

Decodificação Iterativa Conjunta Fonte-Canal

Alexandre de Andrade, Jaime Portugheis e Fernando Pujaco Rivera

Resumo—Nesse artigo, apresentamos uma aplicação da estrutura de grafos-fatores no problema de decodificação iterativa conjunta fonte-canal. Partimos de um modelo genérico de fonte com memória, discreta no tempo e contínua em amplitude e consideramos quantização vetorial, tratando o problema da decodificação iterativa conjunta através da integração dos modelos destes componentes com o resto do sistema. Alguns resultados de simulações para os esquemas propostos são descritos.

Palavras-Chave—Decodificação conjunta fonte-canal, decodificação iterativa, quantização vetorial.

Abstract—In this article we present an application of the factor graph framework to the problem of joint iterative source-channel decoding. We start with a generic model of a discrete-time, continuous amplitude source with memory. We consider vector quantization and treat the joint iterative decoding problem by integrating these component models with those of the rest of the system. Some simulation results for the proposed schemes are described.

Keywords—Joint source-channel decoding, iterative decoding, vector quantization.

I. INTRODUÇÃO

Nesse artigo, consideramos uma instância do problema geral de transmissão digital de uma fonte contínua em amplitude. Em particular, como modelar o sistema de tal forma que ele admita decodificação iterativa conjunta fonte-canal. Partimos de um modelo genérico de fonte, um processo estocástico discreto no tempo e contínuo em amplitude, que passa por quantização vetorial e indexação binária, e os bits resultantes são adicionalmente protegidos por codificação para uma transmissão através de um canal ruidoso. A idéia é fornecer uma visão geral de como todos os elementos de uma decodificação vetorial, podem ser modelados para uma decodificação iterativa conjunta.

Através da teoria de grafos-fatores [1], será vislumbrada uma completa caracterização do que seriam as variáveis do sistema e suas inter-relações, tudo sendo sintetizado na fatoração de um função custo conjunta entre todas elas, que eventualmente seria a distribuição conjunta entre as variáveis, tal que uma decodificação por marginalização aproximada se torna realizável pelo algoritmo soma-produto iterativo num grafo com ciclos.

Portanto, várias conceitos são abordados: decodificação com grafos-fatores envolvendo variáveis contínuas, modelos para a fonte baseados em processos estocásticos discretos lineares, incorporando-os aos grafos-fatores da rede de decodificação proposta, quantização vetorial e mapeamento binário com seus mapas inseridos no grafo-fator do sistema. As principais

referências com as quais nosso estudo se relaciona são [2], [3], [4], [5], [6], [7] e principalmente [8]. No final, apresentamos resultados simulados para decodificação iterativa conjunta.

II. ESQUEMA DE CODIFICAÇÃO E DECODIFICAÇÃO CONJUNTA

O sistema pode ser ilustrado na figura abaixo, um sistema mais completo de comunicação digital, representado analiticamente de forma conveniente pelos seguintes componentes: fonte, quantizador vetorial, mapeador entre vetores e palavras binárias (“bit assignment”), codificador de canal, e canal.

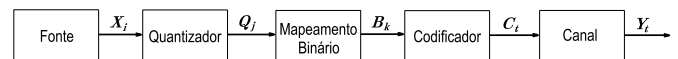


Fig. 1. Esquematização do sistema para decodificação conjunta.

A fonte emite símbolos X_l ($l = 0, 1, \dots$), um processo aleatório estacionário discreto no tempo e contínuo em amplitude. Esse processo é separado em blocos de m símbolos e quantizado vetorialmente por mapa $\psi : \mathbb{R}^m \rightarrow \{0, \dots, n-1\}$, que resulta na sequência Q_j ($j = 0, 1, \dots$), um processo discreto com alfabeto $\{0, \dots, n-1\}$. Acoplado ao quantizador consideramos o mapeamento injetor para associar índices $\{0, \dots, n-1\}$ a palavras binárias de comprimento s , ou seja, elementos em $\{0, 1\}^s$, (taxa resultante do sistema de quantização é s/m bits por amostra). Eventualmente, esse mapeamento pode adicionar redundância para proteção contra erros. Denotamos B_k a sequência de bits emitidos serialmente pelo sistema de quantização. Um codificador toma a sequência de bits B_k ($k = 0, 1, \dots$) como entrada e emite serialmente a sequência de bits codificados C_t ($t = 0, 1, \dots$) sinalizados como $\{-1, +1\}$ para um canal BI-AWGN. Y_t é a sequência de observações ruidosas na saída do canal. O objetivo final do sistema de codificação e decodificação é transmitir o processo X_l ($l = 0, 1, \dots$) com a mínima distorção possível.

De forma geral, um modelo de uma fonte genérica é um processo que emite símbolos discretos no tempo e contínuos na amplitude, com alguma propriedade de estacionaridade. Por simplicidade vamos nos restringir a processos auto-regressivos e/ou médias-móveis (processos A.R.M.A.) definidos a seguir. Podemos destacar o tipo mais simples deles, os processos de Gauss-Markov gerados pela equação

$$X_{t+1} = \alpha X_t + W_t, \quad (1)$$

onde W_i é um processo gaussiano de média zero e variância constante, com $\alpha < 1$. São processos caracterizados por apenas um parâmetro. O parâmetro α representa o grau de

Jaime Portugheis e Fernando Pujaco Rivera, Departamento de Comunicações, FEEC, UNICAMP, E-mails: <jaime,fpujaico>@decom.fee.unicamp.br, Alexandre de Andrade, Departamento de Estatística, UFPE, E-mail: wwwxooxww@gmail.com. Este trabalho foi parcialmente financiado pela FAPESP (01/07955-9).

determinismo do processo. Para processos deste tipo a função de auto-correlação é simplesmente

$$E [X_t X_{t-s}] = \alpha^s \quad (2)$$

e portanto tem decaimento exponencial.

