

## 4 Códigos LT Bidimensionais

Neste capítulo são apresentadas duas novas distribuições de graus: Sóliton Robusta Melhorada [24] e Sóliton Robusta Truncada para códigos LT sistemáticos [12], além de um esquema bidimensional para os códigos LT.

### 4.1 Distribuição de graus Sóliton Robusta Melhorada

Nesta seção é apresentada a distribuição Sóliton Robusta Melhorada (SRM), definida a partir da distribuição Sóliton Robusta de Luby (SR) que foi definida e estudada nos capítulos anteriores. Tee, Nguyen *et al.* observaram, em [24], a existência de alguns graus  $d_i$ , os quais têm uma probabilidade  $\mu(d_i)$  tão baixa, que o número de símbolos dado pelo produto de  $\mu(d_i)$  e  $k$  pode ser menor que 1, indicando a ausência de símbolos codificados com estes graus. Em alguns casos, a distribuição Sóliton Robusta, pode levar a um prematuro fracasso na decodificação e uma conseqüente perda de símbolos durante o processo de decodificação, a menos que o número de símbolos redundantes seja elevado. Quando o processo de decodificação é interrompido, devido à ausência de símbolos codificados de grau 1, precisamos receber mais símbolos para a recuperação de todos os símbolos originais.

Em vista disto, a proposta de Tee, Nguyen *et al.*, portanto, é melhorar o comportamento da distribuição Sóliton Robusta no caso em que  $\mu(d_i) \cdot k < 1$ , introduzindo um fator extra:

$$\nu = \sum_i \mu(d_i) \cdot k \quad (4-1)$$

com a finalidade de criar uma distribuição de graus mais benéfica, onde  $d_i$  representa o termo de grau- $i$  da distribuição, que satisfaz as seguintes condições:

$$\begin{cases} \left( \frac{1}{d(d-1)} + \frac{\bar{R}}{kd} \right) \cdot \frac{k}{\beta} < 1 & \text{para } 2 \leq d \leq \left( \frac{k}{\bar{R}} - 1 \right), \\ \left( \frac{1}{d(d-1)} \right) \cdot \frac{k}{\beta} < 1 & \text{para } \left( \frac{k}{\bar{R}} + 1 \right) \leq d \leq k. \end{cases} \quad (4-2)$$

Logo, determina-se o parâmetro  $\nu$  que é um fator complementar para a distribuição Sóliton Robusta original, com a ajuda do conjunto de desigualdades dada por (4-2), para maximizar a frequência relativa de ter símbolos de grau 1, ou seja, todos aqueles graus  $d_i$  que satisfazem (4-2) têm redefinidas as suas probabilidades de distribuição a zero e a soma de todas essas probabilidades de distribuição são adicionadas à distribuição de probabilidade de grau 1. A distribuição Sóliton Robusta Melhorada é mostrada na Figura 4.1.

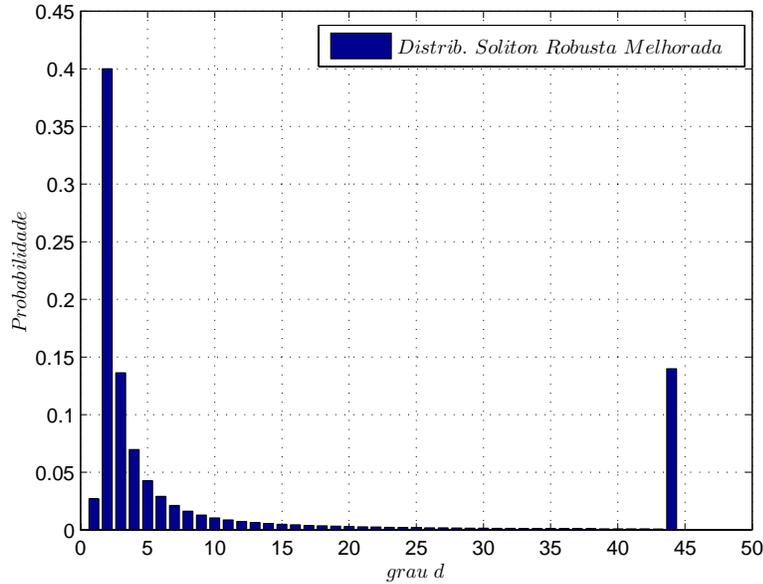


Figura 4.1: A distribuição Sóliton Robusta Melhorada para o caso  $k = 10000$ ,  $c = 0.2$  e  $\delta = 0.1$

É importante fazer uma comparação entre a distribuição Sóliton Robusta e a distribuição Sóliton Robusta Melhorada. Conforme mostram os resultados de simulação, examinados adiante, quando é utilizada a distribuição SR de Luby, aproximadamente 1130 símbolos codificados (*overhead*  $\approx 13\%$ ) são necessários para recuperar o arquivo original formado por  $k = 1000$  símbolos de entrada. Por outro lado, quando a distribuição SRM de Nguyen é utilizada, o número de símbolos necessários foi reduzido a uma média de 1105 (*overhead*  $\approx 10.5\%$ ).

