

Predição não-linear de amostras com tabulação adaptativa *in situ*

José Alexandre Nalon e Yuzo Iano

Resumo—A predição de séries temporais é uma atividade de extrema importância que encontra diversas aplicações no campo do processamento de sinais. Geralmente, os métodos de predição envolvem o relacionamento linear entre a amostra predita e as amostras anteriores de uma sequência. Nem sempre, no entanto, esse relacionamento é linear: diversos sinais reais, como, por exemplo, a fala humana, exigem modelos não-lineares para que a predição seja feita de forma eficiente. Este artigo mostra uma nova forma de realizar a predição não-linear de amostras de um sinal discreto. O método é baseado no algoritmo ISAT, originalmente desenvolvido para solucionar equações diferenciais em processos químicos. O artigo apresenta resultados de simulações e comparações com métodos já conhecidos de predição.

Palavras-Chave— Algoritmo ISAT, Predição não linear, Processamento de sinais.

Abstract— Prediction of time series is an activity of extreme importance that finds numerous uses in signal processing. In general, prediction techniques establish a linear relationship among the predicted sample and past samples in a discrete signal. However, this is not always the case: many types of real-world signals, such as speech, are more efficiently modelled with a nonlinear relation. In this paper, we present a new method for nonlinear prediction of samples of a discrete signal. The method is based on the ISAT algorithm, which was originally developed to solve differential equations of chemical reactions. The paper shows the results of simulations, and compare them with already known methods.

Keywords— ISAT algorithm, Nonlinear prediction, Signal processing.

I. INTRODUÇÃO

Um sinal discreto pode ser encarado como a realização de um processo naturalmente discreto ou proveniente da amostragem de um sinal contínuo [1][2]. Em qualquer situação, as amostras do sinal são uma sequência de medidas ao longo do tempo e descrevem a variação de alguma grandeza. As aplicações de sinais discretos são inúmeras, e em qualquer uma delas, a predição das amostras futuras da sequência pode ser uma informação útil, desde que possa ser feita dentro de uma margem de erro aceitável. Entre as aplicações da predição de amostras, incluem-se a predição de sinais de voz [3], sinais caóticos [4], sinais de radar [5], cancelamento de eco [6], entre outros.

O problema da predição pode ser colocado de forma simples. Seja $x[n]$ um sinal discreto, em que n é o índice temporal. A predição consiste em encontrar uma estimativa adequada para a amostra futura $x[n+1]$ dadas N amostras anteriores

do sinal, ou seja

$$x[n+1] = f(x[n], x[n-1], \dots, x[n-N+1]), \quad (1)$$

em que N é a ordem da predição. Tipicamente, a predição é feita de forma linear, ou seja, a amostra predita é obtida através de uma combinação linear das amostras conhecidas. Um conhecido e bem estudado método de predição linear é realizado pela adequação dos coeficientes de um filtro de topologia conhecida (por exemplo, um filtro FIR) através do algoritmo LMS [7].

No entanto, não são incomuns os sinais em que o relacionamento não pode ser colocado facilmente na forma linear. Os artigos citados anteriormente fornecem exemplos de sinais em que isso acontece. Esses trabalhos desenvolvem técnicas para a predição não-linear de amostras. A técnica clássica utilizada na tarefa é através dos conhecidos filtros de Kalman [7], que fornecem meios para o cálculo de estimativas mesmo na presença de diversos tipos de ruído. No entanto, esses filtros podem ser sensíveis às estimativas iniciais das matrizes de correlação envolvidas, e podem fornecer resultados não ótimos ou até mesmo divergir nesses casos.

Entre as técnicas não-clássicas, as mais utilizadas são as redes neurais [3] e redes de funções de base radial [4], ambas com modificações para tornar a predição mais precisa. Particularmente, em [3], os autores sugerem uma rede neural realimentada com bom desempenho. Redes neurais, no entanto, têm o conhecido problema do projeto da arquitetura [8][9]: é necessário determinar a quantidade de neurônios nas camadas escondidas e um número de outros parâmetros como taxa de aprendizado e fator de esquecimento. A escolha inadequada desses parâmetros pode fazer com que a rede não atinja um comportamento adequado. Além disso, sistemas realimentados correm sempre o risco de divergência.