Tipicamente, apenas processos em que $\alpha \geq 0.85$ a decodificação conjunta é compensadora. Para valores abaixo disto, o custo-benefício da decodificação iterativa não é bom.

O algoritmo básico para a obtenção de quantizadores vetoriais ótimos é o chamado algoritmo GLA (“generalized Lloyd algorithm”), conforme estabelecido em [9]. Há métodos eficientes de escolha dos vetores código iniciais para o algoritmo GLA. O mais reconhecido é dado também em [9]. O algoritmo GLA provido deste método é o denominado algoritmo LBG (Linde-Buzo-Gray). Serve para obtenção de dicionários de cardinalidade 2^n . Em nosso sistema e em nossas simulações, consideramos sempre quantizadores LBG.

Dado o quantizador vetorial conforme descrito anteriormente, voltamos a considerar o seu papel no sistema proposto. Podemos convencionar para os índices o conjunto finito $I = \{0, \dots, n-1\}$, tal que a taxa resultante do quantizador é $\log_2 n/m$ bits por símbolo.

Como componente do esquema proposto, o quantizador vetorial gera a partir do processo contínuo $X_l \in \mathbb{R}$ ($l = 0, 1, \dots$) um processo derivado discreto, a sequência temporal de índices $Q_j \in I$ ($j = 0, 1, \dots$).

Para quantização da sequência de um processo X_l , consideramos o particionamento em blocos de m símbolos consecutivos do tipo $(X_{jm}, \dots, X_{j(m+m-1)})$, sendo emitido o índice $Q_j \in I$, dado pelo mapeamento

$$Q_j = \chi(X_{jm}, \dots, X_{j(m+m-1)}) \quad (3)$$

Como X_l é um processo com memória, o processo discreto Q_j também possui algum tipo de memória. Quando, X_l é um processo ARMA de ordem pequena, menor que a dimensão do quantizador, ou seja $p < m$, uma boa aproximação é assumir que Q_j é um processo de Markov de primeira ordem estacionário, e estimar com sequências de treinamento suas probabilidades

$$P(Q_j = s, Q_{j+1} = t)_{s \in I, t \in I}$$

invariantes em j . O processo consiste em simular o processo X_l , obter o processo Q_j correspondente, e registrar as frequências relativas. A partir da conjunta, são naturalmente obtidas

$$P(Q_j = s)_{s \in I}$$

e as probabilidades de transição

$$P(Q_{j+1} = t | Q_j = s)_{s \in I, t \in I}.$$

A partir das distribuições, as entropias relevantes $H(Q_j)$ e $H(Q_{j+1}|Q_j)$ podem ser obtidas. A entropia $H(Q_{j+1}|Q_j)$ pode ser obtida por

$$H(Q_{j+1}|Q_j) = H(Q_j, Q_{j+1}) - H(Q_j). \quad (4)$$

A importância da condicional $P(Q_{j+1} = t | Q_j = s)$ é servir como função custo para a decodificação forward-backward do

processo Q_j . Passando para a notação de grafos-fatores, denotamos simplesmente $p(q_{j+1}|q_j)$, onde os argumentos definem as variáveis nas quais a função está definida.

Recapitulando, por definição a entropia do processo é

$$\lim_{J \rightarrow \infty} \frac{1}{J} H(Q_1, \dots, Q_J).$$

Assumindo a estacionaridade do processo, vale a igualdade

$$\lim_{J \rightarrow \infty} \frac{1}{J} H(Q_1, \dots, Q_J) = H(Q_{j+1}|Q_j).$$

Portanto, a quantização do processo X_l resulta no processo Q_j , onde a entropia é reduzida, e $H(Q_{j+1}|Q_j)$ é uma aproximação desta entropia resultante.

Evidentemente, para considerar a taxa de entropia na saída do quantizador “por símbolo da fonte”, devemos considerar o ajuste $H(Q_{j+1}|Q_j) / m$.

Os processos X_l e Q_j estão relacionados por (3). Para um decodificador que obtenha probabilidades a posteriori $\beta(q_j)_{q_j \in I}$ de Q_j , a respectiva estimação de mínima distorção quadrática (MSE) para os símbolos da fonte associados utiliza o “mapa inverso” do quantizador φ ,

$$(\hat{x}_{jm}, \dots, \hat{x}_{j(m+m-1)}) = \frac{\sum_{q_j \in I} \beta(q_j) \varphi(q_j)}{\sum_{q_j \in I} \beta(q_j)}.$$

Ou seja, a estimação é uma média ponderada dos vetores código $\beta(q_j)$. Uma média ponderada dos centróides, no caso de um quantizador ótimo.

Para a medida de distorção quadrática, a performance do sistema é usualmente avaliada pela relação sinal-ruído entre a sequência X_l original e a reconstrução no destino \hat{X}_l dada por

$$SNR = 10 \log_{10} \frac{\sum_l x_l^2}{\sum_l (x_l - \hat{x}_l)^2}. \quad (5)$$

Em nosso sistema e em nossas simulações, consideramos sempre o algoritmo BSA (Binary Switch Algorithm) para obtenção de mapeamentos [10].

Consideramos um mapeador (“index assignment”) por blocos que a cada vetor Q_j associa um grupo de bits $\{B_{s_j}, \dots, B_{s_j+s-1}\}$ respectivamente alinhados no índices temporais. É qualquer mapa injetor $\Gamma : \{0, \dots, 2^r - 1\} \rightarrow \{0, 1\}^s$, que associa a cada símbolo r -ário vindo do quantizador uma palavra binária de comprimento s , onde $r \leq s$. Aqui, admitimos a possibilidade de mapeadores que adicionam redundância controladamente, e por isso temos $r \leq s$. Na verdade, isso é equivalente a um mapeador ordinário acoplado a um código de bloco (s, r) acoplado em sua saída.

