

Uma Técnica para Ocultação de Partituras Musicais em Sinais de Áudio

Mário C. L. da Cruz Jr. e Juliano B. Lima

Resumo— Neste trabalho, descreve-se uma técnica para ocultar partituras musicais em sinais de áudio digital. O objetivo é fornecer ao usuário que obtém o arquivo de áudio de uma música a respectiva partitura, a qual pode ser extraída por meio de um processamento sistemático. O método emprega partituras escritas segundo a notação ABC, que codifica em formato de texto os diversos símbolos musicais. Os caracteres, convertidos em bits, são inseridos no sinal de áudio segundo um procedimento de ajuste de fase no domínio da transformada discreta de Fourier. Experimentos que permitem avaliar a robustez do método à compressão mp3 são realizados e estratégias para melhoria de desempenho são discutidas.

Palavras-Chave— Ocultação de dados, transformada discreta de Fourier, partituras musicais.

Abstract— In this paper, a technique for hiding musical scores in digital audio signals is described. The purpose is to provide for a user accessing the audio file of a music the respective musical score, which can be extracted by means of a systematic processing. The method employs musical scores written according to the ABC notation, which encodes the musical symbols in text format. The characters are converted into bits and embedded in the audio signal using a phase adjustment procedure in the discrete Fourier transform domain. Experiments which allow to evaluate the robustness of the method against mp3 compression are carried out and strategies for enhancing its performance are discussed.

Keywords— Data hiding, discrete Fourier transform, music scores.

I. INTRODUÇÃO

Com o crescimento e a popularização das redes de comunicação banda larga, o compartilhamento de informação multimídia passou a ser uma atividade realizada de modo cada vez mais corriqueiro por usuários de todo o mundo [1]. Atualmente, por meio de uma conexão doméstica, é possível enviar e acessar áudio e vídeo de alta qualidade, por meio de serviços pagos ou de portais que disponibilizam gratuitamente este tipo de conteúdo.

Diante de tal expansão, surgiu a necessidade de desenvolver técnicas voltadas à segurança do conteúdo mencionado. Nesse contexto, merecem destaque a esteganografia e a marca d'água digital [2], [3]. A primeira tem o objetivo de embutir numa imagem, num sinal de áudio ou de vídeo, uma informação que se deseja transmitir secretamente. A última tem como propósito a introdução *invisível* de uma informação que permite identificar a proveniência e/ou a integridade da respectiva mídia.

Técnicas para ocultação de dados também podem ser empregadas com a finalidade de prover informações adicionais,

não necessariamente sigilosas, a respeito do sinal em que elas são embutidas [4]. É esse o contexto em que se encontra o presente trabalho, em que se apresenta uma técnica para ocultação de partituras musicais em sinais de áudio digital. O método, que é inspirado na proposta introduzida em [5], tem como objetivo permitir que um usuário, ao obter o arquivo de áudio de uma música, tenha também acesso imediato à partitura dessa música. Com isso, elimina-se a necessidade de que o usuário interessado em executar a música busque a sua partitura noutras fontes, tarefa que demanda tempo e que pode retornar resultados não-confiáveis.

O método proposto faz uso da notação ABC, que permite escrever partituras musicais sem o emprego de *softwares* específicos e que geralmente contêm uma interface gráfica elaborada [6]. Na notação ABC, caracteres ASCII são usados para representar notas e outros símbolos, e uma estrutura puramente textual é gerada. Embora a utilização de ferramentas gráficas para a escrita musical possa ser considerada mais intuitiva e de entendimento mais fácil por parte de músicos iniciantes, a utilização de uma notação textual permite rapidez e praticidade, uma vez que qualquer aplicativo de bloco de notas é suficiente para iniciar a escrita.

De qualquer modo, também existem *softwares* específicos para a escrita em notação ABC. Em relação aos demais *softwares* de composição (notação gráfica), esses *softwares* possuem, dentre outras vantagens, as seguintes:

- Praticamente todos os programas para notação ABC são gratuitos;
- Os programas, em geral, são compactos e possíveis de se utilizar em computadores de baixo rendimento ou mesmo *smartphones*;
- A qualidade das folhas de música que se obtém usando programas para notação ABC são, algumas vezes, melhores do que as obtidas nos programas tradicionais;
- As partituras são criadas em arquivos PostScript ou pdf, e não em formatos reservados aos programas tradicionais, os quais dificultam o compartilhamento.