Uma técnica alternativa que não exige a criação de modelos paramétricos é a tabulação das relações entre as variáveis independentes e dependentes. Esse tipo de estratégia recebe a designação “armazenar e recuperar” (*store and retrieve*). O ISAT (*In-Situ Adaptive Tabulation*) [10][11] é um algoritmo dessa natureza. Criado para aumentar a eficiência da solução de equações diferenciais em reações químicas, é na verdade um método adaptativo para a tabulação de valores de funções. Isso significa que pode ser usado para estabelecer o mapeamento descrito na Eq. (1). Basicamente, consiste em um algoritmo que realiza aproximações lineares locais em um trecho do domínio de definição da função, associadas a um método para encontrar eficientemente o melhor mapeamento. Adicionalmente, caso a aproximação esteja além de uma margem de erro arbitrária, o algoritmo permite o crescimento

da tabulação, de modo a manter a aproximação em níveis adequados.

Como não é um método paramétrico, o algoritmo ISAT encontra algumas vantagens em relação aos métodos baseados, por exemplo, em redes neurais. Entre essas vantagens, citamos a convergência garantida do mapeamento; e a flexibilidade em mapear desde relacionamentos lineares até outros com alto grau de complexibilidade. No entanto, como qualquer método de tabulação, o ISAT pode ser moderadamente exigente em seus requisitos de memória, o que pode fazer com que sua aplicação em sistemas em que esse recurso seja limitado seja infactível.

Este artigo descreve o algoritmo ISAT e sua utilização para a predição não-linear de amostras de uma sequência. Testes foram feitos com sinais de voz, mas a técnica é genérica o suficiente para que seja aplicada em outros processos em que o mapeamento do relacionamento entre variáveis seja necessário. Este artigo se organiza da seguinte forma: a seção II descreve o algoritmo ISAT e como utilizá-lo para a predição de amostras; a seção III descreve os experimentos realizados e seus resultados, comparando-os com as técnicas mais usuais de predição não-linear; a seção IV estabelece as conclusões e o direcionamento para pesquisas futuras.

II. O ALGORITMO ISAT

O algoritmo ISAT (*In Situ Adaptive Tabulation*) é um método relativamente recente no conjunto de técnicas capazes de aprender o mapeamento entre diversas variáveis. Foi descrito pela primeira vez em [10], como uma técnica para aumentar a eficiência da solução de equações diferenciais associadas à descrição de processos químicos. Dada a natureza das equações solucionadas, métodos de aproximação podem apresentar ganho de até 3 ordens de magnitude [10], uma vez que as equações diferenciais não precisam ser solucionadas, apenas aproximadas. Pequenos melhoramentos de desempenho foram apresentados em [11].

Algoritmos de tabulação em geral, e o algoritmo ISAT em particular, realizam uma troca comum quando a eficiência computacional está em jogo: permitem o aumento do consumo de memória para que a quantidade de cálculos diminua. Esse tipo de técnica constrói uma tabela ou estrutura de dados equivalente que permite a consulta rápida a valores já calculados. Se aproximações são permitidas, dentro de uma margem de erro, a tabulação então associa uma aproximação adequada a um intervalo na qual é válida. Se uma aproximação não for possível, então o método deve permitir o crescimento da tabela. Um bom algoritmo dessa natureza permite que isso seja feito de maneira rápida dentro de uma margem de erro arbitrariamente estabelecida.