A seguir, mostramos nas Figuras 4.2 e 4.3 os histogramas normalizados do número de símbolos necessários para a decodificação com sucesso de um código LT, utilizando as distribuições de graus anteriormente mencionadas. Os resultados desses histogramas foram obtidos após 1000 simulações.

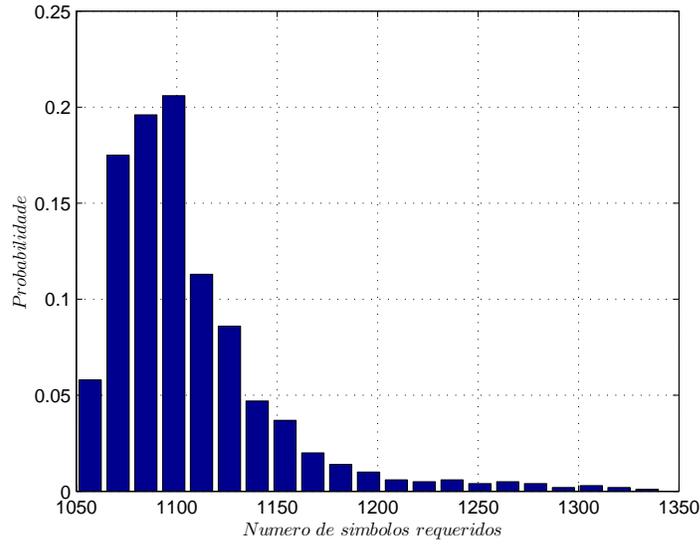


Figura 4.2: Histograma do número de símbolos necessários para a decodificação com sucesso de um código LT com distribuição Sóliton Robusta Melhorada ( $k = 1000$ ,  $c = 0.03$  e  $\delta = 0.1$ ).

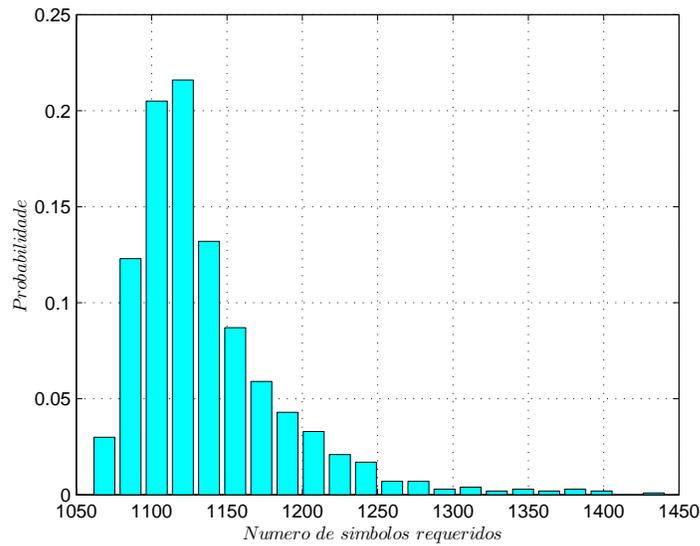


Figura 4.3: Histograma do número de símbolos necessários para a decodificação com sucesso de um código LT com distribuição Sóliton Robusta ( $k = 1000$ ,  $c = 0.03$  e  $\delta = 0.1$ ).

Também foram feitas outras simulações, para comprovar o melhor desempenho da distribuição Sóliton Robusta Melhorada em relação à Sóliton Robusta. O resultado destas simulações é mostrado na Figura 4.4, onde foi calculado a taxa de apagamento de símbolo, ou seja, a probabilidade de um símbolo de entrada não ser recuperado após a decodificação de  $(1+\epsilon)k$  símbolos de saída, em função da capacidade  $(1-P_a)$  do canal BEC. As simulações foram realizadas para 1000 blocos de entrada, com comprimento  $k = 1000$  e *overhead*  $\epsilon = 30\%$ .

