Códigos Corretores de Erros

Textuniversitários 2

José Felipe Voloch

Códigos Corretores *de Erros*

Editora Livraria da Física
São Paulo – 2020

1a. Edição

Editor: José Roberto Marinho
Projeto gráfico e diagramação: Thiago Augusto Silva Dourado
Capa: Fabrício Ribeiro

Texto em conformidade com as novas regras ortográficas do Acordo da Língua Portuguesa.

Dados Internacionais de Catalogação na Publicação (CIP)
(Câmara Brasileira do Livro, SP, Brasil)

Voloch, José Felipe

Códigos corretores de erros / José Felipe Voloch. – São Paulo : Editora Livraria da Física, 2020. – (Série textuniversitários ; 2)

Bibliografia.
ISBN 978-85-7861-640-3

1. Ciência da computação - Matemática 2. Códigos 3. Teoria dos erros I. Título. II. Série.

19-31526 CDD-511.43

Índices para catálogo sistemático:

1. Teoria de códigos corretores de erros : Matemática 511.43

Maria Paula C. Riyuzo – Bibliotecária – CRB-8/7639

ISBN 978-85-7861-640-3

Impresso no Brasil
Printed in Brazil

Editora Livraria da Física
Tel./Fax: +55 11 3459-4327 / 3936-3413
www.livrariadafisica.com.br

Prefácio

Esse pequeno livro foi escrito para um curso que ministrei no 16° Colóquio Brasileiro de Matemática em 1987. O texto não foi alterado para esta edição. Felizmente, a base da teoria continua a mesma e o livro pode ser usado como uma entrada à teoria dos códigos corretores de erros que se mantem relevante hoje em dia.

Alguns avanços recentes merecem ser destacados. Não colocarei referências bibliográficas pois é mais fácil e eficiente fazer uma busca pela internet.

O problema tradicional de obter-se códigos com boa performance para comunicação ponto a ponto foi resolvido empiricamente pelos códigos LDPC (*low-density parity check codes* ou códigos com matriz de paridade com baixa densidade, em português). Esses códigos haviam sido propostos na década de 1960 por R. Gallager mas só recentemente, com o uso de computadores, passou a ser possível o teste empírico para achar códigos LDPC de performance atingindo o melhor valor possível. Porém, ainda falta justificativa teórica que a construção funciona.

Ainda na década de 1970, R. McEliece sugeriu o uso de códigos de Goppa para a construção de sistemas criptográficos de chave pública. Na época, por serem menos eficientes que o RSA e outros sistemas, a proposta de McEliece não foi levada avante. Recentemente,

com a ameaça de que o RSA e outros possam ser quebrado num futuro próximo por computadores quânticos, as ideias de McEliece foram revistas e são base de candidatos fortes a ganhar a competição organizada pela NIST, do governo dos EUA, para sistemas criptográficos pós-quânticos a serem usados pelo governo americano.

Recentemente, com o advento de armazenamento de dados na "Nuvem", surgiram novas aplicações da teoria de códigos onde se buscam códigos com a propriedade de ser localmente recuperáveis (*locally recoverable codes*, LRC), isto é, cada coordenada de um elemento do código pode ser recuperada a partir de um número pequeno de outras coordenadas. As construções algébricas apresentadas nesse livro ganharam nova vida sendo bastante úteis em construir tais códigos.

Finalmente, a conjectura mencionada no final do Capitulo 4 (a conjectura MDS, p. 44) foi demonstrada por S. Ball em 2012 no caso em que o número q é primo mas continua aberta em geral.

Agradeço a Thiago Augusto Silva Dourado e sua equipe da LF editorial pela reedição do livro.

José Felipe Voloch
Christchurch, Nova Zelândia,
02/11/2019

Sumário

Introdução

Desde a sua introdução por Claude Shannon (um matemático que trabalhava no Bell Lab.), a teoria de códigos corretores de erros tem tido inúmeras aplicações. Ela intervém todas as vezes que queremos transmitir ou estocar mensagens ou dados que estão sujeitos à interferência que causem erros na mensagem a ser lida posteriormente. Exemplos usuais são as transmissões por satélite e a estocagem de dados em fitas magnéticas de computadores.