O ISAT divide o domínio de definição da função a ser mapeada em diversos subdomínios, nos quais a aproximação é feita de maneira linear. Seja $\mathbf{f}(\mathbf{x})$ a função sob investigação, e $\hat{\mathbf{f}}(\mathbf{x})$ sua estimativa sobre um vetor \mathbf{x} , sendo

$$\mathbf{x} = [x_1 \quad x_2 \quad \dots \quad x_N]^t \quad (2)$$

o vetor das variáveis independentes, e \mathbf{x}^t é a transposta da matriz \mathbf{x} . A função $\mathbf{f}(\mathbf{x})$ é definida de tal forma que um

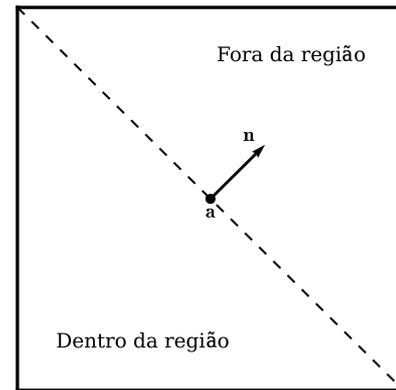


Fig. 1. Determinação da região em que o vetor inspecionado se encontra, em duas dimensões.

relacionamento de muitos-para-muitos seja possível, ou seja

$$\mathbf{f}(\mathbf{x}) = [f_1(\mathbf{x}) \quad f_2(\mathbf{x}) \quad \dots \quad f_M(\mathbf{x})]^t. \quad (3)$$

Apesar da aproximação em cada região ser realizada de forma linear, a divisão do domínio em diversas regiões com aproximações diferentes dá o caráter não-linear do método.

Para determinar em que região da tabulação o vetor inspecionado se encontra, o algoritmo as organiza em uma árvore binária [12], na qual os nós são decisões sobre a posição do vetor no domínio mapeado, e as folhas contêm as aproximações. A aplicação de uma árvore binária é interessante pois reduz significativamente a complexidade da busca. Se L é a profundidade da árvore, então a busca será terminada, no pior caso, após $\log_2 L$ operações. Embora esse índice não seja significativo para pequenas partições, torna-se significativo quando a árvore atinge grande magnitude. Em um nó, uma região é delimitada por um hiperplano definido por um vetor unitário ortogonal \mathbf{n} e um ponto \mathbf{a} a ele pertencente. O ponto investigado estará dentro da região definida pelo hiperplano se [14]:

$$(\mathbf{x} - \mathbf{a}) \cdot \mathbf{n} < 0, \quad (4)$$

em que $\mathbf{x} \cdot \mathbf{n}$ é o produto escalar entre os vetores \mathbf{x} e \mathbf{n} . O algoritmo percorre a árvore a partir do primeiro nó até encontrar uma folha, que conterá uma aproximação linear. A figura 1 ilustra o conceito para duas dimensões

Uma aproximação linear para $\mathbf{f}(\mathbf{x})$ é dada por

$$\hat{\mathbf{f}}(\mathbf{x}) = \mathbf{f}(\mathbf{x}_i) + \mathbf{A}(\mathbf{x} - \mathbf{x}_i), \quad (5)$$

em que \mathbf{x}_i é o centro da região do domínio em que se encontra o vetor inspecionado. É direto reconhecer que a Eq. (5) é a série de Taylor até a primeira ordem para uma função qualquer, se considerarmos a matriz \mathbf{A} como o gradiente da função $\mathbf{f}(\mathbf{x})$, ou seja cada elemento a_{ij} da matriz \mathbf{A} é dado por

$$a_{ij} = \frac{\partial f_i}{\partial x_j}. \quad (6)$$

As componentes da matriz \mathbf{A} podem ser encontradas diretamente por inspeção, quando a relação está disponível. No caso

da predição, no entanto, a relação não é conhecida, portanto, \mathbf{A} precisa ser estimada, o que pode ser feito através de um método adequado, usando os próprios dados disponíveis da sequência [13].

A qualidade da aproximação é medida pelo erro:

$$\epsilon = \|\mathbf{f}(\mathbf{x}) - \hat{\mathbf{f}}(\mathbf{x})\|, \quad (7)$$

em que $\|\mathbf{x}\|$ é a norma euclidiana do vetor \mathbf{x} .