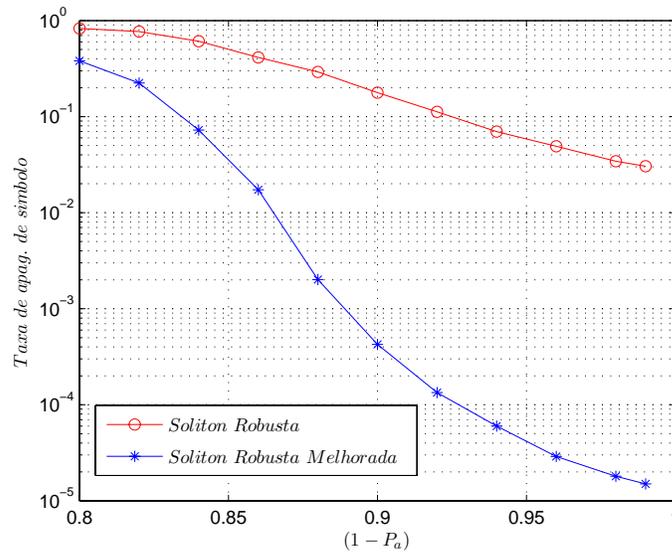


Figura 4.4: Desempenho das distribuições Sóliton Robusta e Sóliton Robusta Melhorada para um código LT(1000,1300) com parâmetros  $c = 0.03$  e  $\delta = 0.1$ .

É importante mencionar que essa grande diferença no desempenho entre as duas distribuições, é devido à quantidade de símbolos de entrada que consegue-se recuperar quando ocorre uma falha na decodificação. Por exemplo, para  $P_a = 0.1$ , quando usamos a distribuição Sóliton Robusta e é registrada uma falha na decodificação, a média de símbolos de entrada que são recuperados é muito baixa (aproximadamente 12%). Por outro lado, quando é usada a distribuição Sóliton Robusta Melhorada, a média de símbolos de entrada recuperados é maior (aproximadamente 94%).

## 4.2

### Distribuição Sóliton Robusta Truncada para Códigos LT Sistemáticos

Na distribuição Sóliton Robusta, há duas condições que devem ser satisfeitas para conseguir recuperar todos os símbolos de entrada a partir dos símbolos codificados que são recebidos pelo decodificador. Primeiramente, o

número de símbolos codificados deve satisfazer à condição  $n \geq k + 2 \ln\left(\frac{\bar{R}}{\delta}\right)\bar{R}$ . Como segunda condição, o número de símbolos que possuem o mais alto grau deve obedecer a  $d \geq \frac{k}{\bar{R}}$  [6], [8]. Nguyen *et al.* quando desenvolverem um código LT sistemático em [11], viram que estas condições continuavam válidas para a decodibilidade de tais códigos com taxas maiores que  $R = \frac{1}{2+\epsilon}$ , onde  $\epsilon = 2 \ln\left(\frac{\bar{R}}{\delta}\right)$  é o *overhead* do código LT original. Entretanto, para taxas menores que  $R = \frac{1}{2+\epsilon}$ , essas condições não são suficientes, já que as densidades de ambas as matrizes, geradora e de paridade, dos códigos LT sistemáticos são baixas.

No intuito de aumentar essas densidades, a distribuição de graus usada para gerar os símbolos de paridade dos códigos LT sistemáticos, denominada distribuição Sóliton Robusta Truncada (SRT), foi redefinida da seguinte forma [12]:

$$\Omega(d) = \mu(d, \gamma, \nu) = \begin{cases} \frac{1}{\beta'} \left[ \frac{1}{k} + \frac{\bar{R}}{k} + \nu(d) \right] & \text{para } d = \gamma, \\ \frac{\gamma}{\beta'} \left[ \frac{1}{d\left(\frac{d}{\gamma}-1\right)} + \frac{\bar{R}}{kd} \right] & \text{para } d = 2\gamma, 3\gamma, \dots, \frac{\gamma \cdot k}{\bar{R}} - 1, \\ \frac{\bar{R}}{\beta' k} \left[ \ln\left(\frac{\bar{R}}{\delta}\right) + \frac{1}{\left(\frac{k}{\bar{R}}-1\right)} \right] & \text{para } d = \frac{\gamma \cdot k}{\bar{R}}, \\ 0 & \text{para } d > \frac{\gamma \cdot k}{\bar{R}} \text{ e } d = 1, \end{cases} \quad (4-3)$$

