Preface

This volume contains a collection of extended and short abstracts of talks presented at **NumAn 2007**. The Conference had as stated aims to bring together and bequeath scientific activities, directions and pursuits of Greek scientists in Greece and abroad on subjects that pertain to the conference; to foster an exchange of views and ideas; to study the theoretical background required for methods, algorithms and techniques used in applications; to search directions of theoretical results towards applications; to highlight open problems and future directions of numerical analysis. The volume also contains contributions by several international participants; we are grateful to them for choosing to contribute to this event. All this, in the magnificent surroundings of **Kalamata**, a wonderful city in the **Messinia** region of southern Greece.

The majority of the papers deal (primarily) with computational topics in the areas of Linear Algebra, Partial Differential Equations and Optimization. There are also papers in Differential, Integral and Nonlinear Equations, Error Analysis, Approximation, and applications such as Fluid Dynamics, Weather Prediction, Statistics and Data Mining.

Several distinguished senior speakers are invited to honor this conference. These are Professors N. Apostolatos (University of Athens), N. Artemiadis (Aca demy of Athens), D. Bertsekas (MIT), C. Dafermos (Brown University), A. Fokas (Academy of Athens and Cambridge), A. Hadjidimos (Thessaly), E. Houstis (Thessaly), P. Ligomenides (Academy of Athens), G. Nicolis (Academy of Athens and Université Libre de Bruxelles) and C. Tsallis (Brazilian Academy of Sciences and Santa Fe Institute). We take this opportunity to extend our warmest welcome to all of them.

We would finally like to extend our thanks to those individuals, universities and corporations that helped in the organization of NumAn 2007. First of all to the members of the program committees and to the researchers who submitted their work for presentation. Without them, NumAn and this volume would not have materialized. We also express our deep gratitude to Professor A. Hadjidimos for his efforts on behalf of the Conference, efforts that went far beyond his duties as Program Committee member. The hard work and help of Michael G. Epitropakis and Aris G. Vrahatis, both students at the Mathematics Department of the University of Patras proved to be invaluable in many critical aspects of local organization and in the compilation of this volume. We finally thank our academic institutions, specifically the University of Patras and Wilfrid Laurier University, the local authorities, specifically the Prefecture of Messinia and the Municipality of Kalamata, and finally Maplesoft and MP & Associates, for supporting this conference. We hope that the present volume will be a good and useful record of NumAn 2007.

The NumAn 2007 Organizing Committee:

M.N. Vrahatis, E. Gallopoulos, I.S. Kotsireas, D. Noutsos, E. Houstis

Organization of the Conference

CONFERENCE IN NUMERICAL ANALYSIS (NumAn 2007) RECENT APPROACHES TO NUMERICAL ANALYSIS: THEORY, METHODS AND APPLICATIONS

September 3-7, 2007 Kalamata, Hellas.

http://www.math.upatras.gr/numan2007/

numan2007@math.upatras.gr

Aims:

1. To bring together and bequeath scientific activities, directions and pursuits of Greek scientists in Greece and abroad on subjects that pertain to the conference.
2. To foster an exchange of views and ideas.
3. To study the theoretical background required for methods, algorithms and techniques used in applications.
4. To search directions of theoretical results towards applications.
5. To highlight open problems and future directions of numerical analysis.

Invited Speakers:

- Prof. N. Apostolatos, University of Athens.
- Prof. N. Artemiadis, Academy of Athens.
- Prof. D. Bertsekas, Massachusetts Institute of Technology.
- Prof. C. Dafermos, Brown University.
- Prof. A. Fokas, Academy of Athens and University of Cambridge.
- Prof. A. Hadjidimos, University of Thessaly.
- Prof. E. Houstis, University of Thessaly.
- Prof. P. Ligomenides, Academy of Athens.
- Prof. G. Nicolis, Academy of Athens and Université Libre de Bruxelles.
- Prof. C. Tsallis, Brazilian Academy of Sciences and Santa Fe Institute, USA.

Discussion Panel:

N. Apostolatos, N. Artemiadis, D. Bertsekas, C. Dafermos, A. Fokas, A. Hadjidimos, E. Houstis, P. Ligomenides, G. Nicolis, C. Tsallis.

Tutorial Committee:

A. Hadjidimos, K.I. Iordanidis, N. Papamichael, T. Papatheodorou.

Publicity Committee:

I.Z. Emiris, E. Gallopoulos, E. Houstis, N.S. Stylianopoulos.

Local Organizing Committee:

H.N. Vrahati, I.S. Kotsireas, M.N. Vrahatis.

Organizing Committee:

E. Gallopoulos, E. Houstis, I.S. Kotsireas, D. Noutsos, M.N. Vrahatis.

Scientific Committee:

- G. Akrivis (Ioannina), N. Alikakos (Athens),
- G.S. Androulakis (Patras), N. Apostolatos (Athens),
- N. Artemiadis (Athens), I. Athanasopoulos (Heraklion),
- G. Athanasoulis (Athens), G. Avdelas (Livadia),
- A. Bacopoulos (Athens), C.N. Bekas (IBM Zurich),
- D.P. Bertsekas (MIT), G. Biros (Pennsylvania),
- C. Botsaris (Livadia), A. Boudouvis (Athens),
- A. Bountis (Patras), C.C. Christara (Toronto),
- N. Chrisochoides (Washington), D. Christodoulou (ETH),
- A.T. Chronopoulos (Texas), C.M. Dafermos (Brown),
- G. Dassios (Patras), I.C. Demetriou (Athens),
- V.A. Dougalis (Athens), P. Drineas (RPI),
- L.B. Drossos (Messologhi), I.Z. Emiris (Athens),
- C.A. Floudas (Princeton), A.S. Fokas (Cambridge),
- E. Gallopoulos (Patras), G. Georgiou (Nicosia),
- T.N. Grapsa (Patras), A. Hadjidimos (Thessaly),
- E. Houstis (Thessaly), K.I. Iordanidis (Patras),
- P.D. Kaklis (Athens), A. Karageorghis (Nicosia),
- D. Karras (Chalkis), E. Kaxiras (Harvard),
- I.G. Kevrekidis (Princeton), E.J. Kontoghiorghes (Nicosia),
- I.S. Kotsireas (Waterloo), C. Koukouvinos (Athens),
- P. Koumoutsakos (ETH), E. Kranakis (Ottawa),
- I. Lagaris (Ioannina), P. Ligomenides (Athens),
- E. Lipitakis (Athens), G.D. Magoulas (London),
- Ch. Makridakis (Heraklion), A. Makroglou (Portsmouth),
- G. Maroulis (Patras), G. Meletiou (Arta),
- J.G. Michopoulos (Washington), N.M. Missirlis (Athens),
- M. Mitrouli (Athens), G. Nicolis (Brussels),
- D. Noutsos (Ioannina), R.C. Papademetriou (Portsmouth),
- N. Papamichael (Nicosia), K. Papadakis (Patras),

– S. Papadopoulou (Heraklion), E. Papadrakakis (Athens),
– G. Papageorgiou (Athens), T. Papatheodorou (Patras),
– P.M. Pardalos (Florida), N.M. Patrikalakis (MIT),
– E.A. Perdios (Patras), O. Ragos (Patras),
– Ph. Rozakis (Heraklion), Y. Saridakis (Chania),
– Y. Smyrlis (Nicosia), A. Stathopoulos (Virginia),
– N.S. Stylianopoulos (Nicosia), C. Tsallis (Santa Fe and CBPF),
– M. Tsatsomeros (Washington), Ch. Tsitouras (Chalkis),
– J.N. Tsitsiklis (MIT), F.N. Valvi (Patras),
– E. Vavalis (Volos), S.A. Vavasis (Waterloo),
– V. Vlachou (Patras), M.N. Vrahatis (Patras),
– L. Xanthis (London), F.A. Zafiropoulos (Patras),

Special thanks:

– Municipality of Kalamata,
– University of Patras, Hellas,
– Wilfrid Laurier University, Canada,
– MapleSoft,
– Journal of Computational and Applied Mathematics.
– MP & Associates

Table of Contents

Abstracts of Invited Speakers Talks

Educational Systems in Mathematics and Other Related Topics.