A matriz de linearização tem propriedades que permitem determinar em que região essa aproximação é válida. Calculando o erro cometido na aproximação, substituindo-se a Eq. (5) na Eq. (7), e tomando-se o quadrado do resultado, obtemos o erro quadrático da aproximação, dado por

$$\epsilon^2 = \|\mathbf{f}(\mathbf{x}) - \mathbf{f}(\mathbf{x}_i) - \mathbf{A}(\mathbf{x} - \mathbf{x}_i)\|^2. \quad (8)$$

A região delimitada por essa equação é de difícil avaliação, mas é possível obter um limite conservador adequado, se considerarmos que a aproximação é constante, ou seja, $\hat{\mathbf{f}}(\mathbf{x}) = \mathbf{f}(\mathbf{x}_i)$ na i -ésima região. Essa hipótese é adequada porque a aproximação por uma constante acarretará em erros de maior magnitude que a aproximação linear. Portanto, se um ponto é encontrado dentro da região de precisão indicada por uma aproximação constante, ele também estará na região de precisão de uma aproximação linear. Nesse caso, o erro cometido é dado por

$$\begin{aligned} \epsilon^2 &= \|\mathbf{A}(\mathbf{x} - \mathbf{x}_i)\|^2 \\ &= \Delta\mathbf{x}^t \mathbf{A}^t \mathbf{A} \Delta\mathbf{x}, \end{aligned} \quad (9)$$

em que $\Delta\mathbf{x} = \mathbf{x} - \mathbf{x}_i$. Se o erro máximo tolerado for dado por ϵ_t , então o que se deseja é que

$$\Delta\mathbf{x}^t \mathbf{A}^t \mathbf{A} \Delta\mathbf{x} / \epsilon_t^2 \leq 1. \quad (10)$$

A matriz $\mathbf{A}^t \mathbf{A}$ pode ser decomposta em seus autovalores e autovetores [14], resultando em

$$\mathbf{A}^t \mathbf{A} = \mathbf{Q}^t \mathbf{\Lambda} \mathbf{Q}, \quad (11)$$

em que \mathbf{Q} é a matriz cujas colunas são os autovetores e $\mathbf{\Lambda}$ é a matriz diagonal em que os elementos na diagonal principal são os autovalores λ_i da matriz $\mathbf{A}^t \mathbf{A}$. Substituindo (11) em (10), obtemos

$$\Delta\mathbf{x}^t \frac{\mathbf{Q}^t \mathbf{\Lambda} \mathbf{Q}}{\epsilon_t^2} \Delta\mathbf{x} \leq 1. \quad (12)$$

Essa equação representa um hiperelipsoide no domínio de definição das variáveis independentes. A Eq. (12) representa a região do domínio em que a aproximação é válida. Essa região é chamada elipsoide de precisão, e é ilustrada em duas dimensões na figura 2.

Tendo disponíveis os métodos para a aproximação em uma determinada região e um critério para checar a adequabilidade dessa aproximação, o algoritmo ISAT pode ser resumido na sequência de passos abaixo:

- 1) Inicializa-se a árvore binária contendo apenas uma aproximação, dada por um ponto inspecionado arbitrariamente. Ou seja, a árvore inicial consiste em apenas uma folha. Para essa folha armazena-se o centro \mathbf{x}_1 , a matriz gradiente \mathbf{A} correspondente e a decomposição em autovalores e autovetores da Eq. (11).

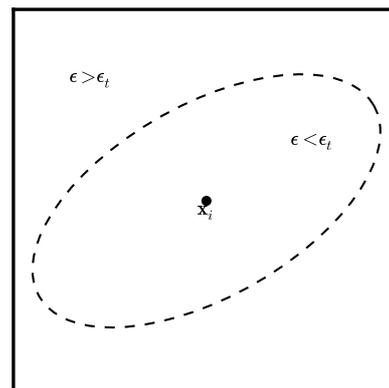


Fig. 2. Elipsoide de precisão, dentro do qual se garante a precisão da aproximação linear.