onde  $k$  é o número total de símbolos de informação originais,  $\bar{R}$  é o número de símbolos com um grau específico  $\gamma$ , que satisfaz a condição de [6]  $\bar{R} \equiv c \ln\left(\frac{k}{\delta}\right) \sqrt{k}$ . Além do mais,  $\nu$  representa o fator extra que garante a decodibilidade da distribuição SRM, já visto na Equação (4-1). Ainda em referência à Equação (4-3), temos

$$\beta' = \sum_d [(\rho(d) + \tau(d)) + \nu(\gamma)],$$

onde  $\gamma$  é um número inteiro maior que 1. Mantendo-se o grau máximo em  $d_{max} = \frac{\gamma \cdot k}{\bar{R}}$ , garantimos que os símbolos originais de entrada serão representados no grupo de símbolos de paridade codificados do código LT sistemático.

A Figura 4.5 mostra a distribuição SRT, na qual o eixo das abscissas possui um passo de tamanho  $\gamma = 6$  e um valor máximo de  $d_{max} = \frac{\gamma \cdot k}{\bar{R}} = \lfloor \frac{6 \times 10000}{230} \rfloor = 260$ . Note que, assim como na Figura 4.1, utilizou-se  $c = 0.2$  para uma melhor visualização do gráfico.

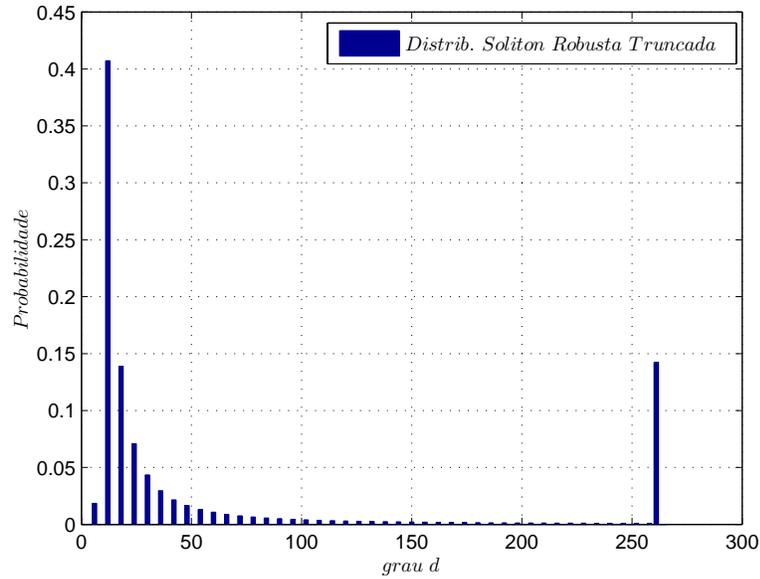


Figura 4.5: A distribuição Sóliton Robusta Truncada  $\Omega(d)$  para o caso  $k = 10000$ ,  $c = 0.2$ ,  $\delta = 0.1$  e  $\gamma = 6$ .

A média dos graus dos símbolos sistemáticos codificados é dada pela seguinte equação:

$$\bar{D} = \sum_d \frac{d \cdot (\rho(d) + \tau(d) + \nu(d))}{\beta'} + 1. \quad (4-4)$$

### 4.3 Esquema de códigos LT bidimensionais

No intuito de melhorar o desempenho dos códigos LT, foi desenvolvida a idéia de códigos LT bidimensionais, onde os símbolos de entrada são rearranjados em forma de uma matriz, como mostra a Figura 4.6.

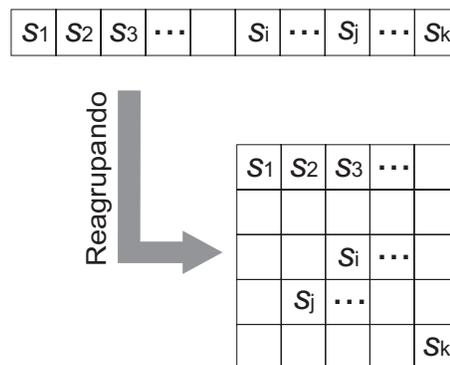


Figura 4.6: Reagrupamento bidimensional dos símbolos de entrada  $s_k$ .

