Ajuste Fino de Parâmetros de Redes Neurais por Convolução Utilizando o Algoritmo de Otimização das Aves Migratórias

Bárbara Caroline Benato, Aparecido Nilceu Marana, João Paulo Papa
Departamento de Computação
Faculdade de Ciências, UNESP
Bauru, Brasil
Email: barbarabenato@gmail.com, nilceu/papa@fc.unesp.br

David Cox
Centro de Ciência do Cérebro
Universidade de Harvard
Cambridge, Estados Unidos
Email: davidcox@fas.harvard.edu

***Resumo*—O problema de ajuste de parâmetros em técnicas de aprendizado em profundidade tem sido considerado nos últimos anos, uma vez que o ajuste manual é mais propenso erros. Neste trabalho, introduzimos a técnica de Otimização das Aves Migratórias (*Migrating Birds Optimization* - MBO) para ajustar os parâmetros de Redes Neurais por Convolução (*Convolutional Neural Networks* - CNNs) e Redes de Crença Profunda (*Deep Belief Networks* - DBNs), comparando os resultados entre outras duas técnicas meta-heurísticas consideradas estados-da-arte. Os experimentos mostraram que MBO obteve bons resultados tanto para CNNs, quanto para DBNs, embora com um alto custo computacional.**

***Keywords*-Aprendizado em Profundidade; Otimização Meta-heurística.**

***Abstract*—The problem of fine-tuning parameters in deep learning techniques has been considerably focused in the last years, since to hand-tune them is painful and prone to errors. In this work, we introduced the Migrating Birds Optimization (MBO) to fine-tune parameters of Convolutional Neural Networks (CNNs) and Deep Belief Networks (DBNs), being the results compared against two other state-of-the-art meta-heuristic techniques. The experiments showed MBO obtained very good results in both CNNs and DBNs, but at the price of a high computational burden.**

***Keywords*-Deep Learning; Meta-heuristic Optimization.**

I. Introdução

Recentemente, técnicas baseadas em "aprendizado em profundidade", do inglês *deep learning*, têm alcançado resultados interessantes em um número considerável de problemas como, por exemplo, reconhecimento facial [1] e identificação de fala [2]. Em suma, redes neurais encontram padrões a partir de exemplos de entrada por meio de um aprendizado não-supervisionado, que é utilizado, posteriormente, para alimentar classificadores supervisionados. Redes Neurais por Convolução (*Convolutional Neural Networks* - CNNs [3]) e Máquinas de Boltzmann Restritas (*Restricted Boltzmann Machines* - RBMs [4]) são técnicas amplamente utilizadas para tal tarefa, por serem consideradas estado-da-arte nesse contexto e pela alta disponibilidade de implementações com código aberto, bem como por serem constantemente aprimoradas durante os últimos anos.

Um dos maiores problemas relatados ao aprendizado em profundidade é atribuído à grande quantidade de parâmetros a serem ajustados, geralmente, de forma manual. Tal processo consome tempo e necessita de um usuário com conhecimento prévio, tornando a tarefa mais sensível a erros. A proposta deste trabalho é lidar com tal deficiência, de forma a modelar o problema de ajuste de parâmetros como uma tarefa de otimização. De fato, qualquer abordagem poderia ser utilizada, mas o uso de otimização meta-heurística mostra-se ser mais interessante, uma vez que não são técnicas baseadas em derivadas, trabalhosas em espaço de busca multidimensional. Contudo, poucos trabalhos têm considerado aplicar técnicas meta-heurísticas para esta concepção.

Papa et al. [5], [6] propuseram ajustar parâmetros de RBMs e Redes de Crença Profunda (*Deep Belief Networks* -DBNs [7]), usando a Busca Harmônica (HS), um algoritmo de otimização meta-heurística baseado no processo de criação de músicas. De forma similar, Papa et al. [8] empregaram HS e Otimização por Exame de Partículas (*Particle Swarm Optimization* - PSO), para aprender parâmetros de Máquinas de Boltzmann Restritas Discriminativas (*Discriminative Restricted Boltzmann Machines* - DRBMs), e Rosa et al. [9] propuseram ajustar os parâmetros de CNNs por meio do HS.