O problema que se põe é: dada uma mensagem recebida que contém um número de erros, como corrigir estes erros e recuperar a mensagem enviada? Se a mensagem enviada contém redundâncias então, se a quantidade de erros é pequena, podemos esperar recuperar a mensagem. Essa é a filosofia dos códigos corretores de erros. No aspecto prático da coisa a percentagem de erros na mensagem é conhecida (por experiência, por exemplo), pois o canal de comunicação é dado. Por outro lado, a quantidade de redundância que podemos colocar é limitada pelos gastos que queremos fazer.

Vamos dar um exemplo específico. Suponhamos que queremos enviar mensagens (a, b, c), com $a, b, c \in \{0, 1\}$, e digamos que o nosso canal de comunicações causa um erro em cada seis dígitos consecutivos. Então se enviarmos a mensagem pura e simples o

receptor vai receber uma mensagem errada a cada duas enviadas. Isso é ruim. Outra tentativa é repetir cada mensagem, introduzindo redundância. Isso também é ruim, pois se o receptor recebe, por exemplo, (a, b, c), (a', b, c), $a \neq a'$, como ele vai saber se o primeiro dígito da mensagem é a ou a'? (Que os outros dois são b, c, ele já sabe, pois vieram repetidos). Se repetirmos a mensagem três vezes então certamente o receptor saberá qual é a mensagem (verifique). Para isso tivemos que introduzir seis dígitos redundantes em cada mensagem. Daremos agora um exemplo de como podemos mandar nossa mensagem corretamente enviando apenas três dígitos redundantes.

Dada a mensagem (a, b, c) enviamos a mensagem $(a, b, c, a + b, a + c, b + c)$. Aqui a soma é a soma "módulo 2", isto é, $0 + 0 = 0$, $0 + 1 = 1$, $1 + 0 = 1$, $1 + 1 = 0$. Vamos então mostrar que podemos recuperar a mensagem.

Suponha que o receptor recebe $(r_1, r_2, r_3, r_4, r_5, r_6)$ e ele já sabe que há um único erro. Ele procede do seguinte modo. Calcula $r_1 + r_2$. Se $r_1 + r_2 = r_4$ então r_1, r_2, r_4 estão corretos, logo $a = r_1$, $b = r_2$ e c é o digito que ocorrer duas vezes em $r_3, r_5 + a, r_6 + b$. Se $r_1 + r_2 \neq r_4$ um dos três está errado, logo r_3, r_5, r_6 estão certos. Logo $c = r_3$, $a = r_3 + r_5$, $b = r_3 + r_6$. Em ambos os casos o receptor recuperou (a, b, c).

A aritmética módulo 2 nos ajudou muito no exemplo acima. Isso é um fenômeno usual, os melhores códigos proveem de objetos com estrutura, por isso é bastante conveniente usar os corpos finitos (para uma referência completa aos corpos finitos ver [12]), o que faremos consistentemente no que segue. O nosso objetivo será mostrar como construir sistematicamente bons códigos corretores de erros e analisar sua "performance".

Os Capítulos 1 e 2 são básicos, neles introduzimos os conceitos e resultados que serão utilizados consistentemente no que segue. Os

capítulos seguintes descrevem construções diversas de códigos e podem ser lidos independentemente uns dos outros.

Com a exceção ao Capítulo 6, essas notas serão acessíveis a quem tiver conhecimento dos conceitos básicos de álgebra e álgebra linear, normalmente vistos na graduação. Mais especificamente usaremos os conceitos de corpos, anéis e espaços vetoriais e suas propriedades básicas. O Capítulo 6, por outro lado, necessita de conhecimentos de geometria algébrica (como, por exemplo, [6]) para sua compreensão. Meu gosto pessoal me "forçou" a incluí-lo e espero que os leitores que tenham lido o resto destas notas se sintam motivados a aprender o necessário para lê-lo.

O que me atrai na teoria dos códigos corretores de erros é como um problema, a princípio tão "aplicado", se relaciona tão intimamente com vários tópicos de matemática "pura" motivando novos problemas e novos resultados em ambos os lados. É essa inter-relação que procurei ilustrar nestas notas. Essas notas não se propõem a ser nem um livro texto ([13] é ótimo) nem uma obra de referência ([16] é a "bíblia" do assunto). Meu objetivo é apenas transmitir minha fascinação pelo assunto aos leitores e, quem sabe, motivá-los a serem usuários ou pesquisadores da teoria dos códigos.