### 4.3.1

#### Codificação bidimensional

No esquema anterior onde os símbolos de entrada estão rearranjados em uma matriz, os blocos correspondentes às linhas da matriz são codificados usando um código LT sistemático. Por trabalhar horizontalmente, denominamos este código de LT horizontal ( $LT_H$ ) que introduz um *overhead* em cada linha. Em seguida, as colunas da matriz resultante também são codificadas usando um código LT não sistemático ( $LT_V$ ), agora verticalmente. O código  $LT_V$ , por sua vez, introduz um *overhead* em cada coluna. A Figura 4.7 ilustra este processo.

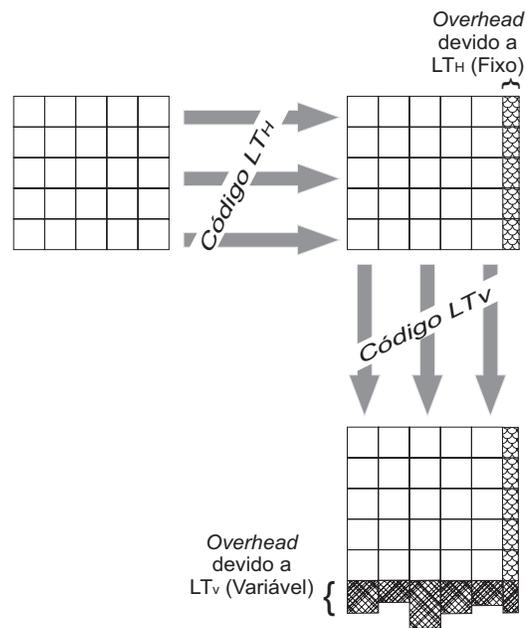


Figura 4.7: Processo de codificação e decodificação bidimensional.

Apesar de a Figura 4.7 mostrar que o *overhead vertical* é variável, neste ponto da codificação ele também é fixo. Torna-se variável na decodificação, quando o decodificador não consegue recuperar os símbolos de entrada originais. Uma explicação mais detalhada é dada na decodificação de códigos LT bidimensionais.

### 4.3.2

#### Decodificação bidimensional

A decodificação bidimensional é feita em duas etapas. Primeiro faz-se uma decodificação vertical, coluna a coluna de cada bloco de saída vertical para tentar recuperar os símbolos codificados da matriz intermediária, sendo alguns deles apagados devido a que não foi possível sua recuperação. Logo,

faz-se uma decodificação horizontal linha a linha na matriz intermediária para tentar recuperar os símbolos de entrada que formam a matriz original.

Quando não consegue-se recuperar todos os símbolos de entrada (a matriz original), ou seja, quando ocorre uma falha na decodificação total, o receptor requer mais símbolos de saída. Estes símbolos extras são adicionados a cada bloco de saída vertical dependendo onde for necessário. Por isso, o *overhead vertical* de cada bloco de saída vertical passa a ser variável (vide Figura 4.7), sendo necessário determinar uma média de tal *overhead*, para assim calcular o *overhead total* do sistema que é definido como:

$$(1 + \epsilon_{Total}) = (1 + \epsilon_{Horizontal}) \cdot (1 + \epsilon_{Vertical}) \quad (4-5)$$

Após o requerimento de símbolos adicionais, o decodificador realiza novamente as decodificações vertical e horizontal e o processo se repete até que todos os símbolos de entrada sejam recuperados.

### 4.3.3 Configurações Utilizadas

Neste trabalho, duas configurações do sistema bidimensional dos códigos LT são utilizadas, conforme a Tabela 4.1.

Tabela 4.1: Configurações Bidimensionais

<i>Códigos LT Utilizados</i>	<b>2D-TSR</b>	<b>2D-TSRM</b>
Sistemático com distribuição SRT	$LT_H$	$LT_H$
Não-sistem. com distribuição SR	$LT_V$	—
Não-sistem. com distribuição SRM	—	$LT_V$

Como pode-se ver na Tabela 4.1, ambas as configurações bidimensionais apresentam o mesmo código  $LT_H$ , que é um código LT sistemático [12], utilizando a distribuição de graus Sóliton Robusta Truncada (SRT). A diferença entre elas está no código  $LT_V$ . A Configuração 2D-TSR faz uso, como  $LT_V$ , de um código LT convencional (não-sistemático) [6], com a distribuição Sóliton Robusta (SR). Já a Configuração 2D-TSRM, utiliza o mesmo código como  $LT_V$ , porém com a distribuição Sóliton Robusta Melhorada (SRM).

