(t)} \equiv R_e^t \mod G$. Since $r_i = f(\alpha_i) + e_i$ linearity implies that $R = f + R_e$. Define $\psi_0 = \lambda$ and note that then also for $t = 0$ we have $\deg\lambda + t(k-1) + 1 > \deg\psi_t$. We then have the chain of congruences modulo G:

$$\lambda R^{(t)} \equiv \lambda R^t \equiv \lambda (f + R_e)^t \equiv \lambda \textstyle\sum_{s=0}^{t} \binom{t}{s} f^s R_e^{t-s} \equiv \sum_{s=0}^{t} \binom{t}{s} f^s \psi_{t-s} \mod G$$

Each term in the last sum has degree $s\deg f + \deg\psi_{t-s} < s(k-1) + \deg\lambda + (t-s)(k-1) + 1 = \deg\lambda + t(k-1) + 1$, which means that

$$\left(\lambda,\ \textstyle\sum_{s=0}^{1} \binom{1}{s} f^s \psi_{1-s},\ \ldots\ ,\ \sum_{s=0}^{\ell} \binom{\ell}{s} f^s \psi_{\ell-s}\right)$$

is a solution to the key equations with $\boldsymbol{r}$ as a received word. The same argument holds in the other direction, so any solution to one of the key equations induces a solution to the other with the same first component; obviously then, their minimal solutions must be in bijection, which directly implies that they either both fail or neither of them fail. □

For the new bound on the failure probability, we first need a technical lemma:

Lemma 1. *Let $U \in \mathbb{F}[x]$ of degree N, and let $K_1 < K_2 < K_3 < N$ be integers. Let $S = \{(f_1, f_2, f_3) \mid f_1 f_3 \equiv f_2^2 \mod U,\ f_2$ monic $,\ \forall t.\deg f_t < K_t\}$. Then*

[1] Decoding may succeed in certain degenerate cases, see [4, Proposition 2.39]. Failure is certain when using the method of [6] since what it considers "solutions" are subtly different than here.

$$\begin{aligned} |S| &\leq 3^{K_2-1} q^{K_2} && \text{if } K_1+K_3-2 < N \\ |S| &\leq 2^{K_1+K_3-2} q^{K_1+K_2+K_3-N-2} && \text{if } K_1+K_3-2 \geq N \end{aligned}$$

Proof. If $K_1+K_3-2<N$, then $f_1 f_3 \equiv f_2^2 \mod U$ implies $f_1 f_3 = f_2^2$. We can choose a monic f_2 in $(q^{K_2}-1)/(q-1)$ ways. For each choice, then f_2 has at most K_2-1 prime factors, so the factors of f_2^2 can be distributed among f_1 and f_3 in at most 3^{K_2-1} ways. Lastly, the leading coefficient of f_1 can be chosen in $q-1$ ways.

If $K_1+K_3-2 \geq N$, then for each choice of f_2, the product $f_1 f_3$ can be among $\{f_2^2 + gU \mid \deg g \leq K_1+K_3-2-N\}$. This yields at most $q^{K_1+K_2+K_3-N-2}/(q-1)$ candidates for $f_1 f_2$; each of these has at most K_1+K_3-2 unique prime factors, which can then be distributed among f_1 and f_2 in at most $2^{K_1+K_3-2}$ ways. Again, the leading coefficient of f_1 leads to a factor $q-1$ more. □

Proposition 4. *For $\ell = 3$, the probability that Power decoding (Gao or Syndrome) fails when $\varepsilon > d/2$ is at most*

$$\begin{aligned} (q/(q-1))^{\varepsilon} (3/q)^{2\varepsilon-(n-2k+1)} q^{3(\varepsilon-\tau(2))+k-1} && \text{if } \varepsilon < \tau(2) - \tfrac{1}{3}k + 1 \\ (q/(q-1))^{\varepsilon} 2^{2(2\varepsilon-d)+2(k-1)} q^{4(\varepsilon-\tau(3))-2} && \text{if } \varepsilon \geq \tau(2) - \tfrac{1}{3}k + 1 \end{aligned}$$