O projeto do mapeador é feito em conjunto com o quantizador vetorial. Na prática, o que é feito é definir as palavras código resultantes desse código, e a partir de algoritmos de mínima distorção no quantizador vetorial define-se a associação entre os vetores e essas palavras código. De acordo com nossas simulações, o algoritmo BSA resulta em mapeamentos eficientes, sem grandes variações de desempenho na distorção média entre dois mapas localmente ótimos quaisquer.

O esquema geral permite concatenações turbo de qualquer natureza na codificação de canal para proteção de erros. Aqui, consideramos duas possibilidades.

Uma delas é o caso em que o codificador de canal tem embutido um código turbo padrão, com dois geradores de paridade convolucionais junto à emissão dos bits não codificados, entrelaçamentos, puncionamentos, etc.

Outra possibilidade considera o fato de já haver outros componentes concatenados seriamente que têm memória e redundância, e portanto não há necessidade de uma composição turbo (dois componentes de paridades internos para interagir na decodificação). Basta portanto que o codificador de canal tenha um componente com memória, e a decodificação iterativa se processa na concatenação serial com os componentes relacionados à fonte (quantizador e mapeamento). Neste caso, temos uma liberdade maior de onde podemos alocar a redundância disponível, ou para codificação de canal ou para proteção dos índices da fonte, através de mapeamento binário com redundância.

Códigos binários convolucionais recursivos, truncados para um dado comprimento de bloco, são usuais para uma decodificação turbo. Anteriormente ao código de treliça, deve haver acoplado um entrelaçador de bloco para quebrar a memória sequencial da redundância dos bits oriundos da quantização-fonte. Analogamente aos sistemas turbo usuais, o entrelaçador define o atraso de codificação e decodificação do sistema como um todo. Em nossas simulações, foram gerados vários entrelaçadores pseudo-aleatórios do tipo *S*-random pelo eficiente algoritmo de Crozier [11].

Em relação aos códigos empregados, aqui não necessariamente temos códigos sistemáticos. De fato, é comum a utilização de uma codificação convolucionária recursiva de taxa 1, ou seja, apenas um gerador de paridade a partir dos bits entrelaçados. Isso não resulta em queda de desempenho, contanto que a taxa extra seja alocada para adicionar redundância no mapeador binário.

III. REPRESENTAÇÃO POR GRAFOS-FATORES

Uma vez descritos os componentes, podemos esquematizar o grafo-fator genérico do sistema proposto. Uma primeira versão é dada na figura 2. Embora com algumas simplificações para viabilizar seu desenho, a generalidade é mantida. Grafos-fatores com outros parâmetros podem ser deduzidos a partir deste. No esquema simples, o comprimento de bloco da codificação de canal é $K = 15$, consideramos uma quantização vetorial com $m = 3$ dimensões, e uma fonte modelada por um AR de primeira ordem, um Gauss-Markov por exemplo. O mapeamento binário considera 2 bits para cada índice. A figura representa portanto um esquema de codificação fonte com $\frac{2}{3}$ bits/símbolo.

Pela esquema da figura, fica ilustrado que os ciclos no grafo-fator do sistema considerado se estendem desde o codificador de canal até a própria fonte e seus nós do modelo de transição de estados.

IV. A DECODIFICAÇÃO SUB-ÓTIMA TURBO

O sistema de decodificação deve reconstruir a sequência X_l dada a sequência de observações Y_t . Considere a decodificação

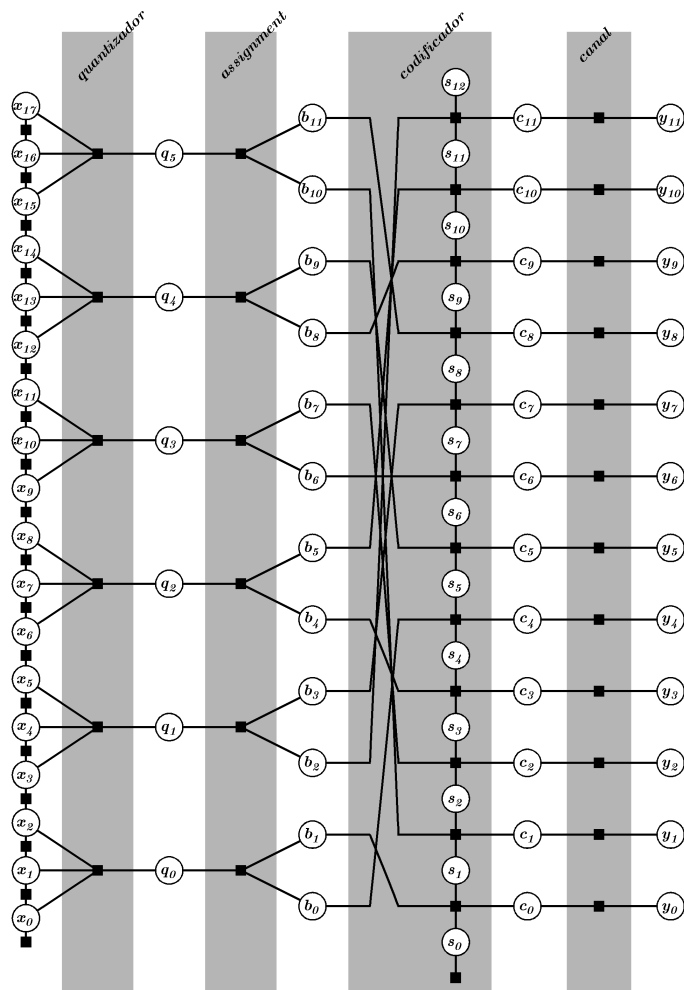


Fig. 2. Grafo-fator básico do sistema completo para decodificação conjunta, com modelo de quantização vetorial integrada.

realizada em blocos de comprimento L . Fixado o bloco $\{X_0, \dots, X_{L-1}\}$, sejam as seqüências truncadas correspondentes $\{Q_0, \dots, Q_{J-1}\}$, $\{B_0, \dots, B_{K-1}\}$, $\{C_0, \dots, C_{T-1}\}$ e $\{Y_0, \dots, Y_{T-1}\}$ internas ao bloco. Por construção, temos $L/K = m/r$ e K/T é igual à taxa do sistema de codificação-canal.