O primeiro passo da técnica proposta neste trabalho é transformar em bits os caracteres de um texto que representa a partitura de uma música escrita segundo a notação ABC. Cada bit é inserido, por meio de uma regra de ajuste de fase [5], [7], [8], em componentes específicas da transformada discreta de Fourier (DFT) de segmentos do respectivo sinal de áudio. O procedimento de extração dos bits consiste, basicamente, em implementar em ordem reversa os passos executados na inserção.

Após esta introdução, na Seção II, a técnica proposta é descrita em detalhes; são apresentadas versões para uso em sinais de áudio mono e estéreo. Na Seção III, a aplicação do

método é ilustrada por meio de experimentos, cujos resultados são exibidos e analisados; são discutidas questões relacionadas à robustez do método à compressão mp3 e sugeridas estratégias para correção de erros e melhoria do seu desempenho. As considerações finais deste trabalho são apresentadas na Seção IV.

II. TÉCNICA PROPOSTA

Nesta seção, são descritos os detalhes da técnica proposta. Inicialmente, é apresentada uma versão aplicável a sinais de áudio mono e, em seguida, propõe-se uma modificação que permite aplicar o método a sinais de áudio estéreo. Neste último caso, que contempla o formato em que se encontra a maioria dos arquivos de áudio digital utilizados comercialmente e para entretenimento, os dois canais são processados de forma conjunta.

A. Descrição da Técnica

O áudio mono apresenta canal audível único e, assim, todo o material sonoro é registrado e/ou transmitido de forma consolidada. Um sinal de áudio digital mono, salvo num arquivo em formato *wave*, por exemplo, pode ser tratado como um vetor

$$\mathbf{A} = \{A(i) | 1 \leq i \leq L\}$$

com L pontos.

A partitura escrita na notação ABC que se deseja inserir em \mathbf{A} é obtida de um arquivo de texto. Como exemplo, é fornecida a seguir a partitura da música “Here, there and everywhere” na notação ABC:

```

1 X:01@
2 T: Here, There and Everywhere@
3 T: From Album 'Revolver' 1966 UK@
4 O: Lennon/McCartney vocal by Paul
5 McCartney/John Lennon and George Harrison@
6 M:4/4@
7 L:1/4@
8 Q:1/4=90@
9 K:G@
10 %@
11 G4 |: B d/2a/2 g c/2e/2 | d4 | B c/2d/2 e G/2A/2
12 | B B/2B/2 B2 | B c/2B/2 B A | G A/2B/2 ^f e |@
13 [1 A4 :|2 A2 z/2A/2 _B/2c/2 | d/2f/2 d2 c/2_B/2 |
14 c/d/_e/c/ A/A/_B/c | d/_B/ G2> | c/d/_e/c/A/A/_
15 B/c | d/_B/ G2> ||@A2 z/D/ G/G/ | G2> B | d B/A/
16 G2 | G2 G> B/ | d e ^f2 | B4]
    
```

Os caracteres “@” não fazem parte da notação ABC propriamente dita. Eles são inseridos para funcionar como um delimitador, indicando pontos em que se deve criar uma nova linha, durante a extração dos bits e a geração do arquivo .txt recuperado; os números no início de cada linha também não fazem parte do texto original. A partitura da mesma música, usando os símbolos musicais convencionais, é apresentada na Figura 1.

Todos os caracteres da partitura são convertidos para binário segundo a tabela ASCII, produzindo um vetor

$$\mathbf{w} = \{w(i) | 1 \leq i \leq M\}$$

com M pontos e componentes de valor 0 ou 1. O processo de inserção é realizado mediante a execução dos seguintes passos [5]:



Fig. 1: Partitura de música gerada a partir de um arquivo de texto escrito usando a notação ABC.