Proof. By Proposition 3, we can assume that $\boldsymbol{c} = 0$, i.e. that $\boldsymbol{r} = \boldsymbol{e}$. That means $R^{(t)}(\alpha_i) = 0$ for $i \notin \mathscr{E}$, so we can write $R^{(t)} = E^{(t)} \Upsilon$ for some $E^{(t)}$ with $\deg E^{(t)} < \varepsilon$, where $\Upsilon = G/\Lambda$ is the "truth-locator". Power Gao decoding fails if and only if there exists $(\lambda, \psi_1, \psi_2, \psi_3)$ such that $\lambda \neq \Lambda$, $\deg \lambda \leq \deg \Lambda$, $\deg \lambda + t(k-1) + 1 > \deg \psi_t$ for $t = 1,2,3$ as well as

$$\lambda R^{(t)} \equiv \psi_t \mod G \qquad \Longleftrightarrow \qquad \lambda E^{(t)} \equiv \hat{\psi}_t \mod \Lambda$$

where $\hat{\psi}_t = \psi_t / \Upsilon$. Note that ψ_t must be divisible by Υ since both the modulus and the left-hand side of the first congruence is.

Denote by E the unique polynomial with degree less than ε having $E(\alpha_i) = e_i$ for $i \in \mathscr{E}$. For any $i \in \mathscr{E}$ then $(\lambda E^{(t)})(\alpha_i) = \lambda(\alpha_i) \Upsilon(\alpha_i)^{-1} e_i^t$, which means $\lambda E^{(t)} \equiv \hat{\lambda} E^t \mod \Lambda$ for some polynomial $\hat{\lambda}$.

After having chosen error positions, drawing error values uniformly at random is the same as drawing uniformly at random from possible E. So given the error positions, the probability that Power decoding will fail is $T_\Lambda/(q-1)^{\varepsilon}$, where T_Λ is the number of choices of E such that there exist $\hat{\lambda}, \hat{\psi}_1, \hat{\psi}_2, \hat{\psi}_3$ having

$$\hat{\lambda} E^t \equiv \hat{\psi}_t \mod \Lambda, \quad t = 1,2,3$$

as well as $\deg \hat{\psi}_t < \deg \Lambda + t(k-1) + 1 - (n - \deg \Lambda) = 2\varepsilon - (n - t(k-1) - 1)$.

Note that these congruences imply $\hat{\psi}_1 \hat{\psi}_3 \equiv \hat{\psi}_2^2 \mod \Lambda$. Denote by $\hat{T}_\Lambda$ the number of triples $(\hat{\psi}_1, \hat{\psi}_2, \hat{\psi}_3) \in \mathbb{F}[x]^3$ satisfying just this congruence as well as the above degree bounds. Then $\hat{T}_\Lambda \geq T_\Lambda$: for if $\gcd(\hat{\lambda}, \Lambda) = 1$ then two different values of E could not yield the same triple since $E \equiv \hat{\psi}_2/\hat{\psi}_1 \mod \Lambda$ uniquely determines E. Alternatively, if $\gcd(\hat{\lambda}, \Lambda) = g \neq 1$ then the congruences imply $g \mid \hat{\psi}_t$ for all t, so that $E \equiv (\hat{\psi}_2/g)/(\hat{\psi}_1/g) \mod \Lambda/g$. This leaves a potential $q^{\deg g}$ possible other choices

of E yielding the same triple; but all these possibilities are counted in the triples since $(t\psi_1/g, t\psi_2/g, t\psi_3/g)$ will be counted for any $t \in \mathbb{F}[x]$ with $\deg t < \deg g$.