- 2) Dado um vetor \mathbf{x} e uma resposta desejada para esse vetor $\mathbf{f}(\mathbf{x})$, determina-se a qual região o vetor pertence. A decisão é feita seguindo o ramo que indica que o vetor está dentro da região caso a Eq. (4) seja satisfeita, e o ramo alternativo caso contrário, até que uma folha da árvore binária seja encontrada.
- 3) Caso o vetor esteja dentro do elipsoide de precisão da região encontrada, retorna-se a aproximação. Caso contrário, verifica-se se a aproximação satisfaz a precisão desejada. Em caso positivo, o elipsoide de precisão da região é atualizado. Isso é feito manipulando-se os autovetores em \mathbf{Q} e autovalores em $\mathbf{\Lambda}$ para que o elipsoide inclua o ponto inspecionado. A figura 3 ilustra o processo. O autovetor associado ao menor semi-eixo deve ter sua direção modificada, e o autovalor correspondente deve ter sua magnitude alterada. Note-se que a matriz gradiente \mathbf{A} não deve ser modificada.
- 4) Caso a precisão não seja atendida, a região é particionada para que uma folha contendo uma nova aproximação seja adicionada à árvore. Isso é feito determinando-se o hiperplano ortogonal ao vetor que une o centro da região ao ponto inspecionado, passando pelo ponto que divide a distância em duas partes iguais. A folha criada deve conter os dados necessários para que as aproximações sejam calculadas, ou seja, o centro \mathbf{x}_i , a matriz gradiente \mathbf{A} e sua decomposição em autovalores e autovetores.

O uso do algoritmo ISAT na predição de amostras se faz com poucas adaptações. A função a ser estimada é aquela da Eq. (1), portanto,

$$\mathbf{x} = [x[n] \quad x[n-1] \quad \dots \quad x[n-N+1]]^t \quad (13)$$

e

$$\mathbf{f}(\mathbf{x}) = x[n+1]. \quad (14)$$

É importante notar que o algoritmo é executado *on-line*, ou seja, as estruturas de dados são atualizadas conforme os resultados se fazem necessários. Quando a aproximação não é adequada o suficiente, o algoritmo apenas retorna o

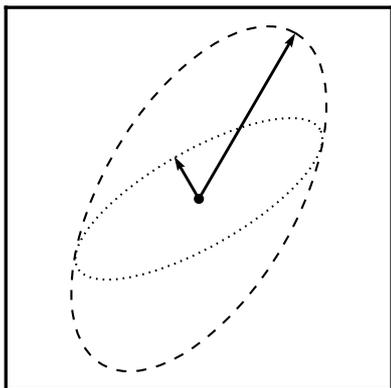


Fig. 3. Crescimento do elipsoide de precisão, incluindo um novo ponto.

valor desejado. Isso faz com que as primeiras aproximações contenham poucos erros, como se verificará na próxima seção. Em um problema de predição, no entanto, o valor desejado da função não está, por definição, disponível quando o algoritmo é executado. Portanto, uma fase de treinamento é necessária para que as primeiras aproximações sejam calculadas. Embora isso represente um processamento adicional, esse é um passo comum em algoritmos que realizam aproximações.

III. EXPERIMENTOS E SIMULAÇÕES

Esta seção apresenta o resultado de simulações da predição não-linear usando o algoritmo ISAT. Comparamos os resultados com os obtidos pelo uso dos filtros de Kalman, e do algoritmo descrito em [3]. Os resultados são apresentados na forma de gráficos mostrando o resultado da predição e o erro cometido. O erro quadrático médio e , no caso do algoritmo ISAT, a ocupação em memória, são registrados como medidas do desempenho. Em todos os resultados de predição, o sinal original é mostrado em linha tracejada, e o sinal predito em linha cheia.

Realizou-se a predição de um sinal de voz de um locutor masculino pronunciando a sílaba “ni”, amostrado a 8 kHz. O sinal original ocupa 0,3 s, totalizando 2400 amostras. Para tornar os gráficos mais legíveis, são mostradas aqui apenas as primeiras 500 amostras do resultado final. As amostras foram normalizadas para situarem-se no intervalo entre $-0,5$ e $0,5$. Como a codificação foi feita em 8 bits, é possível determinar o tamanho do domínio. Se N é a ordem da predição, e cada variável independente pode assumir 256 valores, então o domínio será composto de 256^N valores.