Passo 1. Segmentação do sinal de áudio original: realiza-se a segmentação do vetor \mathbf{A} em pelo menos M segmentos. Pode-se segmentar todo o vetor ou apenas parte dele; tal escolha depende da relação entre os comprimentos L e M . A seguir, considera-se que M divide L e que todo o vetor \mathbf{A} é segmentado. Os segmentos obtidos são denotados por

$$\mathbf{A}_s = \{A_s(h, l) | 1 \leq h \leq M, 1 \leq l \leq L/M\};$$

Passo 2. Cálculo da DFT: calcula-se a transformada de Fourier de cada segmento de \mathbf{A}_s . O resultado obtido é denotado por

$$\mathbf{A}'_s = \{A'_s(h, l)\};$$

Passo 3. Seleção de coeficientes de frequências médias: em cada segmento de \mathbf{A}'_s , são selecionadas C componentes consecutivas na faixa de frequências médias. O objetivo é obter uma boa imperceptibilidade, isto é, o som da música após a inserção dos bits deve ser o mais parecido possível com o da música original. Os novos segmentos (C componentes de cada segmento de \mathbf{A}'_s) são denotados por \mathbf{A}_c ;

Passo 4. Inserção da marca d'água: em cada segmento de \mathbf{A}_c , é escolhida a componente com maior amplitude para inserção de cada bit de \mathbf{w} . Tais componentes são selecionadas a fim de proporcionar robustez ao método, contra manipulações como a compressão mp3. Uma análise mais formal acerca desse aspecto pode ser desenvolvida considerando que a componente com amplitude ρ_0 e fase φ utilizada para inserção do bit da partitura, após alguma manipulação, tem sua amplitude modificada para ρ e sua fase incrementada de δ . Com base na Figura 2, pode-se escrever

$$\rho \cos(\delta) - \rho_0 = \Delta_1, \quad (1)$$

$$\rho \sin(\delta) = \Delta_2, \quad (2)$$

em que Δ_1 e Δ_2 são, supostamente, dois ruídos unidimensionais com alguma distribuição. A partir das equações acima, tem-se

$$\tan(\delta) = \frac{\Delta_2}{\Delta_1 + \rho_0}, \quad (3)$$

$$\rho^2 = (\rho_0 + \Delta_1)^2 + \Delta_2^2. \quad (4)$$

Com base nas duas últimas equações, é possível concluir que um aumento em ρ_0 ocasiona uma elevação em ρ e um decréscimo em $|\delta|$. Dessa forma, a fase de um elemento com a maior amplitude será menos sensível a algum tipo de manipulação.

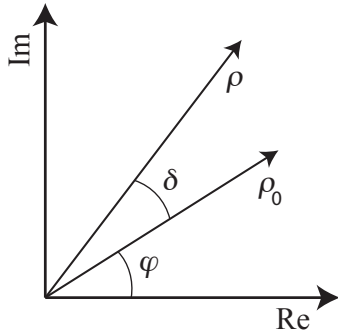


Fig. 2: Ilustração relacionada às modificações sofridas por uma componente do vetor, em função da inserção de um bit via modulação de fase.

A inserção propriamente dita dos bits é feita conforme uma técnica de modulação de fase [8]. Suponha que, em dado segmento de \mathbf{A}_c , a componente de maior amplitude A_m possua fase denotada por α , $0 \leq \alpha < 2\pi$. A inserção do bit w de \mathbf{w} correspondente a esse segmento modifica A_m para

$$A'_m = A_m + \gamma,$$

em que γ é um número real positivo utilizado para distinguir a componente que recebeu o bit das demais componentes, e modifica α para

$$\alpha' = \begin{cases} (temp + 0.5) \times step, & \text{se } w = 0 \text{ e } \text{mod}(temp, 2) = 0 \\ (temp - 0.5) \times step, & \text{se } w = 0 \text{ e } \text{mod}(temp, 2) = 1 \\ & \text{e } \alpha < (temp + 0.5) \times step \\ (temp + 1.5) \times step, & \text{se } w = 0 \text{ e } \text{mod}(temp, 2) = 1 \\ & \text{e } \alpha \geq (temp + 0.5) \times step \\ (temp + 0.5) \times step, & \text{se } w = 1 \text{ e } \text{mod}(temp, 2) = 1 \\ (temp - 0.5) \times step, & \text{se } w = 1 \text{ e } \text{mod}(temp, 2) = 0 \\ & \text{e } \alpha < (temp + 0.5) \times step \\ (temp + 1.5) \times step, & \text{se } w = 1 \text{ e } \text{mod}(temp, 2) = 0 \\ & \text{e } \alpha \geq (temp + 0.5) \times step; \end{cases}$$

o parâmetro $step$ representa um passo de quantização ímpar-par e $temp = \lfloor \alpha/step \rfloor$ [7]. Um aumento no valor do $step$ proporciona também um aumento na robustez do esquema de inserção. Entretanto, como tal incremento é acompanhado de distorções no sinal de áudio processado, é importante selecionar o valor do $step$ de forma equilibrada. Esse aspecto é melhor ilustrado na Seção III.