Certamente, dado o tamanho do texto, muita coisa foi omitida. O leitor interessado deve então redirecionar-se a bibliografia e, se não estiver satisfeito, consultar a bibliografia de [16]. Devemos mencionar a omissão total do importante aspecto probabilístico da teoria dos códigos, o qual pode ser visto em [15]. Àqueles leitores que, por engano, acharam que íamos falar de criptografia, recomenda-se a consulta de [18].

1

GENERALIDADES

Denotaremos por $\mathbb{F}_q$ o corpo finito com q elementos. Um *código linear* C sobre o alfabeto $\mathbb{F}_q$ é um subespaço vetorial de $\mathbb{F}_q^n$. Se d é a dimensão de C sobre $\mathbb{F}_q$ dizemos que C é um $[n, d]$-código.

Há várias maneiras de se descrever um código, podemos, por exemplo, dar uma base $v_1, \ldots, v_d$ de C. Neste caso, se $v_i = (v_{i1}, \ldots, v_{in})$ então a aplicação $V : \mathbb{F}_q^d \to \mathbb{F}_q^n$ dada por

$$V(a_1, \ldots, a_d) = \sum_{i=1}^{d} a_i v_i = \left(\sum_{i=1}^{d} a_i v_{i1}, \ldots, \sum_{i=1}^{d} a_i v_{in} \right)$$

é um "codificador", isto é, pensando $\mathbb{F}_q^d$ como o conjunto das palavras numa linguagem "natural" a função V nos diz como codificar as palavras.

A matriz $V = (v_{ij})$ que descreve a aplicação linear V é chamada a *matriz geradora* do código. Diremos que V está na *forma padrão* se

$$V = (I_d P)$$

para uma matriz P, isto é, $v_{ij} = \delta_{ij}$ para $i, j = 1, \ldots, d$ (onde $\delta_{ij} = 0$ se $i \neq j$, $\delta_{ii} = 1$). Neste caso diremos que os primeiros d símbolos ou

coordenadas de $c \in C$ são os *símbolos de informação* e os restantes os *símbolos de controle*.

Porque símbolos de controle? Dado um vetor $x = (x_1, \ldots, x_n) \in \mathbb{F}_q^n$, para checar se $x \in C$ basta verificar se $x = V(a)$ para algum a e neste caso (estamos supondo C padrão) $x_j = a_j$, $j \leq d$ e $x_j = \sum_{i=1}^{d} a_i v_{ij} = \sum_{i=1}^{d} x_i v_{ij}$ para $j > d$. Logo, para checar se $x \in C$ basta verificar se $x_j = \sum_{i=1}^{d} x_i v_{ij}$, para $j = d+1, \ldots, n$.

Diremos que dois códigos C_1 e C_2 são *equivalentes* se pudermos obter C_2 a partir de C_1 por permutação das coordenadas. Pode-se provar que todo código é equivalente a um código que pode ser gerado por uma matriz na forma padrão (ver Exercício 9, p. 13).

Pelo que vimos acima a informação contida numa palavra $c \in C$ depende de d e de suas coordenadas e o resto é redundância, que é usada para controle. Definimos então a *razão de informação* de C, denotada por $i(C)$, como sendo n/d. Isso medirá então a razão entre o número de coordenadas "informativas" e o número total de coordenas.

Outra maneira de descrever um código é dar uma aplicação linear sobrejetiva $H : \mathbb{F}_q^n \to \mathbb{F}_q^{n-d}$ tal que o núcleo de H é C. Neste caso temos $(x_1, \ldots, x_n) \in C$ se, e somente se, $H(x_1, \ldots, x_n) = 0$. Se $H = (b_{ij})$ então a última equação se escreve

$$\sum_{j=1}^{n} h_{ij} x_j = 0, \quad i = 1, \ldots, n-d.$$

A matriz H é então chamada de *matriz de controle de paridade* de C.

Se C é dado pela matriz geradora $(I_d P)$ como acima, isto é, C é padrão, é fácil calcular a matriz de controle de C. De fato, temos $x \in C$ se e somente se $x_j = \sum_{i=1}^{d} x_i v_{ij}$, $j > d$, como vimos acima, logo a matriz controle é dada por $(-{}^tP I_{n-d})$ onde ${}^tP$ é a transposta de P.