N. Artemiadis

Academy of Athens

Approximate Solution of Very Large Linear Systems of Equations by Simulation[1]

Dimitris P. Bertsekas

Massachusetts Institute of Technology MIT

Abstract. We consider linear fixed point equations, and approximation of the solution by projection on a low-dimensional subspace. We propose stochastic iterative algorithms, based on simulation, which converge to the approximate solution and are suitable for very large-dimensional problems. The methodology generalizes recent approximate dynamic programming methods, which solve a projected form of Bellman's equation by using simulation-based approximations to this equation, or by using a projected value iteration method.

Hyperbolic Balance Laws with Dissipation

C. Dafermos

Brown University

Abstract. Global BV solutions will be constructed to the Cauchy problem for hyperbolic systems of balance laws with dissipative sources induced by relaxation mechanisms.

[1] joint research with Janey Yu (University of Helsinki)

Integrability, boundary value problems and imaging

A. Fokas

Academy of Athens and University of Cambridge

Abstract. Ideas from the theory of nonlinear integrable PDEs have led to the introduction of a new method for solving boundary value problems. This is turn has led to new techniques for the numerical integration of linear evolution PDEs and for elliptic PDEs in a convex polygon. In addition, ideas from integrable PDEs have led to new developments in medical imaging. In this talk, some of these recent developments will be reviewed.

Using Extrapolation for the Solution of the Linear Complementarity Problem[1]

A. Hadjidimos

University of Thessaly

Abstract. The *Linear Complementarity Problem* (LCP) has many applications as, e.g., in the solution of Linear and Convex Quadratic Programming, in Free Boundary Value problems of Fluid Mechanics, etc. In the present work we consider the case where the matrix coefficient $A \in \mathbb{R}^{n,n}$ of LCP is a *positive definite matrix*. Considering a known iterative method for the solution of LCP we introduce the principle of *Extrapolation* and find the best *extrapolation* parameter ω for which the corresponding extrapolated iterative scheme converges asymptotically faster. Various simple and more complicated numerical examples show that it is worth using extrapolation to solve an LCP.

[1] joint work with Michael Tzoumas (University of Ioannina)

Mathematics in the Knowledge Economy

E. Houstis

University of Thessaly

Abstract. The knowledge economy (KE) can be viewed as the combination of information and computer technologies (ICT) and highly skilled labor to produce economic benefits for the whole of our society. Education and Innovation are two of the pillars of this new economy. ICT technologies are becoming part of every productive organization of the human enterprise, from manufacturing to entertainment, telecommunications, transportation and education. ICT is changing the way we deliver, consume, and administer education. Today, we are educating a new generation of engineers and scientists raised on "Sesame Street" like programs, where learning through dynamic visual imagery is emphasized. This generation is immersed in an electronic media world surrounded by television, MP3 music boxes, multimedia cellular phones and Bluetooth/WiFi communication devices, digital cameras, internet cafe, electronic games and toys. Against this background, interactive multimedia based learning is becoming the norm. At the same time, mathematics is increasingly used in almost all areas of human activity and recognized as the basis of KE. It is the language in which knowledge is expressed for solving problems with computers. However, mathematics as a discipline affecting only a small minority of the so-called mathematically gifted people. In this presentation, we will examine proposals to reconfigure the way we educate in this discipline the next generation of engineers & scientists in the context of KE and present some tools for modeling "innovation" in the design of new "products".

The Reality of Mathematics

P. Ligomenides

Academy of Athens

Abstract. The unreasonable effectiveness of mathematics. Why are the laws of nature mathematical?? Why is the physical world knowable?? Computability, Compressibility, and the Physical laws. Scientific mysticism.

Computational Science and Complex Systems Research

G. Nicolis

Academy of Athens and Université Libre de Bruxelles

Abstract. Complex systems are ubiquitous in nature, technology and everyday life. They can give rise under certain conditions to a multitude of states associated with such properties as the spontaneous emergence of large scale self-organizing patterns, adaptation, evolution, or the generation and processing of information. These phenomena unfold on a wide spectrum of space and time scales and need thus to be analyzed by means of a bottom-up, multilevel approach integrating microscopic as well as macroscopic formulations, deterministic as well as probabilistic views, structural as well as dynamical aspects. In this presentation some key problems of present day complex systems research are reviewed and the challenges posed by them to computational science are brought out. Examples of successful cross-fertilization between the two fields are given and a number of significant problems that remain open are identified.

Entropy, nonextensive statistical mechanics, and numerical applications

C. Tsallis

Brazilian Academy of Sciences and Santa Fe Institute, USA

Abstract. The usual Boltzmann-Gibbs-Shannon entropy S_{BGS} and its associated statistical mechanics are well known to be adequate for systems whose elements are not too strongly correlated. Whenever this simplifying hypothesis is not satisfied, we can use instead the nonadditive entropy S_q (with $S_1 = S_{BGS}$), and its associated nonextensive statistical mechanics. The whole theory generalizes that of Boltzmann-Gibbs. Applications to areas such as global optimization techniques, signal and image processing, will be presented as well.
Bibliography: http://tsallis.cat.cbpf.br/biblio.htm

Abstracts of Articles

Optimal detection of redundant features via artificial neural network pruning using genetic algorithms

Adam Adamopoulos

Medical Physics Laboratory
Department of Medicine
Democritus University of Thrace
681 00, Alexandroupolis, Hellas
adam@med.duth.gr

Introduction

The method proposed in the present work is related to the problem of the optimal detection of feature redundancy in artificial neural network (ANN) inputs. The subsequent ANN pruning that the detection and elimination of the redundant and / or overlapping imposes lead to ANN pruning and size minimization. The method is based on Genetic Algorithm (GA) search [1] to detect the optimal (minimal) subset of ANN input parameters that should be used for ANN training, in order the trained ANN to achieve the lowest error possible at the testing phase. The part of the input parameters that is not included in the optimal (minimal) subset of the essential input parameters, are considered as redundant and therefore can be omitted and eliminated during ANN training and testing. Obviously, the investigation for a minimal subset of ANN training parameters is justified only in the case that these parameters that are not omitted and are used for ANN training and testing manage to retain, if not to improve, ANN performance in terms of the success on the task that the ANN was designed for. In that case, the detection of redundancy and overlapping of the input parameters may support the elimination of the redundant input parameters. This may lead to the development of even simpler ANN architectures since fewer neural nodes should be necessary to built-in the input and the hidden layers of the ANN. Simpler ANN architecture with decreased number of neural nodes and synaptic connections may result to less complicated and less time-consuming training and testing procedures and at the same time to performance improvement. Nowadays increased computer power and the contemporary development of ANN training algorithms, provide the essential means for fast and accurate implementation of ANN techniques to solve problems of various types, in different scientific fields, (classification, prediction, system identification, to name a few). Despite the fact that ANN training and testing even for complicated problems and large data sets is accomplished with low computational cost, it is legitimate, if not desirable, to investigate for even faster, more reliable and more accurate ANN training and testing methods. The present work is focused on the detection of any kind

of redundancy in the data sets that are used as inputs during ANN training and testing. Redundancy and overlapping in data sets that are used for ANN training, when exists, definitely increase computational cost for ANN training, while at the same time, may mislead, or suppress the training procedure. This could be generate problems in terms of ANN ability to accomplish successfully and with the desired accuracy the task that it was designed for.

Methods

The problem described above is faced with the use of GA search. For any specific problem, given the data set that is used for ANN, the detection of any redundant and overlapping input parameters is attempted by performing the following two steps: (1) full training and testing, and, (2) pruned training and testing.
As a first step, the ANN is fully trained, that means, all given input parameters in the original data set are used to train the ANN, and all parameters are used to test the performance of the previously trained ANN. The mean square error (MSE) generated by the trained ANN is saved in order to be compared to the corresponding results that would be generated by the ANN that will be trained with a subset of the original data. Ten independent experiments are performed and the mean MSE is recorded.
As a second step a GA is invoked selection of training parameters: In the second step, pruned training and testing is performed. A GA is utilized to search for the optimal subset of the input parameters that should be used for training and testing of the pruned ANN. The individuals of the GA population are consisted of binary strings of length equal to the number of parameters of the original data set. Therefore, each gene of the individual corresponds to an input parameter. Since genes are binary digits, the allele 0 denotes that the corresponding parameter is not included in the subset of the parameters that will be used for ANN training and testing and therefore is omitted. On the opposite, the allele 1 denotes that the corresponding parameter will be considered for ANN training and testing. Since binary representation is adopted for the individuals of the GA, all the well known genetic operators for selection, crossover and binary mutation can be applied on the GA population. Two are the fitness functions that were used in order to evaluate the individuals

$$f_1 = MSE \tag{1}$$

and

$$f_2 = MSE + \frac{I}{N} \tag{2}$$

where, in Eq. (2), I is the number of input parameters used for ANN training and testing, and N is the total number of input parameters in the original, full-sized data set. Obviously, by using f_2 as fitness function, the GA searches to optimize two objectives: first to construct the minimum training data set, and second to minimize the MSE generated at the test phase. For each individual

of the GA, an ANN is constructed, with the number of input nodes that is indicated by the individual. In all cases, the number of hidden neurons is equal to the number of input nodes. Subsequently, the constructed ANN are trained and tested and the generated MSE is recorded.