## 4.4 Resultados obtidos

Nesta seção são apresentados os resultados das simulações realizadas, com o intuito de avaliar o desempenho dos códigos LT bidimensionais. Para isto,

foram comparadas as duas configurações bidimensionais com dois esquemas de codificação tradicionais (unidimensionais). Para facilitar a leitura das legendas dos gráficos que mostram os resultados obtidos, vamos a considerar a seguinte nomenclatura:

- **1D–SR**: configuração que corresponde à codificação com um código LT não-sistemático com distribuição **Sóliton Robusta**.
- **1D–SRM**: configuração que corresponde à codificação com um código LT não-sistemático com distribuição **Sóliton Robusta Melhorada**.
- **2D–TSR**: configuração que corresponde à codificação bidimensional com o código  $LT_H$  sendo LT sistemático com distribuição Sóliton Robusta Truncada e o código  $LT_V$  sendo LT não-sistemático com distribuição **Sóliton Robusta**.
- **2D–TSRM**: configuração que corresponde à codificação bidimensional com o código  $LT_H$  sendo LT sistemático com distribuição Sóliton Robusta Truncada e o código  $LT_V$  sendo LT não-sistemático com distribuição **Sóliton Robusta Melhorada**.

Todas as simulações foram obtidas para canal  $BEC$ , com 1000 blocos de tamanho  $k = 1000$  símbolos sendo que, nos casos bidimensionais (configurações 2D–TSR e 2D–TSRM), estes  $10^6$  símbolos de entrada foram agrupados em uma matriz (1000x1000), conforme a Figura 4.6.

A primeira simulação realizada buscou determinar o *overhead* ( $\epsilon$ ) requerido para conseguir uma decodificação bem sucedida usando a configuração 2D–TSR. A Figura 4.8 mostra o resultado de tal simulação em função da probabilidade ( $P_a$ ) de apagamento do canal  $BEC$ . Fazendo uma comparação deste resultado com a configuração 1D–SR, percebe-se que a proposta bidimensional teve um desempenho pior. Por exemplo, observa-se que, com  $P_a = 0.01$ , a configuração 1D–SR requer  $\epsilon \approx 13\%$ , enquanto a configuração 2D–TSR requer  $\epsilon \approx 18\%$ . Essa diferença de 5% é aproximadamente mantida ao longo dos demais valores de  $P_a$ .

Logo, comparou-se a configuração 2D–TSRM com os resultados das duas configurações que foram apresentadas na Figura 4.8. O resultado dessa comparação é mostrado na Figura 4.9. Pode-se perceber que a configuração 2D–TSRM apresenta uma pequena melhora de desempenho em relação à configuração 1D–SR. Porém, na Figura 4.10, percebe-se que, para altos valores de  $P_a$ , mais precisamente para  $P_a > 0.1$ , a configuração 2D–TSRM requer praticamente o mesmo *overhead* que a configuração 1D–SRM.

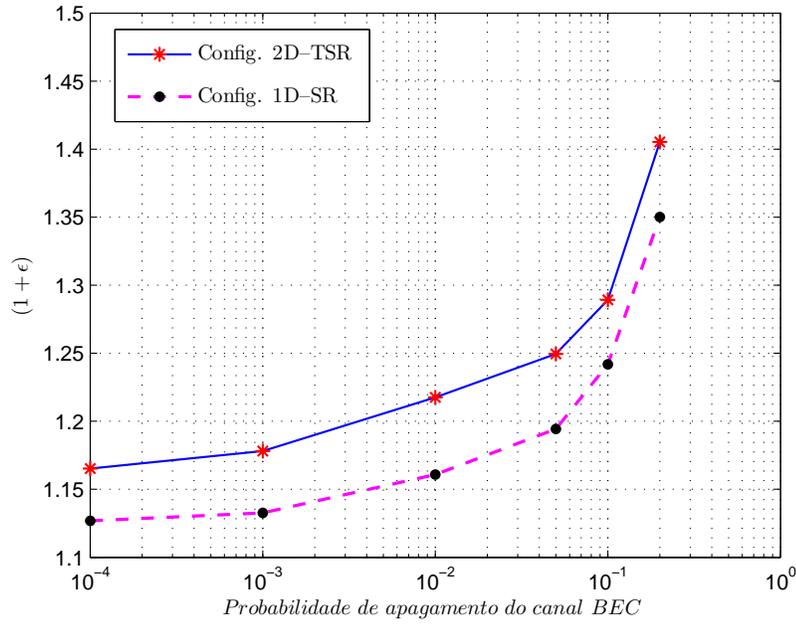


Figura 4.8: Comparação de desempenho entre as configurações que usam a distribuição SR. Parâmetros utilizados:  $k = 1000$ ,  $c = 0.03$ ,  $\delta = 0.1$  e  $\gamma = 6$ .