In fact, we have $\hat{T}_\Lambda \geq (q-1)T_\Lambda$, since whenever $(\hat{\psi}_1, \hat{\psi}_2, \hat{\psi}_3)$ is counted, so is $(\beta\hat{\psi}_1, \beta^2\hat{\psi}_2, \hat{\psi}_3)$, and this doesn't change the fraction $\hat{\psi}_1/\hat{\psi}_2$. Thus, we overestimate instead $\hat{T}_\Lambda/(q-1)$ by counting the number of triples where $\hat{\psi}_2$ is monic. Lemma 1 gives an upper bound for exactly this number, setting $N = \varepsilon$ and $K_t = 2\varepsilon - (n - t(k-1) - 1)$. Divided by $(q-1)^\varepsilon$, this is then an upper bound on the failure probability given the error positions. But since this probability is independent of the choice of Λ, it is also the failure probability over all errors vectors of weight ε. □

By experimentation, one can demonstrate that the bound is not tight: for instance, for a $[250, 30, 221]$ GRS code, the bound is greater than 1 for $\varepsilon > 143$, while simulation indicate almost flawless decoding up to 147 errors. However, in a relative and asymptotic sense the above bound is strong enough to show that up to $\tau(3)$ errors can be corrected with arbitrary low failure probability:

Corollary 2. *Having $\ell = 3$, then for any $\delta > 0$, with $n \to \infty$ while keeping q/n, k/n and ε/n constant, the probability that Power decoding fails goes to 0 when $\varepsilon/n < \tau(3)/n - \delta$.*

Proof (Proof sketch). We consider only the high-error failure probability of Proposition 4. For $n \to \infty$, the failure probability bound will approach

$$2^{2(2\varepsilon-d)+2(k-1)} q^{4(\varepsilon-\tau(3))} \leq (q^n)^{4(\varepsilon/n-\tau(3)/n)+(2(2\varepsilon/n-d/n)+2k/n)/\log q}$$

The contribution $(2(2\varepsilon/n - d/n) + 2k/n)/\log q$ goes to 0 as $n \to \infty$, leaving $(q^n)^{-a}$ for $a = 4(\varepsilon/n - \tau(3)/n) < -4\delta$. □

5 A family of bad errors

Power decoding will usually fail when the powered key equations are linearly dependent; in particular, it will fail if one of the key equations is trivially satisfied.

An anonymous reviewer of this paper suggested the following construction of errors where, for a given sent codeword, the second key equation will be trivial: let $\mathbb{F}$ be a non-binary field, and let $\hat{\boldsymbol{c}} \in \mathscr{C}$ be some non-zero codeword, obtained as the evaluation of $\hat{f}$ with $\deg \hat{f} < k$. Choose $d/2 \leq \varepsilon \leq \tau(2))$ positions for which $\hat{\boldsymbol{c}}$ is non-zero. Let then $\boldsymbol{e} = (e_1, \ldots, e_n)$ be given by $e_i = -2\hat{c}_i$ for i being one of the chosen position, and $e_i = 0$ otherwise. If $\hat{\boldsymbol{r}} = \hat{\boldsymbol{c}} + \boldsymbol{e}$ is received, then the second Lagrangian $\hat{R}^{(2)}$ equals $\hat{f}^2$, i.e. $\deg \hat{R}^{(2)} < 2k - 1$ (in other words, the squared received word $\hat{\boldsymbol{r}}^{(2)} = \hat{\boldsymbol{c}}^{(2)}$ and so is a codeword in the "squared" GRS code). That means that for *any* $\lambda \in \mathbb{F}[x]$, then $\deg(\lambda\hat{R}^{(2)} \bmod G) \leq \deg\lambda + 2k - 1$, and so the second key equation is useless.

Clearly then, if $\hat{\boldsymbol{r}}$ is received, then almost surely[2] Power decoding will fail when $\ell = 2$, and it is easy to show that it will also fail when $\ell > 2$.

[2] As in Corollary 1, failure is not certain but extremely unlikely for just a few errors beyond $d/2$.