Apesar das técnicas meta-heuristicas mencionadas terem obtido resultados interessantes em várias aplicações, não garantem que a melhor solução seja encontrada. Recentemente, Duman et al. [10] apresentaram o algoritmo de Otimização das Aves Migratórias (*Migrating Birds Optimization* - MBO), baseado no comportamento do voo de aves migratórias em formação 'V'. Essa formação é conhecida por favorecer a aerodinâmica do grupo todo e, assim, economizar energia durante o processo de migração. Contudo, até onde se sabe, o MBO ainda não recebeu a atenção adequada na tarefa de ajuste de parâmetros de técnicas de aprendizado em profundidade. Assim, as principais contribuições deste trabalho são: (i) introduzir o algoritmo MBO para o ajuste de parâmetros de CNNs e (ii) DBNs, bem como (iii) enriquecer a literatura de meta-heurísticas e aprendizado em profundidade. A fim de avaliar o desempenho do MBO neste contexto,

também foram considerados PSO, HS e parâmetros ajustados manualmente, considerando CNNs e DBNs. Os experimentos com MBO apresentaram resultados considerados estado-da-arte para algumas situações, como apresentado no decorrer do artigo.

II. Técnicas de Aprendizado em Profundidade

A. Redes Neurais por Convolução

Redes do tipo CNN são uma representação de modelos baseados na arquitetura de Hubel e Wiesel, os quais realizaram um estudo seminal em 1962 sobre o córtex primário de gatos. Esse trabalho identificou, basicamente, dois tipos de células: (i) *células simples*, que possuem uma tarefa análoga à etapa de filtragem por banco de máscaras, e (ii) as *células complexas*, as quais realizam uma tarefa semelhante à etapa de amostragem das CNNs. O primeiro modelo que simulou uma Rede Neural por Convolução, o amplamente conhecido "Neocognitron [11]", aplicava um algoritmo de treinamento não supervisionado na etapa de filtragem por banco de máscaras e um supervisionado na última camada da rede. Posteriormente, LeCun et al. [12] propuseram a utilização do algoritmo de retropropagação para o treinamento da rede toda de maneira supervisionada.

Basicamente, uma CNN pode ser entendida como *L-camadas* de processamento de imagens (ou sinais) em cadeia, que objetivam extrair uma representação de alto nível de determinada entrada, isto é, um vetor de caracaterísticas concatenado, utilizado para posterior aplicação de técnicas de reconhecimento de padrões. Usualmente, temos 3 operações subjacentes para cada camada da CNN, sendo a primeira, uma operação de *convolução* com um banco de filtros, seguida por uma *amostragem* e, então, uma etapa de *normalização*.

B. Redes de Crença Profunda

As DBNs podem ser compreendidas como um tipo de RBM, isto é, uma rede neural estocástica baseada em energia composta por duas camadas de neurônios (visível e escondida), em que a fase de aprendizado é conduzida de forma não-supervisionada. As RBMs são similares às Máquinas de Boltzmann clássicas exceto por não serem permitidas conexões entre neurônios da mesma camada. De grosso modo, a ideia é alimentar a rede com exemplos não-rotulados e então reconstruir o dado de entrada. O processo de reconstrução é usualmente realizado pela etapa de amostragem na cadeia de Markov a fim de aproximar a verossimilhança logarítmica da imagem reconstruída em relação a imagem original.

DBNs são compostas por um conjunto de RBMs empilhadas, que são posteriormente treinadas por meio de um processo de aprendizado guloso. As unidades escondidas de cada camada tornam-se unidades de entrada da camada seguinte.

III. Otimização das Aves Migratórias

A técnica MBO é baseada na formação 'V' durante o voo das aves migratórias, a qual é uma disposição espacial bastante eficiente para minimizar a energia gasta durante o voo. O formato da asa de um pássaro é similar ao de um aerofólio, ou seja, "corta" o ar quando o mesmo encontra-se em direção contrária a ele. O ar deslocado por uma ave é capaz de auxiliar o voo das outras reduzindo o gasto de energia necessário para manter a mesma no ar. Na prática, quanto maior o número de aves em um bando, maior será a energia poupada.

Table I
Configuração de parâmetros.