Um exemplo ilustrativo que origina o nome "controle de paridade" é o código de controle de paridade sobre $\mathbb{F}_2$ definido pela matriz geradora $(I_n 1)$ onde $1 =^t (1, \ldots, 1)$, isto é, $C \subset \mathbb{F}_q^{n+1}$ dado por

$$\left\{(x_1, \ldots, x_n, x_1 + \cdots + x_n) \;\middle|\; (x_1, \ldots, x_n) \in \mathbb{F}_q^n\right\}.$$

A matriz controle de paridade de C é $(1, \ldots, 1)$. É fácil ver então que $x \in C$ se, e somente se, $\sum x_i = 0$, isto é, x tem um número par de coordenadas não nulas, daí o nome controle de paridade.

Falando em erros, mencionamos na introdução que os códigos seriam escolhidos de tal modo que duas palavras do código fossem sempre bem diferentes, para que quando recebêssemos uma mensagem com possíveis erros pudéssemos decodificá-la como a palavra no código mais parecida com a mensagem recebida. Para formalizar os conceitos de "diferente" e "parecido" definimos a *norma de Hamming* em $\mathbb{F}_q^n$ pondo para $x \in \mathbb{F}_q^n$,

$$|x| = \text{número de coordenadas não nulas de } x.$$

Vale então as seguintes propriedades:

1) $|x| = 0$ se e só se $x = 0$;

2) $|\lambda x| = |x|$ se $\lambda \in \mathbb{F}_q$, $\lambda \neq 0$;

3) $|x + y| \leq |x| + |y|$.

Definimos também a *distância de Hamming* pondo

$$d(x, y) = |x - y|,$$

que satisfaz:

1′) $d(x, y) = 0$ se e só se $x = y$;

2′) $d(x, y) = d(y, x)$;

3′) $d(x,z) \leq d(x,y) + d(y,z)$.

Note que $d(x,y)$ é o número de coordenadas onde x e y diferem, logo $d(x,y)$ mede quão diferentes são x e y. d é uma métrica em $\mathbb{F}_q^n$.

Note que 1′) segue de 1), 2′) de 2) (com $\lambda = -1$) e 3′) de 3). As propriedades 1) e 2) são imediatas. Provaremos a propriedade 3).

Sejam

$$I = \{i \in \{1,\ldots,n\} \mid x_i = 0\} \quad \text{e} \quad J = \{i \in \{1,\ldots,n\} \mid y_i = 0\}$$

então, por definição,

$$|x| = n - \#I \quad \text{e} \quad |y| = n - \#J.$$

Logo

$$|x| + |y| = 2n - (\#I + \#J).$$

Por outro lado, temos que $\#I + \#J = \#(I \cup J) + \#(I \cap J)$ e $\#(I \cup J) \leq n$, logo

$$|x| + |y| \geq n - (\#I \cap \#J).$$

Porém, se $i \in I \cap J$, temos que $x_i = y_i = 0$, logo $x_i + y_i = 0$. Então $x+y$ tem a i-ésima coordenada nula para $i \in I \cap J$, consequentemente

$$|x + y| \leq n - \#(I \cap J).$$

Isso conclui a demonstração. □

Já podemos então medir quanto as palavras de um código diferem umas das outras. Definimos então o *peso de um código* C, denotado por $w(C)$ pondo

$$w(C) = \min\{d(x,y) \mid x, y \in C,\ x \neq y\}.$$

Como o código é um subespaço, temos que $x - y \in C$ toda vez que $x, y \in C$, logo

$$w(C) = \min\{|x| \mid x \in C,\ x \neq 0\}.$$

Podemos então passar a correção e detecção de erros. Dizemos que um código C *corrige* e *erros* se para todo $y \in \mathbb{F}_q^n$ existe no máximo um único $x \in C$ com $d(x,y) \leq e$. Isso significa que ao recebermos uma mensagem y com no máximo e erros, isto é, y difere de algum elemento $x \in C$ em no máximo e coordenadas, esse elemento x sendo a mensagem enviada, então x é o único elemento de C tão próximo de y, logo podemos recuperar x a partir de y. Mais adiante veremos como implementar esse procedimento. Agora vamos ver quantos erros um código pode corrigir.

(1.1) Teorema: *Seja C um código de peso $w(C)$, então C corrige*

$$\left\lfloor \frac{w(C)-1}{2} \right\rfloor$$

erros.