Dadas as variáveis observadas $\{Y_0 = y_0, \dots, Y_{T-1} = y_{T-1}\}$, considerando como medida de distorção o erro quadrático médio, a decodificação ótima é a que minimiza o valor esperado da distorção, condicionada à seqüência de variáveis observadas. Seja z_l o valor estimado de X_l , $\{z_0, \dots, z_{L-1}\}$ são tais que

$$D = \min_{\{z_0, \dots, z_{L-1}\}} E \left[\sum_{l=0}^{L-1} \frac{(X_l - z_l)^2}{L} \mid y_0, \dots, y_{T-1} \right], \quad (6)$$

onde $\{y_0, \dots, y_{T-1}\}$ são os valores observados.

Portanto, pelo critério de erro quadrático médio mínimo,

$$z_l = E[X_l \mid y_0, \dots, y_{T-1}], \quad l = 0, \dots, L-1 \quad (7)$$

Antes de desenvolver esta expressão e chegar ao que queremos, devemos considerar a expressão dos centróides associados ao codificador de fonte dado.

Considerando os primeiros elementos da sequência X_l , o mapa de quantização $\chi : R^m \rightarrow \{0, \dots, 2^r - 1\}$ faz a associação $(X_0, \dots, X_{m-1}) \rightarrow Q_0$.

O conjunto de centróides $\{E[(X_0, \dots, X_{m-1}) | Q_0 = q], q = 0, \dots, 2^r - 1\}$ são intrínsecos ao sistema de codificação. Sua expressão é dada por

$$E[(X_0, \dots, X_{m-1}) | q_0] = \int (x_0, \dots, x_{m-1}) p(x_0, \dots, x_{m-1} | q_0) dx_0 \dots dx_{m-1} \quad (8)$$

Para cada um desses vetores, cada coordenada pode ser expressa separadamente. Como exemplo, tomando a primeira coordenada correspondente a X_0 , vem

$$\begin{aligned} E[X_0 | q_0] &= \int x_0 p(x_0, \dots, x_{m-1} | q_0) dx_0 \dots dx_{m-1} \quad (9) \\ &= \int x_0 p(x_0 | q_0) dx_0 \end{aligned}$$

Voltando à expressão da decodificação ótima e fixando a análise à z_0 , o primeiro elemento da sequência de símbolos estimados,

$$\begin{aligned} z_0 &= E[X_0 | y_0, \dots, y_{T-1}] \quad (10) \\ &= \int x_0 p(x_0 | y_0, \dots, y_{T-1}) dx_0 \end{aligned}$$

Como vale, aproximadamente,

$$\sum_{q_0 \in \{0, \dots, 2^r - 1\}} p(x_0 | y_0, \dots, y_{T-1}) = \sum_{q_0 \in \{0, \dots, 2^r - 1\}} p(x_0 | q_0) p(q_0 | y_0, \dots, y_{T-1}), \quad (11)$$

então

$$z_0 = \frac{\int x_0 \sum_{q_0 \in \{0, \dots, 2^r - 1\}} p(x_0 | q_0) p(q_0 | y_0, \dots, y_{T-1}) dx_0}{\sum_{q_0 \in \{0, \dots, 2^r - 1\}} [\int x_0 p(x_0 | q_0) dx_0] p(q_0 | y_0, \dots, y_{T-1})}$$

O problema central na decodificação portanto reduz-se à obtenção da distribuição de probabilidade $p(q_0 | y_0, \dots, y_{T-1})$. Dada esta distribuição, basta usá-la em uma média ponderada dos centróides para obtenção de z_0 . Essa dedução vale por analogia para qualquer z_l .

A distribuição de probabilidade $p(q_0 | y_0, \dots, y_{T-1})$ vem da decodificação iterativa entre as sequências de variáveis $\{Q_0, \dots, Q_{J-1}\}$, $\{B_0, \dots, B_{K-1}\}$, $\{C_0, \dots, C_{T-1}\}$ e $\{Y_0, \dots, Y_{T-1}\}$ internas ao bloco. Se o quantizador usado é de bloco e sem memória, obtemos uma sequência Q_j com memória residual. A obtenção da distribuição $P(Q_j | Q_{j-1})$ pode ser obtida analiticamente ou por simulações numéricas.

V. A DECODIFICAÇÃO SUB-ÓTIMA E CONTÍNUA

Idealmente, podemos ilustrar um caso da decodificação conjunta iterativa no contexto de grafos fatores envolvendo todas as variáveis no grafo acima, inclusive as contínuas. Evidente que nesse caso, teremos nós em que o algoritmo somaproduto resulta em integrações em vez de somas. Portanto a implementação é proibitiva. O grafo-fator da figura 3 ilustra a transmissão por um sistema digital da fonte dada por um processo auto-regressivo (AR) de primeira ordem, quantização vetorial bidimensional, mapeamento binário, codificador e canal. Considere as variáveis X_i evoluindo de acordo com

um processo de Gauss-Markov da forma. Numa decodificação iterativa conjunta, cabe ao decodificador-fonte realizar uma inferência sequencial do processo aleatório (forward-backward com variáveis contínuas) e trocar informação probabilística símbolo a símbolo com o decodificador-canal.