Results and Discussion

The proposed methodology was applied on two different kinds of data and on ANN that are used for two different purposes: (1) breast cancer classification, and, (2) Lorenz system timeseries prediction. In the first case the ANN is used for classification, whereas in the second case the ANN is used for timeseries prediction.

Breast cancer classification

Breast cancer classification data were provided by the UCI Machine Learning Repository [2]. This data set is consisted of 683 cases of breast cancer, classified by medical experts to 239 (37.46%) malignant and to 444 (62.54%) benign cases.

Nine cell features (input parameters) were considered during the clinical evaluation of these cases. All these 9 features were evaluated by medical experts in the range from 1 to 10. Namely, the 9 cell features considered in the study, are the following:

(1) Clump Thickness
(2) Uniformity of Cell Size
(3) Uniformity of Cell Shape
(4) Marginal Adhesion
(5) Single Epithelial Cell Size
(6) Bare Nuclei
(7) Bland Chromatin
(8) Normal Nucleoli
(9) Mitoses

Results of the application of the first step of the proposed methodology, that is, full-sized ANN training and testing with all the 9 features as input, for 10 independent experiments, generated mean MSE = 0.0215 and standard deviation 0.0086. On the other hand, results of the GA search for the optimal subset of input parameters indicated that that data subsets consisted of features (1, 2, 6) generated MSE = 0.0032, the subset of features (1, 2, 6, 9) generated MSE = 0.0029, and the subset of features (1, 3, 6, 7) generated MSE = 0.0037.

In all cases, the subsets of input features obtained by the application of the GA resulted to pruned ANN with MSE on the test set ranging from 13.49% up to 17.21% of the MSE of the full-trained ANN. So, ANN using only 3 or 4 of the 9 in total input features, that is, pruned down to the 1/3 of the original full-size, exhibited remarkable improvement in breast cancer classification.

The Lorenz system

The Lorenz system [3] is described by the following triplet of coupled differential equations:

$$dx/dt = -ax + ay \tag{3}$$
$$dy/dt = -xz + rx - y \tag{4}$$
$$dz/dt = xy - bz \tag{5}$$

where, by setting $a = 10$, $r = 28$, and $b = 8/3$ the well-known non-linear dynamics and a strange attractor are observed. The z-component of the derived triple timeseries was considered for prediction. Ten consecutive samples constructed the input patterns to train an ANN to predict the value of the following sample. When trained and tested with the full-sized input data for 10 independent experiments, the mean MSE derived was $5.8598 \cdot 10^{-7}$. The application of the GA search for the optimal subset of inputs indicated that samples 2, 4, 5, 7, and 8 resulted to MSE $= 2.5482 \cdot 10^{-7}$ on the test set. That is, by using only 5 over 10 (50%) of the input parameters, the predicting ability of the ANN is even improved, since the generated MSE on the test set is reduced down to 43.48% of the corresponding MSE of the full-trained ANN.

The results presented above clearly indicate that the proposed method managed to unveil redundancy and overlapping in data set of both cases. Elimination of the redundant input parameters led to simpler ANN architecture and at the same time to performance improvement in both pattern classification and timeseries prediction tasks.

References

1. Michalewicz Z., Genetic Algorithms + Data Structures = Evolution Programs. Springer-Verlag, 1996.
2. http://www.ics.uci.edu/ mlearn/MLRepository.html
3. Sparrow, C., The Lorenz Equations: Bifurcations, Chaos, and Strange Attractors. New York: Springer-Verlag, 1982.

On an Iterative Algorithm for $H-$Matrices: The Irreducible and Reducible Case

Maria Alanelli[1] and Apostolos Hadjidimos[2]

[1] Department of Mathematics, University of Crete, GR-714 09 Heraklion, Greece (alanelli@math.uoc.gr)

[2] Department of Computer and Communication Engineering, University of Thessaly, 10 Iasonos Street, GR-383 33 Volos, Greece (hadjidim@inf.uth.gr)

Abstract. $H-$matrices appear in many areas of science and engineering, e.g., in the solution of the Linear Complementarity Problem (LPC) in Optimization Theory, in the solution of Free Boundary Value Problems of Fluid Mechanics, etc. Several authors have proposed direct and iterative criteria to identify whether a certain $n \times n$ complex matrix is an $H-$matrix. Based on previous and new ideas we propose a new iterative algorithm (Algorithm $\mathbb{AH}$) for irreducible matrices that decides about the $H-$ or non $H-$matrix character of a certain matrix in a finite number of steps. As an extension of it, Algorithm $\mathbb{AH}2$ is also proposed to cover the *reducible* case as well.

Introduction

The theory of $H-$matrices is very important for the numerical solution of linear systems of algebraic equations arising in various applications. E.g.: a) In the Linear Complementarity Problem (LPC) in Optimization Theory [1] (see also Section 10.1 of [3]), in the Free Boundary Value Problems in Fluid Analysis [3], etc. An $H-$matrix $A \in \mathbb{C}^{n,n}$ can be defined in several ways the most common of which is by means of a *strictly diagonally dominant* matrix. Specifically:

Definition 1. *$A \in \mathbb{C}^{n,n}$ is an $H-$matrix if and only if there exists a positive diagonal matrix $D \in \mathbb{R}^{n,n}$ so that AD is* **(row-wise) strictly diagonally dominant**, *that is $|a_{ii}|d_i > \sum_{j=1,\ j\neq i}^{n} |a_{ij}|d_j,\ i = 1(1)n$.*

The reader is also reminded that: i) An $H-$matrix is also called **generalized (row) strictly diagonally dominant (GSDD)** matrix. ii) Definition 1 implies the nonsingularity of A which is consistent with the original definition by Ostrowski [18] (see also [21]).

For the identification of an $H-$matrix A many criteria have been proposed the majority of which are iterative ones (see, e.g., [11], [15], [14], [16], [17], [10] and [2]). This is because direct criteria (see, e.g., [8], [12], [9], [7] and [5]) have high computational complexities. It is noted that the only iterative criterion that takes into account the sparsity of A is the one in [10], where an extension of the *compact profile* technique of [13] was developed.

Having as starting points the Algorithms in [15], [16] and [17] a new Algorithm is proposed (analytically given in [2]), Algorithm $\mathbb{AH}$, to deal with irreducible matrices **only**. Together with it an extension of Algorithm $\mathbb{AH}$ which covers the reducible case as well, is Algorithm $\mathbb{AH}2$.

Algorithm $\mathbb{AH}$

For both Algorithms the following matrices are needed and are defined. A sequence of positive diagonal matrices that will be defined in the Algorithm $D^{(k)},\ k=0,1,2,\ldots,\ D^{(0)}=I$, and the matrices $A^{(k)}=\left(D^{(k-1)}\right)^{-1}A^{(k-1)}D^{(k-1)}$, $k=1,2,3,\ldots,\ A^{(0)}=(\mathrm{diag}(A))^{-1}A$, assuming that $a_{ii}\neq 0,\ i=1(1)n$. From them, it is readily seen that $a_{ii}^{(k)}=1,\ i=1(1)n,\ k=0,1,2,\ldots$. Algorithm $\mathbb{AH}$ is presented below.

Algorithm $\mathbb{AH}$.