Passo 5. Cálculo da DFT inversa: esta etapa consiste na aplicação aos segmentos do sinal de áudio, após a inserção dos bits, do procedimento inverso ao realizado no Passo 2.

Passo 6. Reorganização dos segmentos: neste último passo do processo de inserção dos bits, os segmentos são reorganizados, a fim de que o vetor correspondente ao sinal de áudio seja remontado e o sinal de áudio completo, com os bits da partitura inseridos, seja obtido.

O processo de extração dos bits para obtenção da partitura é praticamente o inverso do processo de inserção. Num passo inicial, é realizada uma segmentação do sinal de áudio. Após isso, é efetuado o cálculo da DFT de cada segmento e isoladas as componentes associadas às frequências médias. Em cada segmento resultante, seleciona-se a componente de maior

amplitude, a qual possui fase denotada por $\hat{\alpha}'$, $0 \leq \hat{\alpha}' < 2\pi$. O bit \hat{w} de \mathbf{w} , que, sendo extraído de forma correta, coincide com o bit w inserido em determinado segmento do áudio original, é recuperado segundo a regra:

$$\hat{w} = \begin{cases} 0, & \text{se } \text{mod} \left(\left\lfloor \frac{\hat{\alpha}'}{step} \right\rfloor, 2 \right) = 0, \\ 1, & \text{se } \text{mod} \left(\left\lfloor \frac{\hat{\alpha}'}{step} \right\rfloor, 2 \right) = 1. \end{cases}$$

Finalmente, os bits extraídos são organizados convenientemente e convertidos para caracteres ASCII, permitindo a reprodução de um arquivo no formato .txt que corresponde à partitura da música em questão.

Conforme indicado na seção introdutória deste trabalho, a geração de um arquivo no formato pdf ou em formato de imagem, apresentando a partitura da música escrita com os símbolos musicais convencionais, pode ser feita utilizando serviços gratuitos. É possível realizar essa conversão copiando o conteúdo do arquivo de texto resultante da extração dos bits e colando, por exemplo, na interface *web* oferecida em <http://www.mandolintab.net/abconverter.php>. Na mesma interface, o usuário também pode ouvir o resultado (em formato *midi*) da execução da partitura gerada.

B. Adaptação para Áudio Estéreo

A técnica descrita na seção anterior pode ser adaptada para inserção dos bits da partitura em sinais de áudio estéreo, os quais contêm dois canais. Os princípios de inserção e de extração dos bits permanecem os mesmos, porém, a representação do sinal de áudio estéreo por meio de um vetor é realizada conforme descrição a seguir. Como um sinal de áudio estéreo possui dois canais, o que se faz é criar um vetor

$$\mathbf{A} = \{A(i) = A_l(i) + j \cdot A_r(i) | 1 \leq i \leq L, j = \sqrt{-1}\}$$

com L componentes complexas, em que $A_l(i)$ e $A_r(i)$ são, respectivamente, a i -ésima componente do canal esquerdo e a i -ésima componente do canal direito do sinal de áudio correspondente. Daí, os passos 1 a 6 descritos para sinais de áudio do tipo mono podem ser aplicados, bem como as etapas envolvidas no processo de extração dos bits.

Além de os sinais de áudio estéreo serem mais difundidos entre os usuários, outro aspecto motivou a adaptação do método. Após a inserção dos bits de uma partitura num sinal de áudio mono, algumas componentes do vetor resultante adquirem uma parte imaginária residual, tornando-se números complexos; tal resíduo é proveniente do ajuste de fase implementado quando da inserção de cada bit da partitura. Dessa forma, a inscrição dessas componentes como amostras de um sinal de áudio de canal único, que deveriam ser puramente reais, fica comprometida.

A solução adotada para essa questão é, então, desprezar a referida parte imaginária residual e inscrever nas amostras do sinal de áudio produzido apenas a parte real das componentes do vetor. Embora este *truncamento* aparentemente não prejudique as propriedades sonoras do sinal, podem ocorrer erros na extração dos bits, mesmo sem que tenha sido feita qualquer manipulação do áudio; isso depende fundamentalmente dos parâmetros empregados no processo de inserção (γ e $step$). No

caso do áudio estéreo, como os dois canais são interpretados como parte real e parte imaginária, e considerados tanto no processo de inserção dos bits quanto no de extração, a limitação observada para áudio mono não ocorre.