From Proposition 3 it follows that decoding will also fail when receiving $\boldsymbol{c}+\boldsymbol{e}$ for *any* sent codeword $\boldsymbol{c} \in \mathscr{C}$; in particular when sending $\mathbf{0}$ and receiving $\boldsymbol{e}$. This might at first seem counter-intuitive since the second Lagrangian $E^{(2)}$ when $\boldsymbol{e}$ is the received word does not have low degree (i.e. $\boldsymbol{e}^{(2)}$ is *not* in the squared GRS code). However, in this case the key equation involving $E^{(2)}$ will be linearly dependent on that involving $E = E^{(1)}$, and so will not add further requirements. This can be seen directly as follows: since $\boldsymbol{e} = \hat{\boldsymbol{r}} - \hat{\boldsymbol{c}}$ then $\boldsymbol{e}^{(2)} = \hat{\boldsymbol{r}}^{(2)} + \hat{\boldsymbol{c}}^{(2)} - 2\hat{\boldsymbol{c}} \star \hat{\boldsymbol{r}} = 2\hat{\boldsymbol{c}}^{(2)} - 2\hat{\boldsymbol{c}} \star \hat{\boldsymbol{r}}$, where $\star$ is the component-wise product. Thus $E^{(2)} \equiv 2\hat{f}^2 - 2\hat{f}R \mod G$. So if $\lambda \in \mathbb{F}[x]$ satisfies the first key equation, i.e. $\deg(\lambda E \bmod G) \leq \deg\lambda + k - 1$, then we get

$$\deg(\lambda E^{(2)} \bmod G) = \deg(\lambda \hat{f}^2 + \lambda E \hat{f} \bmod G) \leq \deg\lambda + 2(k-1)$$

So λ satisfies the second key equation.

The "bad error" construction can easily be generalised for higher ℓ whenever $\mathbb{F}$ has ℓ'th roots of unity different from 1: then e_i can be chosen as $(\xi_i - 1)\hat{c}_i$ where $\xi_i \neq 1$ is any of those roots of unity. Then for $\hat{\boldsymbol{r}} = \hat{\boldsymbol{c}} + \boldsymbol{e}$ we get $\hat{\boldsymbol{r}}^{(\ell)} = \hat{\boldsymbol{c}}^{(\ell)}$ and so $\deg R^{(\ell)} \leq \ell(k-1)$.

A full Power Gao decoder has been implemented in Sage v5.13 [8] and is available for download at http://jsrn.dk/code-for-articles. Also implemented is randomly constructing "bad errors" $\boldsymbol{e}$ as above (for any ℓ), and a demonstration that Power decoding fails for $\hat{\boldsymbol{r}}$, $\boldsymbol{e}$ and $\boldsymbol{c}+\boldsymbol{e}$ for any random codeword $\boldsymbol{c}$.

References

1. Gao, S.: A new algorithm for decoding Reed-Solomon codes. In: Communications, Information and Network Security, no. 712 in S. Eng. and Comp. Sc., pp. 55–68. Springer (2003)
2. von zur Gathen, J., Gerhard, J.: Modern Computer Algebra. Cambridge Univ. Press, 3rd edn. (2012)
3. Nielsen, J.S.R.: Generalised multi-sequence shift-register synthesis using module minimisation. In: Proc. of IEEE ISIT (2013)
4. Nielsen, J.S.R.: List decoding of algebraic codes. Ph.D. thesis, Technical University of Denmark (2013). Available at jsrn.dk
5. Roth, R.: Introduction to Coding Theory. Cambridge Univ. Press (2006)
6. Schmidt, G., Sidorenko, V., Bossert, M.: Syndrome decoding of Reed-Solomon codes beyond half the minimum distance based on shift-register synthesis. IEEE Trans. Inf. Theory **56**(10), 5245–5252 (2010)
7. Sidorenko, V., Bossert, M.: Fast skew-feedback shift-register synthesis. Designs, Codes and Cryptography p. 1–13 (2011)
8. Stein, W.A., et al.: Sage Mathematics Software. Http://www.sagemath.org
9. Sudan, M.: Decoding of Reed–Solomon codes beyond the error-correction bound. J. Complexity **13**(1), 180–193 (1997)
10. Zeh, A., Wachter, A., Bossert, M.: Unambiguous decoding of generalized Reed–Solomon codes beyond half the minimum distance. In: Proc. of IZS (2012)