INPUT: An irreducible matrix $A:=[a_{ij}]\in\mathbb{C}^{n,n}$.
OUTPUT: $D=D^{(0)}D^{(1)}\cdots D^{(k)}\in\mathfrak{D}_{D^{-1}A}\equiv\mathfrak{D}_A$ or $\notin\mathfrak{D}_A$ if A **is** or is **not** an $H-$matrix, respectively
1. If $a_{ii}=0$ for some $i\in\mathbb{N}$, "A is **not** an $H-$matrix", STOP; Otherwise
2. Set $D=I$, $A^{(0)}=(\mathrm{diag}(A))^{-1}A$, $D^{(0)}=I$, $k=1$
3. Compute $D=DD^{(k-1)}$, $A^{(k)}=\left(D^{(k-1)}\right)^{-1}A^{(k-1)}D^{(k-1)}=[a_{ij}^{(k)}]$
4. Compute $s_i^{(k)}=\sum_{j=1,\ j\neq i}^{n}|a_{ij}^{(k)}|,\ i=1(1)n,\ s^{(k)}=\min_{i=1(1)n}s_i^{(k)},\ S^{(k)}=\max_{i=1(1)n}s_i^{(k)}$
5. If $s^{(k)}>1$, "A is **not** an $H-$matrix", STOP; Otherwise
6. If $S^{(k)}<1$, "A **is** an $H-$matrix", STOP; Otherwise
7. If $S^{(k)}=s^{(k)}$, "$\mathcal{M}(A)$ is **singular**", STOP; Otherwise
8. Set $d=[d_i]$, where $d_i=\frac{1+s_i^{(k)}}{1+S^{(k)}},\ i=1(1)n$
9. Set $D^{(k)}=\mathrm{diag}(d),\ k=k+1$; Go to Step 3. END

For Algorithm $\mathbb{AH}$ the following two statements were proved in [2]:

Theorem 1. *Let $A\in\mathbb{C}^{n,n}$ be an irreducible matrix. Then, Algorithm $\mathbb{AH}$ always converges (except, maybe, when $\det(\mathcal{M}(A))=0$).*

Theorem 2. *Let $A\in\mathbb{C}^{n,n}$ be any irreducible matrix. If Algorithm $\mathbb{AH}$ converges, then its output is correct.*

If for the irreducible matrix $A\in\mathbb{C}^{n,n}$, with $a_{ii}\neq 0,\ i=1(1)n$, we set as in Algorithm $\mathbb{AH}$ $A^{(k)}=\left(\mathrm{diag}(d_1^{(k-1)},d_2^{(k-1)},\ldots,d_n^{(k-1)})\right)^{-1}A^{(k-1)}\mathrm{diag}(d_1^{(k-1)},d_2^{(k-1)},\ldots,d_n^{(k-1)})$, with $d^{(0)}=e$, where $e\in\mathbb{R}^n$ is the vector of ones, and $|A^{(k)}|=I+B^{(k)}$, $k=0,1,2,\ldots$, and where $|X|$ denotes the matrix whose elements are the moduli of the corresponding elements of X, we note that $B^{(0)}$ is the Jacobi matrix associated with the comparison matrix of A, $J_{\mathcal{M}(A)}$. If in the Algorithm we allow $k\to\infty$ then in the proofs of Theorems 1 and 2 it was also proved in [2], among others, that:

Corollary 1. *Under the assumptions and notations so far the Perron vector* d *of* $|A^{(0)}|$ *(and* $B^{(0)}$*) is given by* $d = \left(\lim_{k\to\infty}\left(\prod_{i=1}^{k} D^{(i)}\right)\right) e$. *Also* $\lim_{k\to\infty} |A^{(k)}|\, e = \rho(|A^{(0)}|)\, e$. *Moreover,* $\lim_{k\to\infty} a_{ij}^{(k)} = \frac{d_j}{d_i}\, a_{ij}^{(0)}$.

Even though Algorithm $\mathbb{AH}$ was designed to work for *irreducible* matrices, we had observed that it worked perfectly well for certain classes of *reducible* matrices. This motivated the investigation of the effect of the application of Algorithm $\mathbb{AH}$ to reducible matrices a little further. So, we were led to an extension of it and created Algorithm $\mathbb{AH}$**2** which is shown to converge in all possible cases of *irreducible* and *reducible* matrices.

Algorithm $\mathbb{AH}$2

Using material from combinatorial matrix theory and results from [19], [20] we manage to tackle the problem of the general $p \times p$ block *reducible* case in connection with Algorithm $\mathbb{AH}$. For the new algorithm we need the definitions below.

Let $\mathbb{N} := \{1, 2, \ldots, n\}$ and $\mathbb{N}_0^{(k)} \equiv \mathbb{N}_0(A^{(k)}) := \left\{ i \in \mathbb{N} : |a_{ii}^{(k)}| \le s_i^{(k)} \right\}$, where $s_i^{(k)}$ is defined in Step 4 of Algorithm $\mathbb{AH}$, and let $n_0^{(k)} := n_0(A^{(k)})$ be the cardinality of $\mathbb{N}_0^{(k)}$.

Algorithm $\mathbb{AH}$**2**.

INPUT: A matrix $A := [a_{ij}] \in \mathbb{C}^{n,n}$ and a maximum number of iterations allowed ("maxit")
OUTPUT: $D = D^{(0)} D^{(1)} \cdots D^{(k)} \in \mathfrak{D}_{D^{-1}A} \equiv \mathfrak{D}_A$ or $\notin \mathfrak{D}_A$ if A **is** or is **not** an $H-$matrix, respectively
1. If $a_{ii} = 0$ for some $i \in \mathbb{N}$, "A is **not** an $H-$matrix", STOP; Otherwise
2. Set $D = I$, $A^{(0)} = (\mathrm{diag}(A))^{-1} A$, $D^{(0)} = I$, $k = 1$
3. Compute $D = DD^{(k-1)}$, $A^{(k)} = \left(D^{(k-1)}\right)^{-1} A^{(k-1)} D^{(k-1)} = [a_{ij}^{(k)}]$
4. Compute $s_i^{(k)},\ i = 1(1)n,\ s^{(k)} = \min_{i=1(1)n} s_i^{(k)},\ S^{(k)} = \max_{i=1(1)n} s_i^{(k)}$
5. If $s^{(k)} > 1$, "A is **not** an $H-$matrix", STOP; Otherwise
6. If $S^{(k)} < 1$, "A **is** an $H-$matrix", STOP; Otherwise
7. If $S^{(k)} = s^{(k)}$, "$\mathcal{M}(A)$ is **singular**", STOP; Otherwise
8. Set $d = [d_i]$, where $d_i = \frac{1+s_i^{(k)}}{1+S^{(k)}},\ i = 1(1)n$
9. Set $D^{(k)} = \mathrm{diag}(d)$, If $k <$ maxit, $k = k+1$, Go to Step 3; Otherwise
10. Determine $\mathbb{N}_0^{(\mathrm{iter})}$ and $n_0^{(\mathrm{iter})}$
11. If $n_0^{(\mathrm{iter})} = 1$, "Inconclusive, increase maxit", STOP; Otherwise
12. Compute $s_{i_j}^{(\mathrm{iter})} = \sum_{l=1,\ l\neq j}^{n_0^{(\mathrm{iter})}} |a_{i_j,i_l}^{(\mathrm{iter})}|,\quad j = 1(1)n_0^{(\mathrm{iter})},\quad i_j, i_l \in \mathbb{N}_0^{(\mathrm{iter})}$
13. If $s_{i_j}^{(\mathrm{iter})} \ge 1$, $j = 1(1)n_0^{(\mathrm{iter})},\quad i_j \in \mathbb{N}_0^{(\mathrm{iter})}$, "$A$ is **not** an $H-$matrix", STOP; Otherwise
14. Update $\mathbb{N}_0^{(\mathrm{iter})}$ (by discarding $i_j \in \mathbb{N}_0^{(\mathrm{iter})} : s_{i_j} < 1$) and $n_0^{(\mathrm{iter})}$; Go to Step 11. END

For Algorithm $\mathbb{AH}$**2** we may prove analogous theorems to Theorems 1 and 2 of the previous section.