III. EXPERIMENTOS

Com a finalidade de ilustrar a aplicação e de coletar resultados da técnica proposta, foram realizados experimentos utilizando um sinal de áudio em suas versões mono e estéreo. As simulações foram realizadas no ambiente MATLAB[®]. O áudio empregado corresponde a um trecho com duração de 60 segundos, em sua versão mono, e de 70 segundos, em sua versão estéreo, da música “Here, there and everywhere”, já mencionada na Seção II, escrito num arquivo em formato *wave*, com frequência de amostragem $f_s = 44.100$ Hz; o vetor resultante possui comprimento $L = 2.646.016$, no caso mono, e $L = 3.087.000$, no caso estéreo. O texto em notação ABC correspondente à partitura da música, também anteriormente apresentado, possui $M = 3.143$ bits.

Os vetores associados aos sinais de áudio foram divididos em segmentos de comprimento igual a 512. Observa-se que, neste caso, apenas parte do vetor é empregada para inserção dos bits, uma vez que $512 \times 3.143 = 1.609.216$; este valor é menor que os comprimentos das versões mono e estéreo do áudio usado nos experimentos. Ao se calcular a DFT com 512 pontos de cada segmento, são selecionadas as componentes da posição 50 à 57 e, dentre essas, considerada a de maior amplitude para inserção propriamente dita de cada bit.

No cenário em questão, poderia ser tomado um número maior de segmentos para inserção de bits. Naturalmente, isso só seria necessário se a partitura da música, ao ser escrita na notação ABC e convertida para binário, produzisse uma quantidade de bits que requeresse tal extensão. No caso do áudio mono, por exemplo,

$$\frac{2.646.016}{512} = 5.168 \text{ bits}$$

poderiam ser inseridos, considerando apenas um trecho com duração de 60 segundos da música. Este número poderia ser ainda duplicado, caso os segmentos tivessem comprimento igual a 256, em vez de 512. Outra alternativa válida seria, no caso do áudio estéreo, considerar os dois canais de forma independente e inserir os bits como se esses canais fossem dois sinais de áudio mono; este procedimento também duplicaria o número de blocos para inserção de bits.

Saber se o número de amostras de uma música em formato digital é suficiente para comportar a inserção dos bits correspondentes a sua partitura é algo bastante subjetivo. Todavia, diante da folga verificada no exemplo analisado, é razoável supor que, mesmo no caso de uma música cuja melodia possui predominantemente notas com tempos curtos e cuja partitura na notação ABC produzirá um número relativamente maior de bits, essa questão não deve constituir um empecilho para aplicação do método proposto. Vale a pena ressaltar que, em esquemas esteganográficos usuais, o número relativo de bits que se consegue ocultar em determinado estego-objeto é um parâmetro de grande importância [2], [3].

TABELA I: Números de bits extraídos com erro para diversos formatos de áudio mono e diferentes combinações dos parâmetros *step* ($\pi/8$, $\pi/7$, $\pi/6$) e γ .

Formato	$\gamma = 0,05$			$\gamma = 0,15$		
	$\pi/8$	$\pi/7$	$\pi/6$	$\pi/8$	$\pi/7$	$\pi/6$
wave	0	0	1	0	0	0
mp3 (320 kbps)	0	0	1	0	0	0
mp3 (256 kbps)	1	1	4	0	0	0
mp3 (192 kbps)	35	16	14	2	3	2

TABELA II: Números de bits extraídos com erro para diversos formatos de áudio estéreo e diferentes combinações dos parâmetros *step* ($\pi/8$, $\pi/7$, $\pi/6$) e γ .

Formato	$\gamma = 0,05$			$\gamma = 0,15$		
	$\pi/8$	$\pi/7$	$\pi/6$	$\pi/8$	$\pi/7$	$\pi/6$
wave	0	0	0	0	0	0
mp3 (320 kbps)	9	3	4	1	0	0
mp3 (256 kbps)	44	51	30	9	6	1
mp3 (192 kbps)	276	248	180	106	72	45

Nas linhas das Tabelas I e II que indicam arquivos no formato *wave*, são mostrados os números de bits errados extraídos dos áudios mono e estéreo, respectivamente. Observa-se que, no caso mono, quando se empregou $\gamma = 0,05$ e $step = \pi/6$, extraiu-se um bit de forma incorreta. Isso se deve à questão abordada na parte final da Seção II. No caso estéreo, todos os bits foram extraídos de forma correta.