Demonstração: Seja $e = \lfloor \frac{w(C)-1}{2} \rfloor$, então $2e+1 \leq w(C)$. Suponha que C não corrija e erros, e seja $y \in \mathbb{F}_q^n$ tal que existam $x_1, x_2 \in C$, $x_1 \neq x_2$, com $d(x_1,x_2) \leq e$, $i = 1,2$. Por 3′) temos que

$$d(x_1,x_2) \leq d(x_1,y) + d(y,x_2) \leq 2e.$$

Por outro lado, como $x_1 \neq x_2$, pela definição de $w(C)$ temos que

$$d(x_1,x_2) \geq w(C) \geq 2e+1,$$

contradição. □

Um resultado útil para se determinar $w(C)$ é o seguinte:

(1.2) Proposição: *Seja C um código com matriz de controle H e peso $w(C)$. Então quaisquer $w(C)-1$ colunas de H são linearmente independentes e existem $w(C)$ colunas de H linearmente dependentes.*

DEMONSTRAÇÃO: Seja s o inteiro tal que quaisquer s colunas de H são linearmente independentes e existem $s+1$ colunas de H linearmente dependentes.

Sejam $h_1, \ldots, h_n$ as colunas de H. Se $h_{i_1}, \ldots, h_{i_{s+1}}$ são linearmente dependentes, existem $c_{i_1}, \ldots, c_{i_{s+1}} \in \mathbb{F}_q$ com $\sum c_{i_j} h_{i_j} = 0$. Seja $c = (c_1, \ldots, c_n)$ definido por $c_i = c_{i_j}$ se $i \in \{i_1, \ldots, i_{s+1}\}$, $c_i = 0$ caso contrário. Então $\sum c_i h_i = 0$ e $c \in C$, porém c tem no máximo $s+1$ coordenadas não nulas, logo $w(C) \leq s+1$.

Se $w(C) < s+1$ existe $c \in C$, $c \neq 0$, com no máximo s coordenadas não nulas, digamos $c_i = 0$ se $i \neq i_1, \ldots, i_s$. Como $c \in C$, $\sum_{i=1}^{n} c_i h_i = 0$, logo $\sum_{j=1}^{s} c_{i_j} h_{i_j} = 0$ e então $h_{i_1}, \ldots, h_{i_s}$ são linearmente dependentes, isso contradiz a definição de s, logo $w(C) = s+1$, como queríamos demonstrar. □

(1.3) COROLÁRIO (SINGLETON): *Se C é um $[n,d]$-código, então*

$$w(C) \leq n - d + 1.$$

DEMONSTRAÇÃO: Seja H a matriz de controle de C. Como as colunas de H estão em $\mathbb{F}_q^{n-d}$, quaisquer $n-d+1$ colunas de H são linearmente dependentes. O resultado agora segue da proposição. □

Códigos com $w(C) = n - d + 1$ são chamados *códigos separáveis pela distância máxima* ou MDS (maximum distance separable). Eles tem uma descrição interessante com conjuntos satisfazendo certas propriedades geométricas em espaços projetivos sobre corpos finitos que discutiremos no Capítulo 4.

Passamos agora a dar um procedimento simples de decodificação.

Se C é um $[n,d]$-código dado como núcleo de $H : \mathbb{F}_q^n \to \mathbb{F}_q^{n-d}$. Se $x \in \mathbb{F}_q^n$ chamaremos $H(x)$ de *síndrome* de x. Para cada $v \in \mathbb{F}_q^{n-d}$ escolha e_v tal que $H(e_v) = v$ e tal que $|e_v|$ é mínima. e_v é chamado um líder da classe lateral $H^{-1}(v)$. e_v pode não ser único, mas fixemos

um em cada classe. Se recebermos uma mensagem y, calculamos $H(y) = v$ e tomamos $c = y - e_v$ como a decodificação de y.

Note primeiro que $H(c) = H(y) - H(e_v) = v - v = 0$, logo $c \in C$. Note também que $d(c, y) = |e_v|$. Como e_v foi escolhido minimizando a norma em $H^{-1}(v)$ temos que c é o elemento de C mais próximo de y. Consequentemente, se C corrige e erros, a decodificação nos dará a mensagem enviada toda vez que a mensagem recebida tem síndrome v satisfazendo $|e_v| \leq e$. Esse processo é chamado *decodificação por semelhança máxima.*

Em geral esse procedimento é muito custoso, pois nos obriga a calcular os e_v. Códigos com propriedades particulares podem ter algoritmos de decodificação mais eficientes. Encontraremos alguns exemplos disso mais adiante.