References

1. B.H. Ahn. *Solution of Nonsymmetric Linear Complementarity Problems by Iterative Methods.* J. Optim. Theory Appl., 33 (1981), 175–185.
2. M. Alanelli and A. Hadjidimos. *A New Iterative Criterion for $H-$Matrices.* SIAM J. Matrix Anal. Appl., to appear.
3. A. Berman and R.J. Plemmons. *Nonnegative Matrices in the Mathematical Sciences. Classics in Applied Mathematics.* SIAM, Philadelphia, 1994.
4. R. Bru and M. Neumann. *Nonnegative Jordan Basis.* Linear Multilinear Algebra, 23 (1988), 95–109.
5. Lj. Cvetković and V. Kostić. *New Criteria for Identifying $H-$Matrices.* J. Comput. Appl. Math., 180 (2005), 265–278.
6. D.K. Faddeev and V.N. Faddeeva. *Computational Methods of Linear Algebra.* W.H. Freeman, San Francisco, 1963.
7. T.-B. Gan and X.-H. Huang. *Simple Criteria for Nonsingular $H-$Matrices.* Linear Algebra Appl., 374 (2003), 317–326.
8. Y.-M. Gao and X.-H. Wang. *Criteria for Generalized Diagonally Dominant Matrices and $M-$Matrices.* Linear Algebra Appl. 169 (1992), 257–268.
9. Y.-M. Gao and X.-H. Wang. *Criteria for Generalized Diagonally Dominant Matrices and $M-$Matrices II.* Linear Algebra Appl., 248 (1996), 339–353.
10. A. Hadjidimos *An Extended Compact Profile Iterative Method Criterion for Sparce $H-$Matrices.* Linear Algebra Appl., 389 (2004), 329–345.
11. M. Harada, M. Usui and H. Niki. *An Extension of the Criteria for Generalized Diagonally Dominant Matrices.* Internat., J. Comput. Math., 60 (1996), 115–119.
12. T.-Z. Huang. *A Note on Generalized Diagonally Dominant Matrices.* Linear Algebra Appl., 225 (1995), 237–242.
13. D.R. Kincaid, J.R. Respess, D.M. Young and R.G. Grimes. *ITPACK 2C: A Fortran Package for Solving Large Sparse Linear Systems by Adaptive Accelerated Iterative Methods.* ACM Trans. Math. Software, 8 (1982), 302–322.
14. T. Konho, H. Niki, H. Sawami and Y.-M. Gao. *An Iterative Test for $H-$Matrix.* J. Comput. Appl. Math., 115 (2000), 349–355.
15. B. Li, L. Li, M. Harada, H. Niki and M.J. Tsatsomeros. *An Iterative Criterion for $H-$Matrices.* Linear Algebra Appl., 271 (1998), 179–190.
16. L. Li. *On the Iterative Criterion for Generalized Diagonally Dominant Matrices.* SIAM J. Matrix Anal. Appl., 24 (2002), 17–24.
17. K. Ojiro, H. Niki and M. Usui. *A New Criterion for $H-$Matrices.* J. Comput. Appl. Math., 150 (2003), 293–302.
18. A.M. Ostrowski. *Über die Determinanten mit Überwiegender Hauptdiagonale.* Comment. Math. Helv., 10 (1937), 69–96.
19. U.G. Rothblum. *Algebraic Eigenspaces of Nonnegative Matrices.* Linear Algebra Appl., 12 (1975), 281–292.
20. H. Schneider. *The Influence of the Marked Reduced Graph of a Nonnegative Matrix on the Jordan Form and Related Properties: A Survey.* Linear Algebra Appl., 84 (1986), 161–189.
21. R.S. Varga. *Matrix Iterative Analysis.* 2nd revised and expanded edition, Springer, Berlin, 2000.

Nonextensive Particle Swarm Optimization Methods

Aristoklis D. Anastasiadis[1], George D. Magoulas[1], George Georgoulas[2] and Anthony Tzes[3]

[1] London Knowledge Lab and School of Computer Science, Birkbeck College, University of London, 23-29 Emerald Street, WC1N 3QS, London, UK.
[2] School of Electrical and Computer Engineering, Georgia Institute of Technology, Atlanta, USA.
[3] Electrical and Computer Engineering Department, University of Patras, Achaia 26500, Hellas.

Abstract. A class of adaptive particle swarm optimization (PSO) methods is presented that combines the traditional position update rule with annealing schedules based on the nonextensive entropy. Preliminary results show that the tested algorithms are very promising, outperforming in most cases the global version of PSO.
Keywords:Particle Swarm Optimizer, Swarm Intelligence, Global Search, Nonextensive Statistical Mechanics, Nonlinear Optimization

Introduction

The Particle Swarm Optimization (PSO) algorithm is an evolutionary computation technique for global optimization [4]. Many variants of the PSO method have been proposed so far following Eberhart and Kennedy's work in this area [4, 5]. A Fuzzy PSO algorithm, proposed by Eberhart [7], demonstrated reliable performance in many cases. Another modification of the PSO, the Hybrid Particle Swarm Optimizer (HPSO) produced even better results on unimodal and multimodal functions [9] using mass extinction. In other PSO methods, velocities were updated using gaussian distributions [6]. In this work, new variants of the PSO algorithm, which are based on nonextensive statistical mechanics [2], are investigated. The next sections present the basic theory underlying our approach, preliminary results and some plans for future work.

Nonextensive Particle Swarm Optimization Methods

PSO is part of the swarm intelligence paradigm and it is based on updating a population of potential solutions. In mathematical notation this is expressed by the following set of equations:

$$v_i(t+1) = \phi(t)v_i(t) + \eta_1 r[p_i - x_i(t)] + \eta_2 r[p_{g(t)} - x_i(t)], \quad (1)$$

$$x_i(t+1) = x_i(t) + v_i(t), \quad (2)$$

where i is the particle's index, $v_i(t)$ is the current velocity of the i–th particle, $\phi(t)$ is an inertia function (usually a linearly decreasing one), $x_i(t)$ is the current position of the i–th particle, p_i is the position with the best fitness value visited by the i–th particle, $g(t)$ is the particle with the best fitness among all the particles (best position found so far – global version of the PSO), r is a positive constant called acceleration constant and η_1, η_2 are random vectors uniformly distributed in $[0, 1]$.

In our proposed approach, the PSO based methods are characterized by the nonextensive entropic index q. In particular, Tsallis has defined the nonextensive entropy [2]:

$$S_q \equiv K \, \frac{1 - \sum_{i=1}^{W} p_i^q}{q - 1} \quad (q \in \mathbb{R}), \tag{3}$$

where W is the total number of microscopic configurations, whose probabilities are $\{p_i\}$, and K is a conventional positive constant. When the entropic index $q = 1$, Equation (3) recovers to Boltzmann–Gibbs entropy. The entropic index works like a biasing parameter: $q < 1$ privileges rare events (values of p close to 0 are benefited), while $q > 1$ privileges common events (values of p close to 1). The optimization of the entropic form (3) under appropriate constraints, [2], yields for the canonical ensemble:

$$p_i \propto [1 - (1 - q)\beta E_i]^{\frac{1}{(1-q)}} \equiv e_q^{-\beta E_i}, \tag{4}$$

where β is a Lagrange parameter, $\{E_i\}$ is the energy spectrum, and the *q-exponential function* is defined as:

$$e_q^x \equiv [1 + (1 - q)x]^{\frac{1}{(1-q)}} = \frac{1}{[1 - (q - 1)x]^{\frac{1}{(q-1)}}} \tag{5}$$

The first PSO variant, called *Nonextensive Hybrid PSO*– NHPSO, incorporates stochasticity in search by adopting the following model:

$$Q_{(T,k)} = e_q^{-T(\ln 2)\cdot k} = [1 - (1 - q)T(\ln 2) \cdot k]^{\frac{1}{1-q}}, \tag{6}$$

where T is the temperature and k indicates iterations. In this approach particles have no neighborhood restrictions, the velocity equation uses an inertia weight as in the classical PSO method and the location of a particle is updated as follows:

$$x_{id}^{k+1} = x_{id}^k + Q_{i(T,k)} \cdot v_{id}^k, \tag{7}$$

where $Q_{i(T,k)}$ is defined by Eq. (6). By tuning the entropic index q and the temperature T, the term $Q_{i(T,k)}$ provides an alternative to using a fixed constriction coefficient [3] to control the velocity term without compromising the diversity of the search. In this way, we can have better control and constriction of the velocity.