Técnica	Parametros
PSO	$c_1 = 1{,}7$, $c_2 = 1{,}7$, $w = 0{,}7$
HS	$HMCR = 0{,}7$, $PAR = 0{,}7$, $\varrho = 10$
MBO	$k = 5$, $x = 2$, $m = 5$

A. Algoritmo MBO

O algoritmo MBO realiza uma busca pela vizinhança de uma ave, codificando cada ave na formação 'V' como uma possível solução do problema. O algoritmo verifica se cada ave pode ser melhorada analisando a sua vizinhança. Se alguma ave oferecer uma solução melhor, ela é substituída. Depois de todas as aves do bando serem avaliadas, a melhor solução é deslocada para a última posição da formação 'V' e a segunda solução torna-se a nova líder. Esse procedimento é realizado até algum critério de convergência ser atingido. O número de vizinhos utilizados para avaliar uma solução pode ser interpretado como a força necessária para levantar voo. Um maior número de vizinhos sugere menor velocidade de voo e uma melhor percepção dos detalhes à sua volta.

IV. Metodologia

A. Configuração Experimental

O foco do trabalho é avaliar o desempenho do MBO considerando o problema de ajuste de parâmetros em técnicas baseadas em aprendizado em profundidade. Em suma, a ideia é codificar um conjunto de parâmetros para cada abordagem (CNN/DBN) em uma única solução (partícula/harmonia/ave). A partir de um conjunto de soluções possíveis geradas aleatoriamente, o processo objetiva mover partículas no espaço de busca, a fim de minimizar uma função de aptidão. Considerando CNNs, objetivamos minimizar a função de perda chamada "softmax" e, para DBN, minimizar o erro médio quadrático (*mean square error* - MSE). Foram comparados MBO com PSO e HS, sendo o primeiro uma técnica metaheurística baseada em enxame e uma das mais utilizadas na literatura. O HS é uma alternativa para trazer variabilidade para a seção experimental, uma vez que costuma ser mais rápido e não baseado em enxames. A Tabela I apresenta a configuração dos parâmetros para cada uma dessas técnicas. Tais parâmetros foram escolhidos empiricamente.

1) Parâmetros da CNN: Para CNNs, foi utilizada a implementação oriunda da Caffe[1], uma biblioteca de código aberto e desenvolvida em plataforma GPGPU (*General-purpose Computing on Graphics Processing Units*). Os experimentos foram conduzidos por meio um procedimento de validação cruzada com 10 rodadas e 15 soluções

[1]http://caffe.berkey.vision.org (Acesso em: 28 ago. 2016)

(partículas/harmonias/aves), bem como 50 iterações para convergência considerando todos os algoritmos de otimização. O número máximo de iterações da CNN foi definido como 200. A arquitetura da CNN para a base de dados MNIST foi configurada com três camadas, cada uma composta por uma operação de covolução e uma de amostragem. O passo para todas as camadas de convolução foi inicializado dentro do intervalo $[1, 3]$ e os núcleos de amostragem no intervalo $[1, 5]$. A respeito das outras duas bases de dados, usamos a CNN com cinco camadas de convolução e três de amostragem. O passo da primeira camada de convolução foi inicializado dentro do intervalo $[1, 4]$, enquanto das camadas restantes, $[1, 3]$. O núcleo de convolução para a primeira camada foi definido dentro do intervalo $[1, 11]$, para a segunda camada, $[1, 5]$, e para as três camadas restantes, $[1, 3]$. Com relação à camada de amostragem, foram utilizados valores de passos dentro do intervalo $[1, 2]$ para as primeiras duas camadas, e $[1, 3]$ considerando as outras. O tamanho dos núcleos de amostragem foi inicializado no intervalo $[1, 3]$ e, para a última camada, $[1, 5]$.

2) Parâmetros da DBN: Foi empregado um processo de validação cruzada com 20 rodadas, 5 soluções e 50 iterações. Os parâmetros da DBN foram definidos de acordo com os seguintes intervalos: $n \in [5, 100]$, $\eta \in [0{,}1, 0{,}9]$, $\lambda \in [0{,}1, 0{,}9]$ e $\alpha \in [0{,}00001, 0{,}01]$, onde n e η denotam o número de neurônios escondidos e taxa de aprendizado, respectivamente. Variáveis α e λ denotam o parâmetro de penalidade e momento, respectivamente. Também foi empregado $T = 10$ como o número de épocas para os pesos de aprendizado da DBN com lotes de tamanho 20. A fim de promover uma maior validação experimental, as DBNs foram treinadas com três diferentes algoritmos: Divergência Contrastiva (*Contrastive Divergence* - CD), Divergência Constrastiva Persistente (*Persistent Contrastive Divergence* - PCD) e Persistência Constrastiva Persistente Rápida (*Fast Persistent Contrastive Divergence* - FPCD).