Também foram realizados experimentos de extração dos bits da partitura após a conversão dos arquivos de áudio para o formato ISO/MPEG-1 Audio Layer III (comumente chamado mp3); de fato, este é um dos tipos de arquivo de áudio mais difundidos nas redes de comunicação. A conversão para mp3 proporciona a compressão do áudio original (formato *wave*) e pode ser realizada por meio de uma gama bastante ampla de aplicativos. Nos experimentos, foram consideradas três taxas de compressão na conversão do sinal para mp3: 320 kbps, 256 kbps e 192 kbps; com essas taxas, a qualidade do áudio original é preservada e o tamanho dos arquivos produzidos é razoável, considerando o espaço requerido para armazenamento e o tempo para *upload / download* usando uma conexão de banda larga doméstica. O objetivo é verificar quantos bits são extraídos com erro em cada um desses casos e formar uma ideia qualitativa a respeito da robustez do método a este tipo de manipulação¹.

Nas linhas das Tabelas I e II que indicam arquivos no formato mp3, observa-se que erros na extração de bits ocorrem tanto para sinais de áudio mono quanto para estéreo. Em geral, o número de erros na extração de bits de sinais mono é menor, o que sugere que o método, quando aplicado a sinais estéreo,

¹Usualmente, erros de bits são apresentados sob a forma de taxas (BER, *bit-error rate*). Todavia, como o número total de bits inseridos no método proposto é pequeno quando comparado, por exemplo, a experimentos que envolvem o envio de bits por um canal de comunicação, optou-se por preencher as tabelas apresentadas com o número absoluto de bits recuperados com erro; em vez de enfatizar a ordem de grandeza das taxas de erro, essa alternativa favorece uma análise mais precisa dos resultados.

possui maior sensibilidade à compressão mp3. Enquanto a sensibilidade do método a mudanças no *step* depende também da taxa de compressão em sinais mono, para sinais estéreo, a tendência é que o número de bits extraídos com erro diminua com a diminuição deste parâmetro. Quando γ muda de 0,05 para 0,15, nos dois tipos de áudio e para todas as taxas de compressão, as taxas de bits extraídos com erro diminuem. Naturalmente, isso acontece ao preço de uma maior distorção do sinal, visto que γ representa um incremento direto na amplitude da componente que recebe o bit. Ainda assim, a referida distorção mantém-se dentro de uma faixa aceitável de valores; em todos os casos (para áudio mono e estéreo), a relação sinal-ruído² foi superior a 50 dB.

A. Discussão

Os números de bits extraídos com erro e apresentados nas Tabelas I e II variam de 0 a 276; o último valor representa cerca de 9% de bits errados. O impacto desses resultados sobre a recuperação efetiva da partitura musical escrita na notação ABC requer uma avaliação e abre novas possibilidades. Em primeiro lugar, é importante observar que a posição, no arquivo de texto associado, de um bit recuperado com erro pode produzir consequências bem distintas. Se o erro ocorrer em caracteres associados às notas musicais, pausas ou divisões de compassos (a partir da linha 11, no caso do exemplo fornecido na Seção II), estes símbolos podem mudar de posição ou ser trocados por símbolos diferentes, produzindo uma partitura incorreta; outra possibilidade é estes símbolos se transformarem em símbolos que não têm sentido na notação ABC, impedindo a interpretação automática realizada por ferramentas como a disponibilizada em <http://www.mandolintab.net/abcconverter.php>.

Se o erro ocorrer no cabeçalho do texto (linhas 1 a 10, no caso do exemplo já mencionado), a consequência pode ser menos crítica; o título da música ou os nomes dos seus autores podem aparecer com erro. O compasso, o andamento ou o tom também podem sofrer alterações. O fato é que toda a estrutura e a simbologia empregadas na notação ABC possuem propriedades das quais se pode tirar proveito para corrigir os erros cometidos na extração dos bits pelo método proposto. As notas musicais, por exemplo, só podem ser representadas por letras pertencentes ao conjunto {a, b, c, d, e, f, g}, bem como por suas versões em letras maiúsculas. Algo semelhante ocorre com os caracteres usados para atribuir os tempos de duração das notas, as pausas, o término de um compasso etc.