The second PSO variant described here is inspired by [1, 8] and uses a cooling procedure. This defines the relationship between the Temperature T and the entropic index q values. The application of cooling helps to regulate better the PSO algorithm. This new *Nonextensive Evolving Particle Swarm Optimization*– NEPSO behaves in a more stochastic way during the initial stages, compared to the PSO and NHPSO, and then becomes more deterministic as the number

of iterations increases. Thus, as we are getting close to the neighborhood of a minimizer, the algorithm hopefully will avoid oscillations and converge faster. In this approach the velocity equation as well as the location of a particle are updated as in NHPSO but the term $Q_{(T,k)}$ is changing dynamically by the cooling procedure that is described by the next equation:

$$T = T_0 \cdot [\frac{2^{q-1} - 1}{(1+k)^{q-1} - 1}], q > 1, \tag{8}$$

where T_0 is the initial temperature, T is the current temperature, k is the number of iterations, and q is the Tsallis entropic index. The cooling procedure makes the temperature to decrease as a power–law of time, in contrast to the much slower decrease (logarithmic in time) of the $q = 1$ case.

Simulation Study

In this section, we evaluate the performance of the Nonextensive Particle Swarm Optimization (NPSOs) algorithms and compare them with with the standard PSO (SPSO). Figure 1 shows the average fitness performance calculated over 20 runs for the new class of the tested algorithms compared with the standard PSO. The population size is 20 and the dimension is also 20 for all graphs. The new class of the PSO methods converges faster than the Standard PSO.

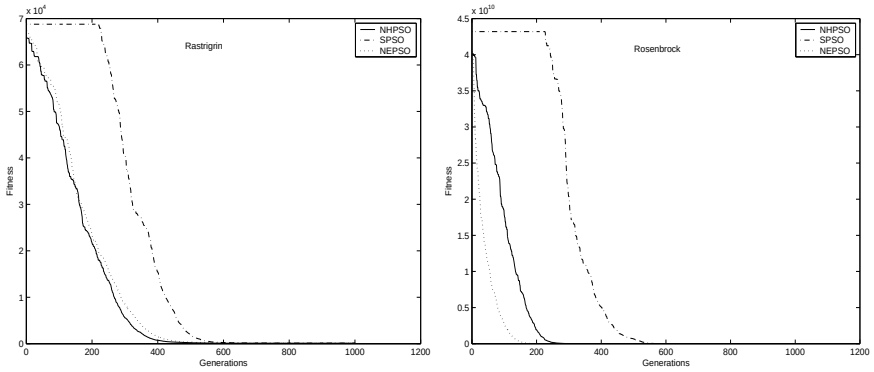


Fig. 1. The average fitness performance for Standard PSO (SPSO) and the new modifications in the Rastrigrin and Rosenbrock functions.

Discussion and Concluding Remarks

In this paper, we proposed a new class of Particle Swarm Optimization algorithms, which constitute efficient modifications of the standard PSO method. The approach builds on principles from nonextensive statistics and the two algorithms, called NHPSO and NEPSO, are characterized by the nonextensive

entropic index q. A formula that adaptively controls the relationship between the T and q parameters was also proposed.

Further experiments, which are not reported here due to lack of space, show that there is a range of q values ($1.1 < q < 4.0$) that allows the new PSO methods to achieve reliable and improved performance when compared with other PSO variants, such as the fuzzy PSO and the hybrid PSO.

Further testing is of course necessary to fully explore the advantages and identify possible limitations of this class of PSO methods. Moreover, exhaustive testing of the new methods in other classes of problems is under way.

Acknowledgments

Aristoklis Anastasiadis gratefully acknowledges support by the London Knowledge Lab, UK, as well as the University of Patras, Electrical and Computer Engineering Department, Greece.

References

[1] A. Anastasiadis and G. Magoulas. Evolving stchastic learning algorithms based on tsallis entropic index., *European Physical Journal B.*, 50 (2006), 277-283.

[2] C.Tsallis. Possible generalization of boltzmann-gibbs statistics. *J. Statistical Physics*, 52(1-2), (1988), 479–487.

[3] R. Eberhart and Y. Shi. Comparing inertia weights and constriction factors in particle swarm optimization,. In Proc. 2000 Congress Evolutionary Computation, San Diego, CA, (2000). 84–88.

[4] J. Kennedy and R. Eberhart. Particle swarm optimization. In Proc. IEEE Int. Conf. on Neural Networks, (1995), 1942–1948.

[5] J. Kennedy and R. Eberhart. Swarm Intelligence. Morgan Kaufmann Publishers, 2001.

[6] H.-F. Krohling, R. A. and L. dos Santos Coelho. Co-evolutionary particle swarm optimization for min-max problems using gaussian distribution. IEEE Proc. Congress Evolutionary Computation, 1, (2004), 959–964.

[7] Y. Shi and R. Eberhart. Fuzzy adaptive particle swarm optimization. In Proc. IEEE Int. Conf. on Evolutionary Computation., (2001), 101–106.

[8] C. Tsallis and D. A. Stariolo. Generalized simulated annealing. *Physica A*, (1996), 395–406.

[9] Z.-L. Y. Xiao-Feng Xie, Wen-Jun Zhang. Hybrid particle swarm optimizer with mass extinction. In Int. Conf. on Communication, Circuits and Systems (ICCCAS), (2002), 1170–1174.

On the prediction of time series' local optima: a backtrack technique

George S. Androulakis and Eleni G. Lisgara

Department of Business Administration, University of Patras,
GR–265.00 Patras, Greece
<gandroul|lisgara>@upatras.gr

Abstract. In this paper is attempted to propose a backtrack technique that can approximate a time series' future optima. The estimation of the future optimum is based on sequenced points produced from the repetitive process of the continuous optima finding. Additionally, it is proved that the use of a proper optimization technique finally results to local optimum point.

Introduction

Times series is considered as a sequence of data points arranged according to time. In finance the daily stock prices may comprise a time series, and in meteorology, daily maximum or minimum temperatures may report one, too. Agriculture, physics, ecology and demography, as most scientific fields interested in reporting data based on time observations tend to produce time series reports.

Apart from the data history itself, time series has promoted into a major forecasting tool, based on statistical methodologies that use historical data to predict not future points, but the future prices, of any time series regardless their data content.

In this paper we propose a backtrack technique that allows any optimization algorithm that obtains "memory" being applied in finding future local optima. Section 4 includes a brief literature review of most methodologies based on which time series prediction is made. The methodology proposed in 4 shows how the proposed backtrack technique is generated, while section 4 promotes further research interests and applications.

Time Series prediction techniques

The most applicable form of a time series assumes that if Y_1, Y_2, ..., Y_n forms the time series, at time n for $n \geq 1$, interest is focused in predicting the next point's value Y_{n+1} based on the observed realizations of Y_1, Y_2, ..., Y_n, [1]. It was not until H. Markowitz [2] proposed the use of the mean-variance model that would finally predict future prices quite accurate in respect to the real ones. Based on his pioneer contribution that future points may be detected through

the historical information provided by past data, and statistical assumption including means, variances and covariances, many applications in several subject areas introduced. Several bibliography on time series forecasting for finance incorporated with the probability theory [3–9].

Furthermore, the distinctive introduction of the exponential smoothing model provided by Brown [10] and Box and Jenkins [11] arose new evidence towards predicting time series prediction most efficiently. Such methodologies applied the so called auto-regressive integrated moving average (ARIMA) models to find the best fit of a time series on its own past values; the effectiveness of both methodologies though is a rather controversial issue [12].

Most methodologies put aside the issue of the next point on which a research may focus and be interested in, and they highlight forecasting the next value. It was the financial research the first to focus on the next point's prediction instead of the next value's prediction, since portfolio optimization itself craves for the best time-to-market regardless prices. Therefore, [13] and [14] included the concept of the stock market timing in theoretical means, involving financial trends and macroeconomic policies. Applications appeared at the late 90's, when [15] and [16] incorporated with the best time-to-market issue in terms of Artificial Intelligence to predict future stock price movements including weighted factors such as past data and market volatility.

Additionally, [17, 18] incorporated the time series modeling and prediction through the spectrum of feed-forward neural networks as one-step local predictors applied on exchange rates. [19, 20] incorporated results from genetic algorithms and neural network applications together, in a single multiobjective algorithm to conclude that the obtained results appear more accurate than the single use of one technique.

The Backtrack Procedure

Lots of the unconstrained minimization methods are iterative in nature and hence they start from an initial random solution, $X_0 \in \mathbb{R}^n$, and proceed towards the minimum point in a sequential manner. The general iterative scheme of an optimization algorithm is shown in Figure 1.