A figura 4 mostra o resultado da aplicação dos algoritmos. Em (a), o gráfico representa as 500 primeiras amostras do sinal original; em (b), as respectivas amostras preditas pelo filtro de Kalman; em (c), as amostras preditas pelo método descrito em [3]; em (d), o resultado obtido pelo algoritmo ISAT com ordem de predição $N = 4$ e margem de erro $\epsilon_t = 0,1$; e, por fim, em (e), o algoritmo ISAT com ordem de predição $N = 4$ e margem de erro $\epsilon_t = 0,01$. Em todos os gráficos, o eixo horizontal representa o tempo discreto.

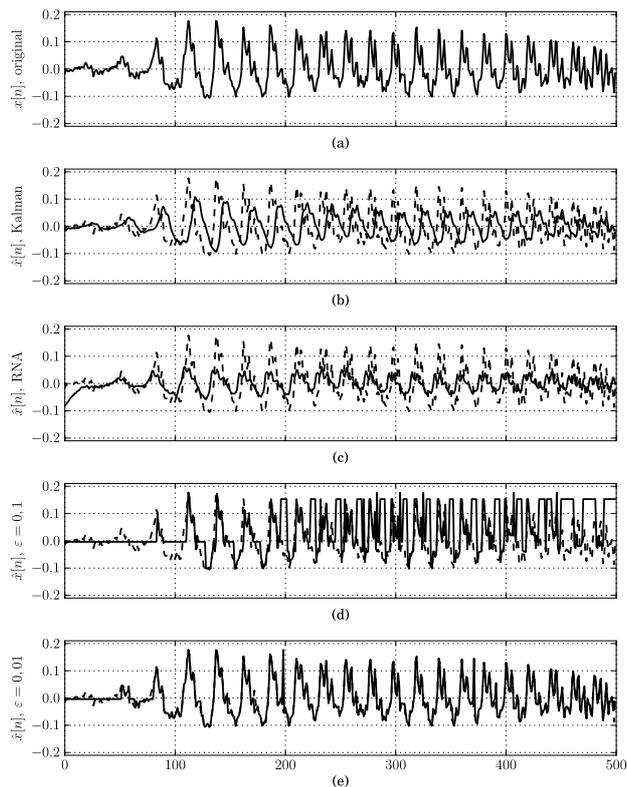


Fig. 4. Resultado da aplicação do algoritmo de predição em um sinal de voz. Em (a), o sinal original; em (b) o resultado com a aplicação de filtro de Kalman; em (c) o resultado da aplicação de redes neurais; em (d) o algoritmo ISAT com $\epsilon_t = 0,1$ e em (e) o ISAT com $\epsilon_t = 0,01$.

A figura 5 mostra os sinais de erro obtidos. O sinal original não é mostrado. O gráfico em (a) representa o erro cometido pelo filtro de Kalman; em (b), o erro cometido com o uso das redes neurais segundo o algoritmo descrito em [3]; em (c) o erro pelo algoritmo ISAT com ordem de predição $N = 4$ e margem de erro $\epsilon_t = 0,1$; e, por fim, em (d), o algoritmo ISAT com ordem de predição $N = 4$ e margem de erro $\epsilon_t = 0,01$. Em todos os gráficos, o eixo horizontal representa o tempo discreto.

A figura 4(b) mostra o resultado obtido pelo filtro de Kalman, e a figura 5(a) o erro cometido. O erro quadrático médio obtido através desse método de predição foi de cerca de $-22,9$ dB. A figura 4(c) mostra o resultado da predição segundo o algoritmo descrito em [3]. Para essa simulação, os parâmetros utilizados foram conforme sugeridos no mesmo artigo. O erro cometido na predição é mostrado na figura 5(b). O erro quadrático médio obtido através desse método de predição foi de cerca de $-23,7$ dB.