B. Bases de Dados

Considerando os experimentos da CNN, três conjuntos de bases para a tarefa de reconhecimento de expressões faciais foram utilizados, bem como um dos mais conhecidos para o reconhecimento de dígitos manuscritos, como descrito abaixo:

- MNIST[2]: essa base de dados é composta por imagens de dígitos manuscritos. Sua versão original contém um conjunto de treinamento com 60.000 imagens dos dígitos '0'-'9', e para teste de 10.000 imagens.
- *Japanese Female Facial Expression* (JAFFE)[3]: essa base contém 213 imagens de 7 expressões faciais (6 expressões básicas e 1 neutra) obtidas de 10 mulheres japonesas. Foram utilizadas 128 de treinamento e 85 para teste.
- *Cohn-Kanade AU-Coded Expression*[4]: a base contém 486 imagens de 97 indivíduos com diferentes poses e expressões faciais. Entretanto, apenas as imagens com pose frontal foram utilizadas, totalizando 251 imagens, sendo 147 de treinamento e 104 para teste.
- *Taiwanese Facial Expression*[5]: a base consiste em 7.200 imagens capturadas de 40 modelos, contendo 8 expressões faciais (neutra, raiva, desprezo, medo, felicidade, tristeza e surpresa). Apenas as imagens de pose frontal foram utilizadas, totalizando 320 imagens, sendo 192 de treinamento e 128 para teste.

Para os experimentos da DBN, usamos três conjuntos de dados, além da já mensionada MNIST, como descrito abaixo:

- *CalTech 101 Silhouettes*[6]: essa base de dados é baseada na famosa base Caltech 101, sendo composta por silhuetas de imagens distribuídas em 101 classes.
- *Semeion Handwritten Digit*[7]: é formada por 1.593 imagens de digitos manuscritos de '0' - '9', escritos de modo normal (preciso) e, também, de modo rápido (impreciso).

V. Resultados Experimentais

A. CNN

As tabelas II, III, IV e V apresentam as taxas médias de reconhecimento considerando os conjuntos de dados MNIST, JAFFE, CK e TFEID, respectivamente. Adicionalmente, o número de chamadas ao processo de treinamento é apresentado. As técnicas com melhor acurácia estão em negrito.

Table II
Acurácia Média sobre o conjunto de teste para a base MNIST.

	Acurácia (%)	# chamadas
Caffe	$\mathbf{98{,}18 \pm 0{,}00}$	**1**
PSO	$93{,}86 \pm 0{,}01$	750
HS	$97{,}65 \pm 0{,}00$	65
MBO	$94{,}68 \pm 0{,}003$	11.750

Table III
Acurácia Média sobre o conjunto de teste para a base JAFFE.

	Acurácia (%)	# chamadas
Caffe	$29{,}29 \pm 0{,}03$	1
PSO	$28{,}51 \pm 0{,}04$	750
HS	$24{,}74 \pm 0{,}04$	65
MBO	$\mathbf{34{,}05 \pm 0{,}09}$	**11.750**

Table IV
Acurácia Média sobre o conjunto de teste para a base CK.

	Acurácia (%)	# chamadas
Caffe	$67{,}18 \pm 0{,}06$	1
PSO	$68{,}18 \pm 0{,}05$	750
HS	$\mathbf{68{,}51 \pm 0{,}05}$	**65**
MBO	$65{,}35 \pm 0{,}05$	11.750

A primeira conclusão interessante refere-se à etapa de otimização através de meta-heurísticas, a qual mostra-se apta a obter melhores resultados do que a CNN padrão e ajustada

[2]http://yann.lecun.com/exdb/mnist/ (Acesso em: 28 ago. 2016)
[3]http://www.kasrl.org/jaffe.html (Acesso em: 28 ago. 2016)
[4]http://www.pitt.edu/ emotion/ck-spread.htm (Acesso em: 28 ago. 2016)
[5]http://bml.ym.edu.tw/tfeid/ (Acesso em: 28 ago. 2016)
[6]https://people.cs.umass.edu/ marlin/data.shtml (Acesso em: 28 ago. 2016)
[7]https://archive.ics.uci.edu/ml/datasets/Semeion+Handwritten+Digit (Acesso em: 28 ago. 2016)

Table V
ACURÁCIA MÉDIA SOBRE O CONJUNTO DE TESTE PARA A BASE TFEID.