Recall that a time series Y as a sequence of data points t produces the $Y = Y_t : t \in T$ function, where t data points only refer to time. Let Y_t be a time series and Y_n be the time series' value. The $f(t)$ function interprets the continuity that any time series is characterized from, and its value at the last known data point is equal to the Y_t.

In many applications, predictions are made regardless the next point's value but concentrate when the maximum or minimum would appear i.e. in weather forecasting, knowing the most highest temperatures during summer is rather useful that just know the next day's -probably ordinary- temperature. In mathematical means this could be translated as the local minimum and maximum of the function $f(t)$, respectively.

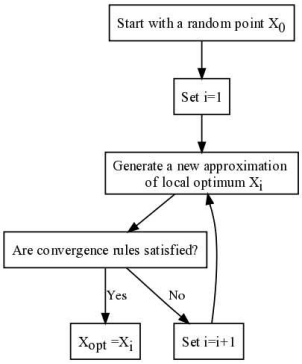


Fig. 1. General iterative scheme for optimization

When applying the optimization techniques, it is concluded that most of them in order to generate the new point t_{n+k}, use prior knowledge collected from the process data including points, function values, gradient values, matrix approximations etc. Thus, the next estimate of local optimum is calculated as follows

$$t_{n+k} = t_n - \Phi(t_0, t_1, \ldots, t_n), \tag{1}$$

where $\Phi(\cdot)$ corresponds to the to process of the repetitive algorithm described in Fig.1.

Practically, we seek for future optima Y_{t_max}, where $max > n$, which is possible to obtain only by possessing information about future values. Based on the past data, a past local minimum, denoted by t_{min} may approximated, using a "sequence" of m past points that starts at the last known data point. This sequence can be shown as $[t_n, t_{k_1}, t_{k_2}, \ldots, t_{k_{m-2}}, t_{min}]$, where $n > k_1 > k_2 > \cdots > k_{m-2} > min$.

When this "sequence" of points is viewed as a forward process, appears as a "sequence" that starts from the minimum past point t_{min}, crisscrosses the last known point t_n and it probably leads to a maximum future one t_{max}. When applying this backtrack technique, the constructed "sequence" of points provides us with all the information needed to proceed in estimating a future maximum. Thus, by applying a maximization technique

$$t_{n+k} = t_n + \Psi\left(t_{k_{m-2}}, \ldots, t_{k_1}, t_n\right) \tag{2}$$

that uses prior information about points and since the "sequence" of points is known, may lead to the most appropriate $\Psi(\cdot)$ that finally converges to a future local maximum. Thus, the previously described process gives the following backtrack algorithm for the estimation of future local maximum:

1. Start with last known value t_n.
2. Use Algorithm described in Fig.1 to compute a sequence of points t_n, t_{k_1}, t_{k_2}, $\ldots$, $t_{k_{m-2}}$, t_{min} leads to "past" local minimum.

3. Calculate "future" point t_{n+k} by applying optimization algorithm described in Fig.1 using points t_{min}, $t_{k_{m-2}}$, ..., t_{k_2}, t_{k_1}, t_n.

The proposed application is tested on the daily closing prices of the Athens' Stock Market. The data consists of the daily closing prices of 18 years - from 1985 until 2002. In Figure 2 is presented an application of the backtrack algorithm towards predicting Athens' Stock Market general index for the randomly chosen date of April 14, 1999. The last 50 known values of general index are used; i.e. in the case of April 14, 1999, the 50 last known indexes are from January 29, 1999 until April 14, 1999. These points are represented on Figure 2 with the square symbol. In Figure 2, again, the gray circles stand for the index's actual values index for the exchanging period from April 15, 1999 till July 14, 1999. Future values are connected together with the discontinuous line. The points depicted from the backtrack algorithm are symbolized with the rhomb symbol.

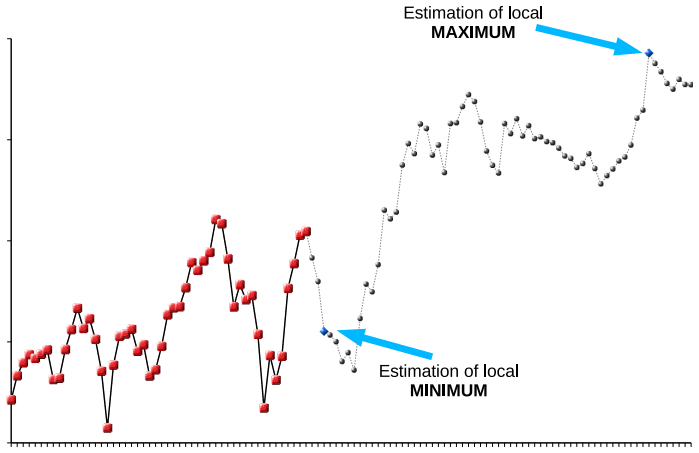


Fig. 2. An application of the backtrack algorithm on Athens Stock Market general index on April 14, 1999

Further research interests and Applications

In this paper we proposed a backtrack technique which enables us to forecast the future optima of a time series.

As mentioned before such application allows forecasts in a depth of time instead of the "next day's value". This appear to be a useful tool in research areas that focus on predicting future optima. These may include not only meteorology, but geology and finance in terms of portfolio optimization.

Further research may be on applying the proposed backtrack technique in the forementioned subject areas and compare the obtained results with other methodologies in order to show its efficiency.

References

1. H. Zou and Y. Yang. Combining time series models for forecasting. *International Journal of Forecasting*, 20:69–84, 2004.
2. H. Markowitz. Portfolio selection. *The Journal of Finance*, 7:77–91, 1952.
3. W.F. Sharpe. *Portfolio Theory and Capital Markets*. McGraw–Hill, New York, 1970.
4. R.C. Merton. An analytic derivation of the efficient frontier. *Journal of Finance and Quantitative Analysis*, 9:1851–1872, 1972.
5. J.S. Pang. A new efficient algorithm for a class of portfolio selection problems. *Operational Research*, 28:754–767, 1980.
6. A.F. Perold. Large-scale portfolio optimization. *Management Science*, 30:1143–1160, 1984.
7. M.J. Best and R.R. Grauer. The efficient set mathematics when mean-variance problems are subject to general linear constrains. *Journal of Economics and Business*, 42:105–120, 1990.
8. M.J. Best and J. Hlouskova. The efficient frontier for bounded assets. *Mathematical Methods of Operations Research*, 52:195–212, 2000.
9. K.N. Levy B.I. Jacobs and H. Markowitz. Portfolio optimization with factors, scenarios, and realistic short positions. *Operations Research*, 53:586–599, 2005.
10. R. G. Brown. *Statistical Forecasting for Inventory Control*. McGraw-Hill, New York, 1959.
11. G. E. P. Box and G. M. Jenkins. *Time Series Analysis: Forecasting and Control*. 2nd Edition, Holden–Day, San Francisco, 1976.
12. C. M. Kuan and T. Lim. Forecasting exchange rates using feedforward and recurrent neural networks. *Journal of Applied Econometrics*, 10:347–364, 1994.
13. Jr. R. A. Taggart. A model of corporate financing decisions. *The Journal of Finance*, 32:1467–1484, 1977.
14. R. C. Merton. On market timing and investment performance. i. an equilibrium theory of value for market forecasts. *The Journal of Business*, 54:363–406, 1981.
15. K.H. Lee and G.S. Jo. Expert system for predicting stock market timing using a candlestick chart. *Expert Systems with Applications*, 16:357–364, 1999.
16. J. Magee R. D. Edwards and W. H. Bassetti. *Technical Analysis of Stock Trends*. AMACOM, Division of the American Management Associations, 2001.
17. N. G. Pavlidis, V. P. Plagianakos, D. K. Tasoulis, and M. N. Vrahatis. Financial forecasting through unsupervised clustering and neural networks. *Operational Research: An International Journal*, 6(2):103–127, 2006.
18. N. G. Pavlidis, D. K. Tasoulis, V. P. Plagianakos, and M. N. Vrahatis. Computational intelligence methods for financial time series modeling. *International Journal of Bifurcation and Chaos*, 16(7):2053 – 2062, 2006.
19. C. Grosan and A. Abraham. *Stock Market Modeling Using Genetic Programming Ensembles*. Springer Berlin / Heidelberg, 2006.
20. Y. Chen, L. Peng, and A. Abraham. Stock index modeling using hierarchical radial basis function networks. *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, 4253 LNAI - III:398–405, 2006.