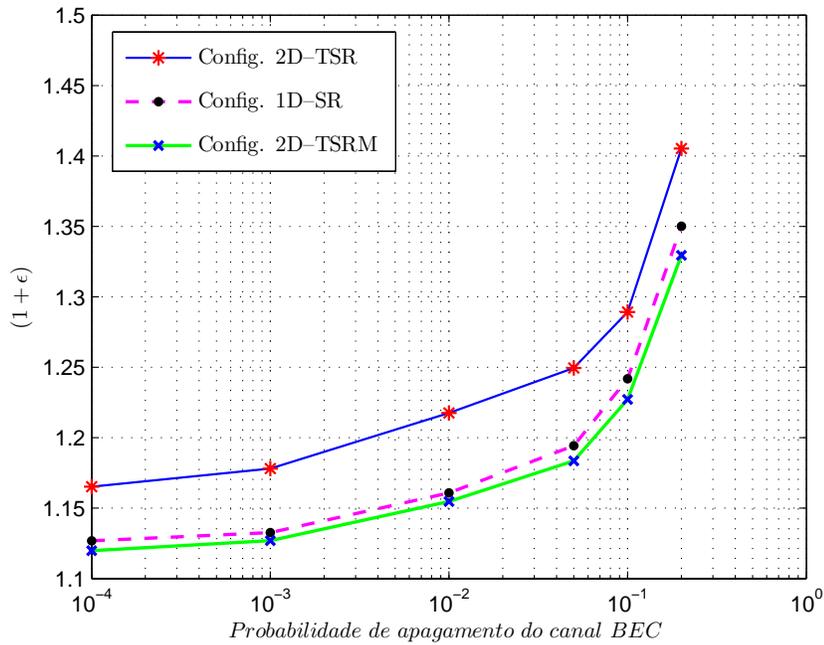


Figura 4.9: *Overhead* ( $\epsilon$ ) requerido para uma decodificação bem sucedida, em função de  $P_a$ . Parâmetros utilizados:  $k = 1000$ ,  $c = 0.03$ ,  $\delta = 0.1$  e  $\gamma = 6$ .

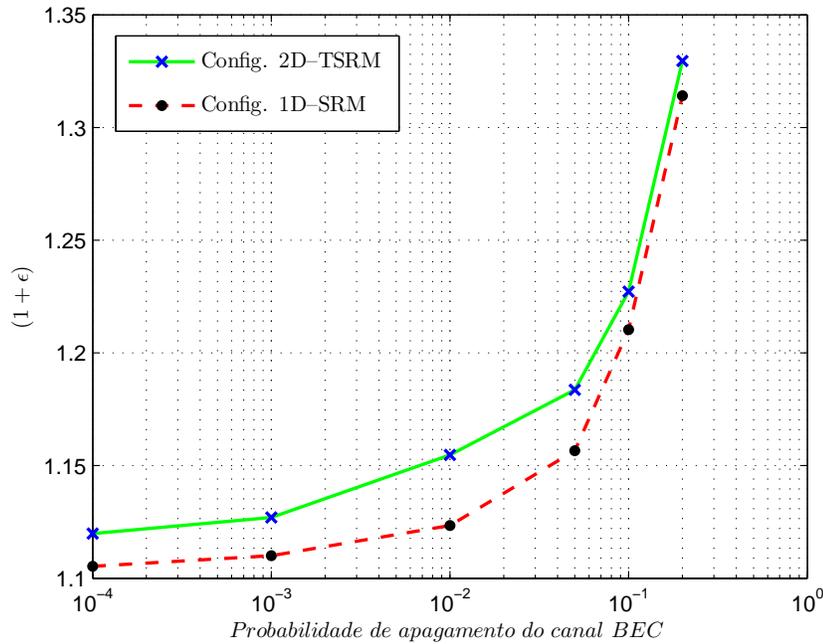


Figura 4.10: Comparação de desempenho entre as configurações que usam a distribuição SRM. Parâmetros utilizados:  $k = 1000$ ,  $c = 0.03$ ,  $\delta = 0.1$  e  $\gamma = 6$ .