	Acurácia (%)	# chamadas
Caffe	$91{,}17 \pm 0{,}01$	1
PSO	$\mathbf{93{,}78 \pm 0{,}02}$	**750**
HS	$92{,}11 \pm 0{,}01$	65
MBO	$93{,}40 \pm 0{,}02$	11.750

Table VI
MÉDIA MSE SOBRE O CONJUNTO DE TEST PARA A BASE MNIST.

	1L			2L			3L		
	CD	PCD	FPCD	CD	PCD	FPCD	CD	PCD	FPCD
PSO	0,1057	0,1058	0,1057	0,1060	0,1059	0,1058	0,1058	0,1059	0,1058
HS	0,1059	0,1325	0,1324	0,1059	0,1061	0,1057	0,1059	0,1058	0,1057
MBO	**0,0876**	**0,0876**	**0,0895**	**0,0876**	**0,0876**	**0,0884**	**0,0876**	**0,0876**	**0,0880**

manualmente para três de quatro conjuntos de dados. Embora MBO tenha obtido o melhor resultado para apenas um conjunto de dados, os seus resultados estão bem próximos aos melhores considerando as bases CK e TFEID. Contudo, a principal deficiência atribuída ao MBO está relacionada à sua complexidade computacional. Observando-se o número de chamadas à função de aptidão, é possível verificar que HS requer 65 avaliações (50 iterações e 15 possíveis soluções, isto é $50 + 15 = 65$), PSO requer 750 avaliações ($50 * 15 = 750$) e MBO necessita de 11.750 avaliações, onde número de chamadas é dado por $K{*}m * [k + (N-1) * (k-x)]$, com K sendo os *tours*, m as aves, k as aves vizinhas, N a dimensão do problema e x as aves vizinhas compartilhadas.

B. DBN

Os experimentos da DBN utilizaram 3 modelos distintos: com uma (1L), duas (2L) e três (3L) camadas. A abordagem 1L refere-se a RBM padrão. Consideramos, também, três algoritmos de aprendizado: CD, PCD e FPCD. As TabelasVI, VII e VIII apresentam os resultados MSE para as bases MNIST, Caltech 101 Silhouettes e Semeion Handwritten Digits sobre o conjunto de teste, respectivamente. As técnicas com melhor acurácia estão em negrito, de acordo com o desvio padrão aqui omitido.

Pode-se observar que o MBO obteve o melhores resultados para os conjuntos de dados, embora uma diferença estatística significante não tenha sido notada entre os modelos. Isso deve-se ao reduzido número de iterações para convergência. Atualmente, se um número considerável de iterações for utilizado, então o espaço de busca randômico deve convergir, mas a um alto custo computacional.

Table VII
MÉDIA MSE SOBRE O CONJUNTO DE TEST PARA A BASE CALTECH 101SILHOUETTES.

	1L			2L			3L		
	CD	PCD	FPCD	CD	PCD	FPCD	CD	PCD	FPCD
PSO	0,1691	0,1690	0,1689	0,1689	0,1691	0,1688	0,1692	0,1692	0,1690
HS	0,1695	0,1696	0,1691	0,1695	0,1699	0,1693	0,1694	0,1696	0,1692
MBO	**0,1566**	**0,1583**	**0,1609**	**0,1606**	**0,1606**	**0,1609**	**0,1606**	**0,1607**	**0,1609**

Table VIII
MÉDIA MSE SOBRE O CONJUNTO DE TEST PARA A BASE SEMEION HANDWRITTEN DIGIT.