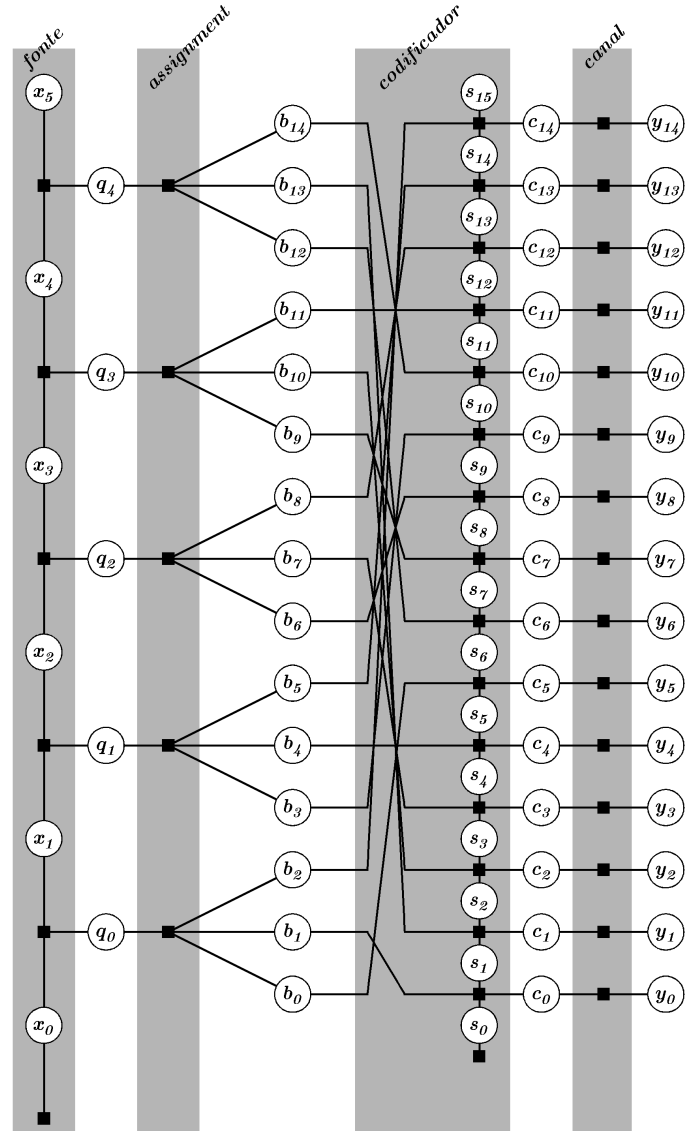


Fig. 3. Grafo-fator num caso particular, para ilustrar a decodificação incluindo variáveis contínuas da fonte.

Podemos descrever resumidamente o conjunto de funções envolvidas em cada 'nó' da fatoração pelo grafo da distribuição conjunta.

A quantização associa a variável discreta $Q_i \in \{0, \dots, 2^r - 1\}$ a cada par de variáveis (X_i, X_{i+1}) . Na figura acima, temos o caso particular $r = 3$ ilustrado.

Dessa forma, fica representado no mesmo 'node' duas funções: a correlação entre variáveis X_{i+1} e X_i

$$Pr\{x_{i+1} | x_i\} =^{def} Pr\{X_{i+1} = x_{i+1} | X_i = x_i\},$$

e o mapa da quantização bidimensional

$$Pr\{q_i | x_i, x_{i+1}\} =^{def} Pr\{Q_i = q_i | X_i = x_i, X_{i+1} = x_{i+1}\}.$$

A função resultante $T(x_i, x_{i+1}, q_i)$ no ‘node’ do grafo é o produto

$$T(x_i, x_{i+1}, q_i) = Pr\{x_{i+1}|x_i\} Pr\{q_i|x_i, x_{i+1}\}$$

No mapeamento cada Q_i é mapeado em variáveis binárias $\{B_{ri}, \dots, B_{ri+r-1}\}$. A função no “node” envolve apenas variáveis discretas, dada por

$$T(q_i, b_{ri}, \dots, b_{ri+r-1}) = Pr\{b_{ri}, \dots, b_{ri+r-1}|q_i\}.$$

Para o entrelaçador, o codificador e o canal as funções T envolvidas são as usuais,

$$T(b_j, s_i, s_{i+1}, c_i) = Pr\{s_{i+1}|b_j, s_i\} Pr\{c_i|b_j, s_i\}$$

e

$$T(c_i, y_i) = Pr\{y_i|c_i\}$$

Daqui ficam definidas as equações do algoritmo soma-produto aplicado ao esquema. Particularmente, no bloco “fonte-quantização”, as equações são

$$\alpha(x_{i+1}) = \sum_{q_i=0}^{2^r-1} \int_{x_i=-\infty}^{+\infty} T(x_i, x_{i+1}, q_i) \beta(q_i) \alpha(x_i),$$

$$-\infty < x_{i+1} < +\infty;$$

$$\beta(x_i) = \sum_{q_i=0}^{2^r-1} \int_{x_{i+1}=-\infty}^{+\infty} T(x_i, x_{i+1}, q_i) \beta(q_i) \beta(x_{i+1}),$$

$$-\infty < x_i < +\infty;$$

$$\alpha(q_i) = \int_{x_i=-\infty}^{+\infty} \int_{x_{i+1}=-\infty}^{+\infty} T(x_i, x_{i+1}, q_i) \alpha(x_i) \beta(x_{i+1}),$$

$$q_i = \{0, \dots, 2^r - 1\}.$$

Terminada a decodificação iterativa, à variável X_i é fornecida a densidade marginal $\alpha(x_i) \beta(x_i)$, e a decisão de mínima distorção quadrática é a decisão pela média dessa densidade,

$$z_i = \int_{x_i=-\infty}^{+\infty} \alpha(x_i) \beta(x_i).$$

Caberia portanto ao decodificador fonte realizar uma estimação sequencial do processo aleatório (forward-backward com variáveis contínuas) e trocar informação probabilística símbolo a símbolo com o decodificador de canal, através do mapa de quantização.