O gráfico na figura 4(d) mostra o resultado do processamento do sinal utilizando o algoritmo ISAT, com ordem de predição $N = 4$ e margem de erro $\epsilon_t = 0,1$. É possível notar que a predição não é especialmente eficiente quando o treinamento é completado, ou seja, após as primeiras amostras serem apresentadas. A análise desse resultado revela que isso acontece porque a margem de erro é excessivamente alta,

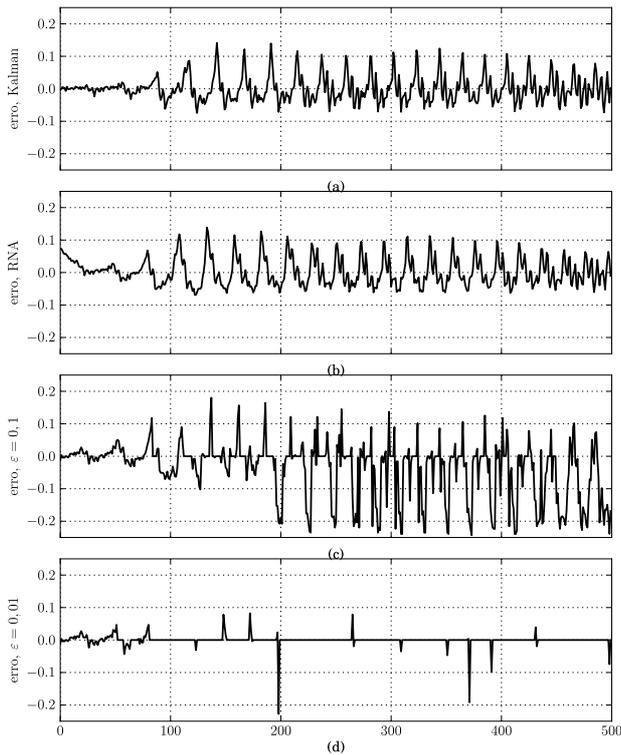


Fig. 5. Erros cometidos nas previsões de um sinal de voz. Em (a), o erro cometido pelo filtro de Kalman; em (b), redes neurais; em (c), o ISAT com $\epsilon_t = 0, 1$; e em (d), o ISAT com $\epsilon_t = 0, 01$.

20% da amplitude máxima, o que faz com que aproximações inadequadas sejam utilizadas. A figura 5(c) mostra o erro cometido a cada amostra. O erro quadrático médio ficou em torno de -21,9 dB. A árvore resultante apresentou um total de 2143 folhas e nós. Ainda que o tamanho final da árvore pareça excessivo para uma sequência de apenas 2400 amostras, é preciso lembrar que esse número inclui tanto nós como folhas, não representando 2143 regiões de aproximação, mas um total de aproximações e hiperplanos de decisão.

Para comprovar que o algoritmo pode alcançar a meta adequada de erro, repetimos o experimento utilizando agora a margem de erro $\epsilon_t = 0, 01$, o que corresponde a 2% da amplitude máxima. A figura 4(e) mostra o resultado da previsão. O sinal original fica praticamente obscurecido pelo sinal predito, ocorrendo picos de erro em algumas amostras. Esses picos acontecem porque, nesses pontos, novas aproximações precisaram ser criadas. Ainda que o algoritmo pudesse retornar à amostra calculada para emitir uma estimativa mais adequada, isso não é a prática comum, pois pode comprometer a eficiência da execução do algoritmo (impedindo, por exemplo, sua aplicação em sistemas de tempo real). Além disso, considera-se que, a partir de certo ponto, o aprendizado já está concluído, e o algoritmo não tem a capacidade de adicionar novos resultados. A figura 5(d) mostra mais claramente esses pontos. O erro quadrático médio obtido na previsão foi de cerca de -31,6 dB. A árvore resultante apresentou um total de 4021 folhas e nós.

Comparativamente aos algoritmos existentes, pode-se concluir que é possível obter resultado consistentemente me-

lhor desde que adotada uma margem de erro baixa o suficiente. Como o algoritmo ISAT permite a especificação desse parâmetro, é fácil escolher o desempenho necessário. Algoritmos baseados em redes neurais, por sua vez, exigem a manipulação do modelo, e não existe uma relação direta entre o desempenho obtido e a arquitetura da rede utilizada.