Nas informações contidas no cabeçalho, observa-se algo semelhante: na linha 6 do exemplo, em que se indica o compasso da música, normalmente aparece 2/4, 3/4 ou 4/4; a linha 9, que indica o tom da música, também só pode ser preenchida por uma letra de A a G. Isso significa que é possível atrelar ao processo de extração dos bits um esquema que faz uso de uma espécie de proteção desigual de erros [9]. Para ilustrar essa possibilidade, pode-se considerar novamente os caracteres usados para indicar o tom da música, os quais possuem a seguinte representação em binário:

A: 10100001 B: 10100010 C: 10100011
D: 10100100 E: 10100101 F: 10100110
G: 10100111

A partir da relação acima, conclui-se que, ainda que os cinco primeiros bits relacionados a este campo sejam extraídos com erro, o caractere correto é recuperado. Naturalmente, para determinar a capacidade de detecção e a de correção associadas a cada campo presente num texto em notação ABC, é necessário um estudo mais sistemático e criterioso da estrutura de cada linha. De qualquer modo, embora esse e outros aspectos ainda se encontrem sob investigação, é razoável afirmar que os números de erros contidos nas tabelas apresentadas podem ser reduzidos. Tais reduções também podem ser conseguidas por meio de ajustes nos parâmetros usados na inserção dos bits ou ainda pelo emprego de códigos de bloco usuais, uma vez que o número de segmentos criados a partir do sinal de áudio seria suficiente para acomodar os bits de dados e os de redundância.

IV. CONSIDERAÇÕES FINAIS

Neste artigo, foi descrito um método para ocultação de partituras musicais em sinais de áudio. A proposta emprega uma técnica de modulação de fase e é aplicável a sinais mono e estéreo. A distorção provocada pelo método no sinal que recebe os bits resultantes da partitura é tolerável e, mesmo após uma compressão mp3, mais de 90% desses bits podem ser recuperados corretamente. Atualmente, têm sido investigadas estratégias para elevar a robustez da técnica à referida compressão e avaliados trabalhos com os quais se possa realizar comparações de desempenho.

AGRADECIMENTOS

Os autores agradecem à Coordenação de Aperfeiçoamento de Pessoal de Nível Superior (CAPES) e ao Conselho Nacional de Desenvolvimento Científico e Tecnológico (CNPq).

REFERÊNCIAS

- [1] N. F. Mir, *Computer and Communication Networks*. Prentice Hall, 2nd ed., 2014.
- [2] I. Cox, M. Miller, J. Bloom, J. Fridrich, e T. Kalker, *Digital Watermarking and Steganography*. The Morgan Kaufmann Series in Multimedia Information and Systems, Morgan Kaufmann, 2nd ed., 2007.
- [3] J. Fridrich, *Steganography in Digital Media*. Cambridge University Press, 1st ed., 2009.
- [4] A. A. Zharkikh e A. V. Gurin, "Methods for data embedding into an audio file without changing the data and audio file structures under inverse transformation," *Pattern Recognition and Image Analysis*, vol. 23, p. 160–167, March 2013.
- [5] M. Fan e H. Wang, "Chaos-based discrete fractional sine transform domain audio watermarking scheme," *Computers & Electrical Engineering*, vol. 35, p. 506–516, May 2009.
- [6] J. Chambers, "Abc music notation," <http://trillian.mit.edu/~jc/music/abc/doc/ABCTutorial.html>.
- [7] B. Chen e G. W. Wornell, "Quantization index modulation: a class of provably good methods for digital watermarking and information embedding," *IEEE Transactions on Information Theory*, vol. 47, p. 1423–1443, May 2001.
- [8] X. Dong, M. F. Bocko, e Z. Ignjatovic, "Data hiding via phase manipulation of audio signals," in *Proc. of the IEEE International Conference on Acoustics, Speech, and Signal Processing*, (Montreal, Canada), p. 377–380, 2004.
- [9] S. Borade, B. Nakiboglu, e L. Zheng, "Unequal error protection: An information-theoretic perspective," *IEEE Transactions on Information Theory*, vol. 55, p. 5511–5539, December 2009.

²Neste caso, é considerada ruído a diferença entre o sinal original e o sinal após a inserção dos bits.