O exposto acima ilustra a complexidade envolvida para implementar a decodificação iterativa conjunta em sua plenitude. No caso acima, com quantização vetorial bidimensional, temos integrais duplas.

Na literatura, há tentativas de uma implementação simplificada e aproximada do algoritmo soma-produto em grafos envolvendo variáveis contínuas (em que o algoritmo passa a ser “integral-produto”). O método natural de implementação (integração numérica) resulta numa complexidade proibitiva em se tratando de um algoritmo iterativo.

A complexidade piora à medida que queremos uma quantização vetorial em mais dimensões. a decodificação completa do esquema da figura 2 envolveria ainda integrais multidimensionais.

VI. REDUÇÃO AO CASO DISCRETO

O exposto na seção anterior descreve uma decodificação que considera todas as variáveis na distribuição conjunta associada à fatoração, inclusive as variáveis contínuas X_l . Como visto na seção IV, podemos reduzir a análise à obtenção das distribuições a posteriori

$$p(q_j|y_0, \dots, y_{T-1}).$$

Ou seja, o problema geral de decodificação conjunta entre variáveis discretas e contínuas pode ter sua complexidade reduzida, onde a única hipótese é que vale a aproximação dada na fórmula 11.

No contexto de grafos fatores, isso é equivalente a dizer que podemos simplificar a distribuição conjunta utilizada. Em vez de $p(x_0, \dots, x_{L-1}, q_0, \dots, q_{J-1}, b_0, \dots, b_{K-1}, c_0, \dots, c_{T-1}, y_0, \dots, y_{T-1})$, a decodificação reduz-se a obter marginais a partir de

$$p(q_0, \dots, q_{J-1}, b_0, \dots, b_{K-1}, c_0, \dots, c_{T-1}, y_0, \dots, y_{T-1}).$$

O esquema inicial proposto da figura 2 pode ser traduzido num novo esquema (equivalente ou aproximadamente equivalente), representado na figura 4 abaixo, onde a redundância residual é modelada nas variáveis discretas (no caso dos esquemas acima, as variáveis Q_k).

Heuristicamente, estamos considerando que a estatística (modelo) do processo contínuo X_k induz a estatística do processo discreto Q_k de forma bem definida (função do mapa de quantização). Seria possível em princípio por cálculo analítico até derivar o comportamento do processo Q_k a partir de X_k . Mas mesmo quando isso não é realizável, pode-se extrair o comportamento de Q_k estatisticamente, por seqüências de treinamento. A redundância residual fica assim implícita no modelo do processo Q_k , e apenas esse modelo será envolvido no algoritmo de decodificação iterativa. De fato, devido ao processo de codificação digital de fonte (redução da taxa de entropia), somente a redundância residual pode auxiliar na decodificação conjunta.

A decodificação conjunta iterativa é então realizada apenas utilizando informação do processo Q_k , aplicando a este o algoritmo “forward-backward”. Após a saturação do algoritmo iterativo, o algoritmo dá como resultado probabilidades a posteriori $p(q_0|y_0, \dots, y_{T-1})$. A decisão nas variáveis X_k é obtida por critério de erro quadrático médio mínimo, através da equação 12.

VII. SIMULAÇÕES

A partir do esquema acima, curvas de desempenho por simulações podem então ser obtidos envolvendo codificadores fonte com quantização vetorial, seja isoladamente ou conjunta com o decodificador de canal. Destacamos aqui os alguns obtidos por cálculos computacionais simulando alguns sistemas com parâmetros típicos.

Um primeiro estudo relevante são simulações comparando as curvas de desempenho entre duas situações: decodificação conjunta iterativa entre decodificador fonte e decodificador canal e a decodificação iterativa apenas no decodificador de canal. São curvas de SNR de fonte vs SNR de canal.

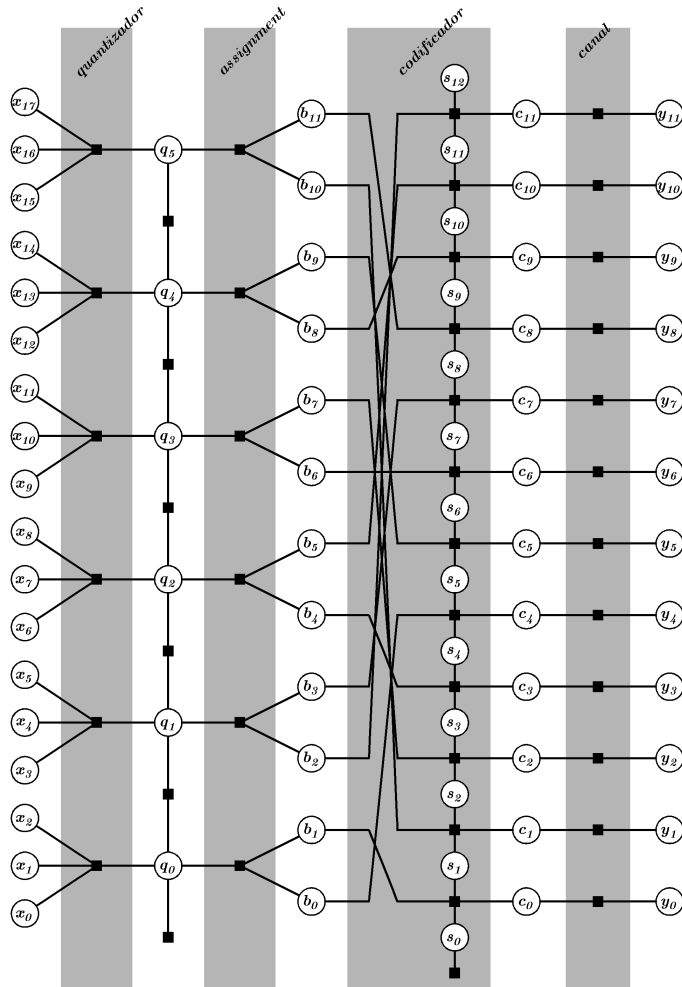


Fig. 4. Nova versão para o grafo-fator do sistema proposto, com a aproximação e redução ao caso da modelagem por $p(q_{j+1}|q_j)$.