Vale observar que, nas Figuras 4.8, 4.9 e 4.10, para as configurações unidimensionais (1D-SR e 1D-SRM) é transmitido um *overhead*  $\epsilon = 5\%$ . Nos casos bidimensionais (configurações 2D-TSR e 2D-TSRM), os *overheads* transmitidos inicialmente são:  $\epsilon_H = 5\%$  e  $\epsilon_V = 5\%$ . Em caso de falha no processo de decodificação, o decodificador requisita que mais símbolos sejam enviados (aumentando o *overhead vertical*), até que a decodificação tenha sucesso. Portanto, nas configurações bidimensionais o *overhead vertical* é variável. O *overhead total*, isto é, o previamente transmitido mais o requerido pelo decodificador, é que esta sendo calculado e mostrado no eixo vertical dos gráficos mencionados anteriormente.

Finalmente, na Figura 4.11 compara-se a taxa de apagamento de símbolo em função da capacidade  $(1 - P_a)$  do canal BEC, para as quatro configurações. Nestas simulações, utilizou-se um *overhead* fixo  $\epsilon_{Total} = 30\%$ . Sendo que, nos casos bidimensionais, o *overhead* foi utilizado da seguinte forma:  $\epsilon_H = 5\%$  e  $\epsilon_V = 24\%$  calculado de acordo à Equação (4-5), ambos fixos. Então, no decodificador, contabilizou-se os símbolos de entrada que não foram recuperados após a decodificação, gerando-se a taxa de apagamento do gráfico.

Além disso, pode-se observar na Figura 4.11, que as configurações que fazem uso da distribuição Sóliton Robusta (1D-SR e 2D-TSR) apresentam uma alta taxa de apagamento de símbolos, mesmo quando o canal apresenta boas condições (baixos valores de  $P_a$ ). Por outro lado, as configurações que usam a distribuição Sóliton Robusta Melhorada (1D-SRM e 2D-TSRM), têm seus desempenhos melhorados à medida que a qualidade do canal aumenta. Por exemplo, com  $P_a = 0.12$ , tem-se que a taxa de apagamento de símbolo da configuração 2D-TSRM é de  $7 \times 10^{-2}$ , enquanto a da configuração 1D-SRM é  $2 \times 10^{-3}$ . Porém, observa-se que, para  $P_a < 0.06$ , a configuração 2D-TSRM proposta passa a apresentar desempenho melhor que o esquema convencional 1D-SRM.

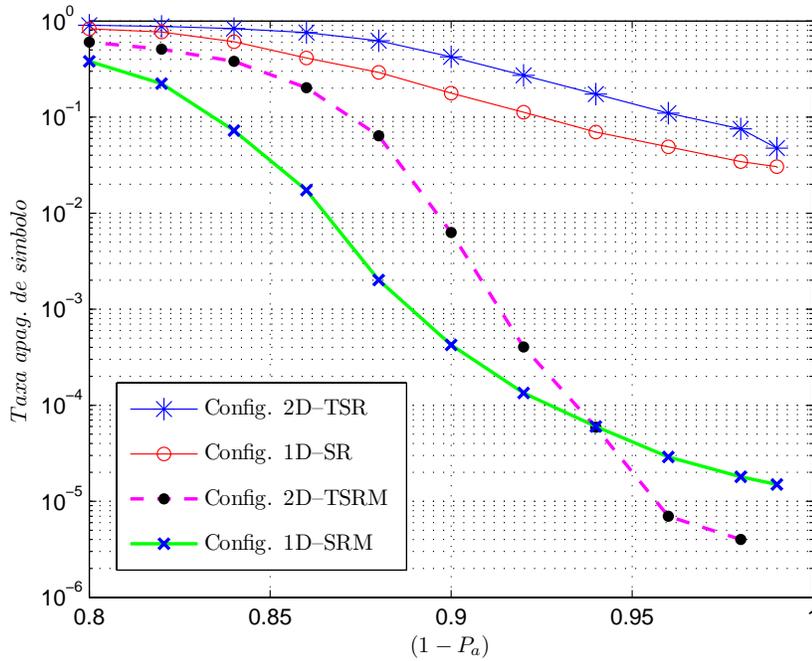


Figura 4.11: Taxa de apagamento de símbolo em função da capacidade do canal  $(1 - P_a)$ . Parâmetros utilizados:  $k = 1000$ ,  $c = 0.03$ ,  $\delta = 0.1$ ,  $\gamma = 6$  e  $\epsilon = 30\%$ .

A melhora de desempenho do esquema bidimensional 2D-TSRM proposto em relação ao esquema unidimensional 1D-SRM, mesmo que apenas para determinada condição de canal ( $P_a < 0.06$ ), mostra que a idéia do esquema bidimensional pode ser aperfeiçoada para a obtenção de melhores resultados.