Note-se, no entanto, que o ganho em precisão aparece ao custo de maior ocupação em memória. Ainda que a execução do algoritmo seja rápida, requisitos de memória podem tornar proibitiva a implementação quando esse recurso é limitado. Apesar disso, a velocidade de execução é alta, pois as operações são poucas e rápidas. Portanto, ISAT se mostra uma alternativa para situações em que o mapeamento deve ser realizado rapidamente e com alto grau de precisão.

IV. CONCLUSÕES

Este artigo descreveu o algoritmo ISAT (*In Situ Adaptive Tabulation*) e o seu uso na previsão não-linear de amostras. Esse algoritmo fornece um método para o rápido armazenamento, recuperação e aproximação de funções de um número arbitrário de variáveis dependentes. A aproximação é linear e realizada localmente, obedecendo uma margem de erro arbitrária.

A aplicação do algoritmo à tarefa da previsão de sinais não-lineares, atividade comum em diversos campos do processamento de sinais, foi testada contra um sinal de voz. Notou-se que a previsão obteve bons resultados, com erro quadrático médio de baixo valor. Na comparação com outro algoritmo tradicionalmente utilizado para a mesma tarefa, o ISAT mostrou resultados superiores.

Apesar da aplicação restrita desse artigo, o ISAT é um algoritmo genérico de mapeamento. Pode ser aplicado em qualquer tarefa que exija o aprendizado de um relacionamento qualquer entre diversas variáveis, seja esse relacionamento linear ou não. Investigações futuras das aplicações do algoritmo em telecomunicações em geral, e processamento de sinais em específico estão planejadas.

REFERÊNCIAS

- [1] J. A. Nalon, *Introdução ao Processamento Digital de Sinais*. LTC Editora, 2009.
- [2] A. Oppenheim & R. W. Schaffer, *Discrete-time Signal Processing*, Prentice-Hall, 1989.
- [3] S. Haykin & L. Li, *Nonlinear Adaptive Prediction of Nonstationary Signals*, IEEE Trans. on Signal Processing, vol. 43, No. 2, pp. 526-535, Feb. 1995.
- [4] M. R. Cowper, B. Mulgrew & C. P. Unsworth, *Nonlinear prediction of chaotic signals using a normalised radial basis function network*, Signal Processing, 82, pp. 775-789, 2002.
- [5] N. J. Gordon, D. J. Salmond & A. F. M. Smith, *Novel Approach to Nonlinear/Non-Gaussian Bayesian Estimation*, IEE Proceedings-F, vol. 140, No. 2, pp. 107-113, 1993.
- [6] T. van Waterschoot, G. Rombouts, P. Verhoeve & M. Moonen, *Double-Talk-Robust Prediction Error Identification Algorithms For Acoustic Echo Cancellation1*, IEEE Trans. on Signal Processing, vol. 55, No. 3, pp. 846-858, 2007.
- [7] S. Haykin, *Adaptive Filter Theory*, 4th ed., Prentice-Hall, 2002.
- [8] S. Haykin, *Neural Networks: A Comprehensive Foundation*, 2nd ed., Prentice-Hall, 1999.
- [9] R. Eberhart & Y. Shi, *Computational Intelligence: Concepts to Implementations*, Morgan Kaufmann, 2007.
- [10] S. B. Pope, *Computationally Efficient Implementation of Combustion Chemistry Using In Situ Adaptive Tabulation*, Combust. Theory Modelling, vol. 1, pp. 41-63, 1997.

- [11] L. Lu & S. B. Pope, *An Improved Algorithm for In Situ Adaptive Tabulation*, Journal of Computational Physics, 228, pp. 361-386, 2009.
- [12] A. V. Aho, J. E. Hopcroft & J. D. Ullman, *Data Structures and Algorithms*, Addison-Wesley, 1987.
- [13] R. W. Hamming, *Numerical Methods for Scientists and Engineers*, 2nd ed., Dover Publications, 1986.
- [14] H. Anton & C. Rorres, *Álgebra Linear com Aplicações*, Ed. Bookman, 2001.