A figura 5 abaixo apresenta curvas de desempenho obtidas por simulações do algoritmo soma-produto no grafo-fator da concatenação nas duas situações. A fonte é Gauss-Markov com coeficiente $\alpha = 0.9$. O quantizador vetorial codifica com 4 bits/4 amostras, e portanto o bloco usa 4096 amostras da fonte. O codificador de canal é um turbo (concatenação paralela com entrelaçador) de taxa 1/3 de 16 estados com geradores de paridade $(1 + D^3 + D^4) / (1 + D + D^2 + D^4)$ e comprimento de bloco 16384 bits. A distorção quadrática média resultante é dada em dB, em função da relação sinal ruído do canal E_b/N_0 (dB). As curvas mostram os desempenhos para 3, 6, 9, 12, e 15 iterações nos dois casos.

As curvas já estão transladadas no eixo E_b/N_0 considerando a entropia H do processo Q_j como redundância intrínseca, calculada a partir das transições $P(Q_{j+1}|Q_j)$ obtidas por seqüências de treinamento suficientemente extensas. No caso do sistema acima, a entropia residual é $H = 3.13202$. O ajuste é da forma

$$(E_b/N_0)_{dB}^* = (E_b/N_0)_{dB} - 10\log_{10}(H/r),$$

onde E_b/N_0 é o valor padrão considerando apenas o codificador de canal, obtido a partir da taxa e o valor direto no

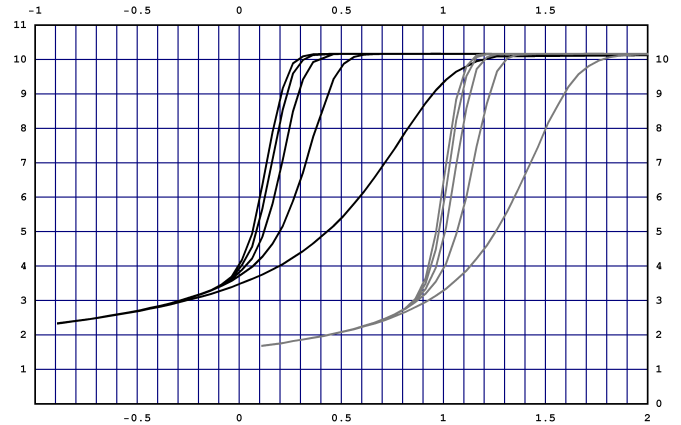


Fig. 5. Curvas de SNR (dB) de reconstituição da fonte vs SNR (dB) no canal, para decodificação conjunta (mais escuras) e não conjunta (mais claras), e diferentes números de iterações.

canal E_s/N_0 , que para sinalização binária $\{+1, -1\}$ no canal gaussiano de variância σ^2 .

Na comparação entre as curvas, vemos que a integração da informação da fonte na decodificação iterativa resulta em ganhos significativos. Em $E_b/N_0 = 0.3dB$, a região de maiores discrepâncias, obtemos um ganho pela decodificação combinada da ordem de $0.8dB$ em distorção quadrática da fonte.

Também são interessantes análises de ganho entre diferentes complexidades do codificador de fonte. Para um mesmo codificador de canal e fixada a taxa do codificador fonte, a figura 6 abaixo ilustra os resultados de desempenho para $m/r = 3/3, 4/4, e 5/5$. As entropias dos três quantizadores vetoriais são respectivamente $H = 2.51869, 3.50569, e 4.48972$. O processo X_l é um Gauss-Markov de coeficientes $\alpha = 0.8$. O codificador de canal é um turbo com geradores de paridade $(1 + D^3 + D^4) / (1 + D + D^2 + D^4)$ e comprimento de bloco 16380.

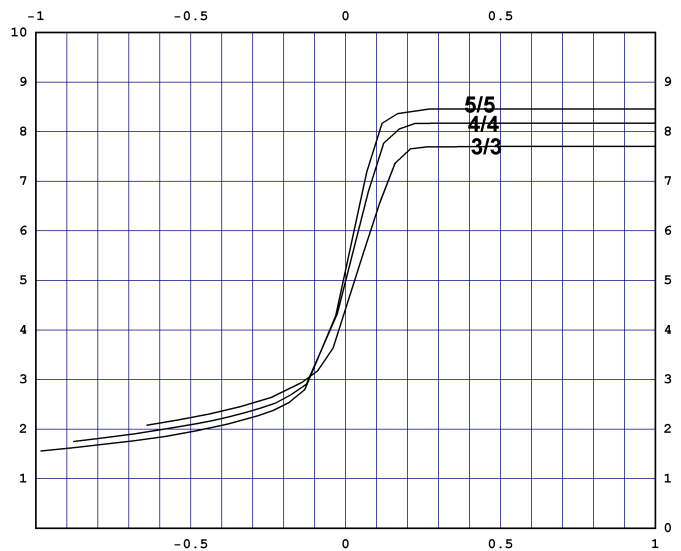


Fig. 6. Curvas de SNR (dB) de reconstituição da fonte vs SNR (dB) no canal para diferentes codificadores fonte, taxa 1bit/símbolo fixa.