Exercícios

1. Considere o $[7, 4]$-código sobre $\mathbb{F}_2$ que tem a matriz de controle

$$\begin{pmatrix} 1 & 0 & 1 & 0 & 1 & 0 & 1 \\ 0 & 1 & 1 & 0 & 0 & 1 & 1 \\ 0 & 0 & 0 & 1 & 1 & 1 & 1 \end{pmatrix}.$$

   Este código é chamado $[7, 4]$*-código de Hamming*

   (a) Calcule o peso de C;

   (b) Calcule uma matriz geradora de C;

   (c) Calcule líderes para as classes laterais de C;

   (d) Escreve alguns elementos de $\mathbb{F}_2^7$ aleatoriamente e decodifique-os.

2. Considere o $[4, 2]$-código sobre $\mathbb{F}_3$ que tem a matriz geradora

$$\begin{pmatrix} 1 & 0 & 2 & 2 \\ 0 & 1 & 2 & 1 \end{pmatrix}.$$

(a) Calcule o peso de C;

(b) Calcule uma matriz de controle de C;

(c) Decodifique $(0,1,1,1)$, $(1,1,1,0)$ e $(0,0,2,2)$.

3. Sejam C e C' respectivamente $[n,d]$ e $[n',d']$-códigos. Considere o código $C \oplus C'$, que é um $[n+n', d+d']$-código. Prove que $w(C \oplus C') = w(C) + w(C')$ e calcule as matrizes geradoras e de controle de $C \oplus C'$ em função de C e C'.

4. Seja C um $[n,d]$-código. Se $\ell \leq d$ e $i_1, \ldots, i_\ell \in \{1, \ldots, n\}$, considere C' o código consistindo dos $c \in C$ tais que $c_{i_1} = \cdots = c_{i_\ell} = 0$. Considere, de maneira natural, C' como um código em $\mathbb{F}_q^{n-\ell}$. Mostre que $i_1, \ldots, i_\ell$ podem ser escolhidos tais que $\dim C' = d - \ell$ e $w(C') = w(C)$.

5. Seja C um $[n,d]$-código. Se $i_1, \ldots, i_r \in \{1, \ldots, n\}$, considere $A : \mathbb{F}_q^n \to \mathbb{F}_q^r$, $A(x_1, \ldots, x_n) = (x_{i_1}, \ldots, x_{i_r})$. Seja C' a imagem de C por A. Prove que se $r \leq n - d$ pode se escolher $i_1, \ldots, i_r$ tais que C' seja um $[r,d]$-código e que $w(C') = w(C) - n + r$.

6. Defina para $x, y \in \mathbb{F}_q^n$ o produto interno $\langle x, y \rangle = \sum_{i=1}^n x_i y_i$. Cuidado, o produto interno nada tem a ver com a norma. Dê um exemplo de $x \in \mathbb{F}_3^2$ com $x \neq 0$ e $\langle x, x \rangle = 0$. Se C é um código em $\mathbb{F}_q^n$ defina o *código dual*

$$C^\perp = \left\{ y \in \mathbb{F}_q^n \;\middle|\; \langle x, y \rangle = 0 \text{ para todo } x \in C \right\}.$$

Qual a relação entre as matrizes geradora e de controle de C e $C^\perp$? *

Dê um exemplo de um código C onde $C = C^\perp$. Calcule $C^\perp$ para C como nos Exercícios 1 e 2.

* A relação entre os pesos de C e $C^\perp$ em geral não é simples, mas temos o seguinte resultado devido a MacWilliams (ver, por exemplo, [13]).

Sejam para $C \subset \mathbb{F}_q^n$ um $[n,d]$-código, $A_i = \#\{x \in C \mid |x| = i\}$ e $P_C(r) = \sum_{i=1}^n A_i r^i$. Então

$$P_{C^\perp}(t) = q^{-d}(1 + (q-1)t)^n P_C\left(\frac{1-t}{1-(q-1)t}\right).$$