	1L			2L			3L		
	CD	PCD	FPCD	CD	PCD	FPCD	CD	PCD	FPCD
PSO	0,2128	0,2128	0,2128	0,2128	0,2128	0,2128	0,2128	0,2128	0,2127
HS	0,2128	0,2128	0,2129	0,2202	0,2128	0,2128	0,2103	0,2109	0,2119
MBO	**0,1977**	**0,2020**	**0,2073**	**0,2096**	**0,2096**	**0,2098**	**0,2096**	**0,2096**	**0,2100**

VI. CONCLUSÕES

Neste trabalho, introduzimos o MBO para o ajuste de parâmetros para ambas as técnicas: CNNs e DBNs. Mesmo exigindo um custo computacional considerável, o MBO é capaz de encontrar taxas de classificação satisfatórias considerando CNNs e, também, obter os menores erros de reconstrução para todos os conjuntos de dados relativos aos experimentos com DBNs. A respeito de trabalhos futuros, objetivamos buscar por outras variantes do algoritmo MBO, que sejam mais rápidas e apresentem maior acurácia.

AGRADECIMENTOS

Os autores agradecem à FAPESP pelos subsídios #2014/16250-9, #2014/12593-9, #2014/25214-6, #2014/24491-6 and #2015/25739-4, e CNPq pelos subsídios #470571/2013-6 e #306166/2014-3.

REFERENCES

[1] Y. Taigman, M. Yang, M. Ranzato, and L. Wolf, "Deepface: Closing the gap to human-level performance in face verification," in *Proceedings of the 2014 IEEE Conference on Computer Vision and Pattern Recognition*, ser. CVPR '14. Washington, DC, USA: IEEE Computer Society, 2014, pp. 1701–1708.

[2] G. E. Hinton, D. Li, Y. Dong, G. E. Dahl, A. Mohamed, N. Jaitly, A. Senior, V. Vanhoucke, P. Nguyen, T. N. Sainath, and B. Kingsbury, "Deep neural networks for acoustic modeling in speech recognition: The shared views of four research groups," *IEEE Signal Processing Magazine*, vol. 29, no. 6, pp. 82–97, 2012.

[3] Y. LeCun, L. Bottou, Y. Bengio, and P. Haffner, "Gradient-based learning applied to document recognition," in *Intelligent Signal Processing*, S. Haykin and B. Kosko, Eds. IEEE Press, 2001, pp. 306–351.

[4] G. Hinton, "Training products of experts by minimizing contrastive divergence," *Neural Computation*, vol. 14, no. 8, pp. 1771–1800, 2002.

[5] J. P. Papa, G. H. Rosa, K. A. P. Costa, A. N. Marana, W. Scheirer, and D. D. Cox, "On the model selection of bernoulli restricted boltzmann machines through harmony search," in *Proceedings of the GECCO*. New York, NY, USA: ACM, 2015, pp. 1449–1450.

[6] J. P. Papa, W. Scheirer, and D. D. Cox, "Fine-tuning deep belief networks using harmony search," *Applied Soft Computing*, pp. –, 2015.

[7] G. E. Hinton, S. Osindero, and Y.-W. Teh, "A fast learning algorithm for deep belief nets," *Neural Computation*, vol. 18, no. 7, pp. 1527–1554, 2006.

[8] J. P. Papa, G. H. Rosa, A. N. Marana, W. Scheirer, and D. D. Cox, "Model selection for discriminative restricted boltzmann machines through meta-heuristic techniques," *Journal of Computational Science*, vol. 9, pp. 14–18, 2015.

[9] G. H. Rosa, J. P. Papa, A. N. Marana, W. Scheirer, and D. D. Cox, "Fine-tuning convolutional neural networks using harmony search," in *Progress in Pattern Recognition, Image Analysis, Computer Vision, and Applications*, ser. Lecture Notes in Computer Science, A. Pardo and J. Kittler, Eds. Springer International Publishing, 2015, vol. 9423, pp. 683–690, 20th Iberoamerican Congress on Pattern Recognition.

[10] E. Duman, M. Uysal, and A. F. Alkaya, "Migrating birds optimization: A new metaheuristic approach and its performance on quadratic assignment problem," *Information Sciences*, vol. 217, pp. 65–77, 2012.

[11] K. Fukushima and S. Miyake, "Neocognitron: A new algorithm for pattern recognition tolerant of deformations and shifts in position," *Pattern Recognition*, vol. 15, no. 6, pp. 455–469, 1982.

[12] Y. LeCun, B. Boser, J. S. Denker, D. Henderson, R. E. Howard, W. Hubbard, and L. D. Jackel, "Backpropagation applied to handwritten zip code recognition," *Neural Computation*, vol. 1, no. 4, pp. 541–551, 1989.