Aqui, diferentes taxas implicam em considerável aumento na complexidade envolvida na decodificação. Quanto maior r , mais estados no decodificador.

As simulações mais interpretativas são os casos em que o processo X_k é Gauss-Markov (auto-regressivo de primeira ordem). Entretanto, o método de aproximação vale também para fontes com processos auto-regressivos e/ou médias-móveis (ARMA) quaisquer. Ainda que não seja possível obter analiticamente de forma exata a estatística que o processo contínuo X_k induz no processo discreto Q_k , é sempre possível obter aproximadamente por simulação das sequências. Como ilustração, a seguir a simulação para um processo ARMA estacionário dado por $X_{l+1} = W_l + 0.41X_l + 0.27X_{l-1} + 0.17X_{l-2}$.

O codificador de canal é um turbo de taxa 1/2 (puncionado) e 16 estados com geradores de paridade $(1 + D^4) / (1 + D + D^2 + D^3 + D^4)$ e comprimento de bloco 16380 bits. A figura 7 abaixo apresenta os resultados de desempenho para $m/r = 3/3$ e $4/4$. As entropias de transição para os símbolos emitidos pelo quantizador são $H = 2.35269bits$ e 3.37614 , respectivamente.

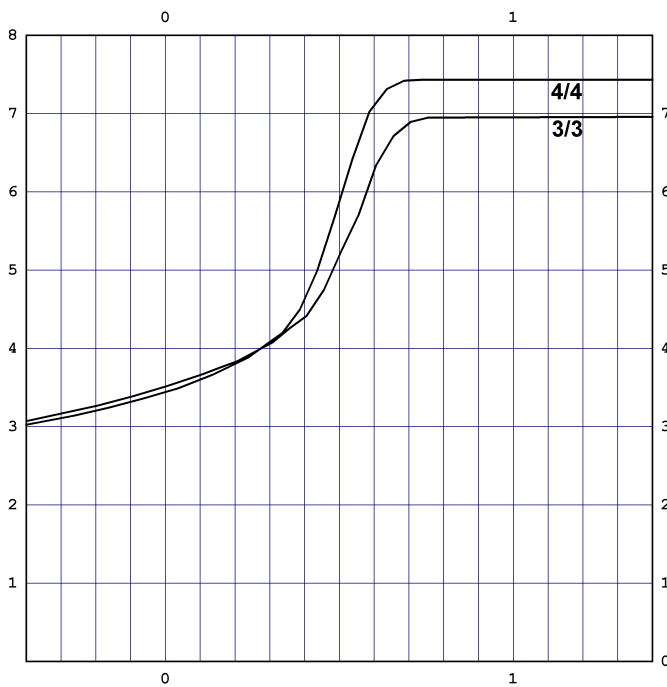


Fig. 7. Curvas de SNR (dB) de reconstituição da fonte vs SNR (dB) no canal na decodificação de um processo AR. Dois casos de complexidade e redundância residual no codificador de fonte.

VIII. CONCLUSÕES

Os exemplos acima apresentados evidenciam a amplitude das vantagens da decodificação iterativa conjunta. Além disso, nas simulações percebe-se que não há necessidade do recurso da proteção desigual de erros nos bits emitidos pelo quantizador. Isso deve-se à inserção da informação de redundância do processo Q_j na decodificação. Analogamente à decodificação sub-ótima turbo padrão, o desempenho é próximo à capacidade de canal.

REFERÊNCIAS

- [1] F. R. Kschischang & B. J. Frey & H.-A. Loeliger, *Factor graphs and the sum-product algorithm*, IEEE Transactions on Information Theory, vol. 47, no. 2, pp. 498-519, February 2001.
- [2] M. Adrat & P. Vary & T. Clevorn, *Optimized bit rate allocation for iterative sourcechannel decoding and its extension towards multi-mode transmission*, Proceedings of IST Mobile and Wireless Communications, pp. 1153-1157, June 2005.
- [3] T. Clevorn & P. Vary & M. Adrat, *Iterative source-channel decoding using short block codes*, Proceedings of IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP), vol. 4, pp. 221-224, May 2006.
- [4] T. Clevorn & L. Schmalen & P. Vary & M. Adrat, *On Redundant Index Assignments for Iterative Source-Channel Decoding*, IEEE Communications Letters, vol. 12, no. 7, July 2008.
- [5] N. Farvardin & V. Vaishampayan, *Optimal quantizer design for noisy channels: An approach to combined source-channel coding*, IEEE Transactions on Information Theory, vol. IT-33, pp. 827-838, November 1987.
- [6] J. Garcia-Frias & J. D. Villasenor, *Joint Turbo Decoding and Estimation of Hidden Markov Sources*, IEEE Journal on Selected Areas in Communications, vol. 19, no. 9, pp. 1671-1679, September 2001.
- [7] N. Goertz, *On the Iterative Approximation of Optimal Joint Source-Channel Decoding*, IEEE Journal on Selected Areas in Communications, vol. 19, pp. 1662-1670, September 2001.
- [8] D. J. Miller & M. Park, *A sequence-based approximate MMSE decoder for source coding over noisy channels using discrete hidden Markov models*, IEEE Journal Transactions on Communications, vol. 46, pp. 222-231, February 1998.
- [9] Y. Linde & A. Buzo & R. Gray, *An Algorithm for Vector Quantizer Design*, IEEE Transactions on Communications, vol. 28, January 1980.
- [10] K. Zeger & A. Gersho, *Pseudo-Gray coding*, IEEE Transactions on Communications, vol. 38, No. 12, pp. 2147-2158, 1990.
- [11] S. Crozier, *New High-Spread High-Distance Interleavers for Turbo-Codes*, Proceedings of the 20th Biennial Symposium on Communications, pp. 3-7, May 2000.