Discontinuous Galerkin Methods for the linear Schrödinger equation in non-cylindrical domains

D.C. Antonopoulou[1,2]

[1] Department of Applied Mathematics, University of Crete,
GR–714 09 Heraklion, Crete, Greece
danton@tem.uoc.gr
[2] Institute of Applied and Computational Mathematics, FO.R.T.H.,
GR–711 10 Heraklion, Crete, Greece

Abstract. The Discontinuous Galerkin (DG) methods (Reed and Hill 1973 [8]) are finite element methods highly parallelizable, and they handle complicated geometries and adaptivity strategies since refinement can be applied without continuity restrictions. The degree of the polynomial approximation may be changed from one element to the other ([3], [5], [6], [1]). We use Space-Time (S-T) methods which are discontinuous in time (special case of DG) for the linear Schrödinger evolution equation in non-cylindrical domains of R^m with Dirichlet boundary conditions, and prove stability and error estimates in finite element spaces of general type. When $m = 1$ the resulting problem is the standard 'parabolic' approximation of Helmholtz equation as it appears in underwater acoustics, [9], [7].

Introduction

Space-Time methods has been applied by Jamet [5], in evolutionary problems with Dirichlet boundary conditions. Karakashian and Makridakis in [6] used (S-T), discontinuous in time methods, to the non-linear Schrödinger equation with Dirichlet boundary conditions in cylindrical domains in $R^2 \times [0, T]$, and proved error estimations of optimal order in L^2. Akrivis and Makridakis in ([1]), applied these methods in general non-linear parabolic equations and proved existence, uniqueness and optimal a priori error estimates. In the present work, we apply (S-T) methods in the linear Schrödinger equation with Dirichlet boundary conditions in non-cylindrical domains in m dimensions, using finite element spaces of general type. We prove uniqueness of the discrete solution, L^2 stability for the discrete schemes, and L^2 general error estimates. In the case of non-cylindrical domain in two dimensions we transform the domain into a horizontal one and specify the choice of finite element spaces. We discretize in space and time and prove uniqueness of discrete solution, and L^2 stability for the scheme. In the case of non-uniform adaptive mesh we present L^2 error estimates. Karakashian and Makridakis in [6], by use of Lagrange polynomials at Radau points in subintervals of time discretization, under some assumptions for the mesh, proved optimal

error estimates in L^2 for the non-linear Schrödinger equation with constant coefficients. In our case - the case of linear Schrödinger equation in non - cylindrical domains - after the horizontal bottom change of variables, the coefficients become dependent by t, here we present suboptimal L^2 estimates, in t, and optimal L^2 estimates, in space variables.

The problem

Let $0 \leq t \leq T$, and $\Omega(t)$ bounded domain in R^m, continuously depended from $t \in [0,T]$. We consider the boundary $\Gamma(t)$ of $\Omega(t)$ and define

$$S_T = \{(x,t) : \ x \ \in \Omega(t),\ 0 \leq t \leq T\},\ \Sigma_T = \{(x,t) : \ x \in \Gamma(t),\ 0 \leq t \leq T\}. \tag{1}$$

Σ_T is piecewise smooth. Let $a \in R^*$, f, b and u^0 are functions of t and the space variables, $f,\ b \in L^2(S_T),\ u^0 \in L^2(\Omega(0))$, where $\triangle$ is the Laplacian in space variables. We consider the Dirichlet initial and boundary value problem for the linear p.d.e. of Schrödinger type

$$u_t = \mathrm{i}a\triangle u + \mathrm{i}bu + f \text{ in } S_T,\ \ u = 0 \text{ in } \Sigma_T,\ \ u = u^0 \text{ in } \Omega(0). \tag{2}$$

This problem admits a unique solution if f, b, u^0 are smooth enough. We define: $(\cdot,\cdot)_{\Omega(t)}$ =inner product in $L^2(\Omega(t))$, $|\cdot|_{\Omega(t)}$ =norm in $L^2(\Omega(t))$, $((\cdot,\cdot))_G$ =inner product in $L^2(G)$, with G subdomain of S_T, $||.\|_G$=norm in $L^2(G)$. Let $G = \{(x,t) : x \in \Omega(t),\ \tau_0 \leq t \leq \tau_1\}$, $(\cdot,\cdot)_{\Omega(t)} = (\cdot,\cdot)_{L^2(\Omega(t))}$, and $\tilde{H}^1(G) = \{f \in H^1(G) : f(x,t) = 0 \text{ if } (x,t) \in \Sigma_T \cap \overline{G}\}$ is the space of all functions of $H^1(G)$ that equal zero on $\Sigma_T \cap \overline{G}$ - the lateral boundary of G. If ψ and ϕ are smooth, we define the sesquilinear form

$$B_G(\psi,\phi) = -((\psi, \tfrac{\theta\phi}{\theta t}))_G + \mathrm{i}a((\nabla\psi,\nabla\phi))_G - \mathrm{i}((b\psi,\phi))_G + (\psi,\phi)_{\Omega(\tau_1)} - \\ -(\psi,\phi)_{\Omega(\tau_0)}. \tag{3}$$

Discontinuous approximations

Let $0 = t^0 \leq t^1 \leq \cdots \leq t^n \leq t^{n+1} \leq \cdots \leq t^N = T$, $\Omega^n = \Omega(t^n)$, $G^n = G(t^n, t^{n+1})$, $\widetilde{G}^n = \overline{G}^n - \overline{\Omega}^n = \{(x,t) : x \in \overline{\Omega}(t),\ t^n < t \leq t^{n+1}\}$. We consider a subspace V_h^n of $\tilde{H}^1(G^n)$ of finite dimension, for $0 \leq n \leq N-1$, and $V_h = \{v_h$ defined in $\overline{S_T}$: for any n exists $\phi_h \in V_h^n$: $v_h|\widetilde{G}^n \equiv \phi_h|\widetilde{G}^n\}$. The functions of V_h are in general discontinuous in t at the points t^n. We define: $v_h(\cdot,t^n) := v_h^n$, for $0 \leq n \leq N$ ($v_h(\cdot,t^n) = \lim\limits_{\varepsilon\to 0^+} v_h(\cdot,t^n-\varepsilon)$ for $0 \leq n \leq N-1$) and $v_h^{n+0} := \lim\limits_{\varepsilon\to 0^+} v_h(\cdot,t^n+\varepsilon)$ for $0 \leq n \leq N-1$. We will approximate the problem (2) by the discrete analogue of relation (3).

The discrete problem. We seek $u_h \in V_h$ such that $u_h^0 = u^0$ and

$$B_{G^n}(u_h,\phi_h) = ((f,\phi_h))_{G^n},\ \text{for any } \phi_h \in V_h^n \text{ and for any } n,\ 0 \leq n \leq N-1. \tag{4}$$

We define in $V_h \times V_h$ the sesquilinear form $B^n(u_h,v_h) := B_{G^n}(u_h,v_h)$.

Theorem 01 *The discrete problem (4) has a unique solution u_h, which satisfies the estimate*

$$(\min_{x,t} b_I - \frac{1}{2})\|u_h\|^2_{G(0,t^n)} \leq \frac{1}{2}|u^0|^2_{\Omega^0} + \frac{1}{2}\|f\|^2_{G(0,t^n)}. \tag{5}$$

Change of variables for the problem (2)

We consider a bounded domain of R, $(m = 1)$, continuously dependent by $t \in [0,T]$ with $\Gamma(t) = \{x = 0, x = s(t), t \in [0,T]\}$. In problem (2) if $\dot{s}, \ddot{s}$ exist in S_T we apply the transformation $z(y,t) = e^{ct - \mathrm{i}y^2\omega(t)}u(x,t)$ with $y = \frac{x}{s(t)}$ and $\omega(t) = \frac{\dot{s}(t)s(t)}{4a}$ with c a complex constant such that the coefficient of z in equation is bounded, [2]. We define for ψ and ϕ smooth, for $s := s(t)$

$$B_{G_z}(\psi,\phi) = -((s^2(t)\psi, \tfrac{\theta\phi}{\theta t}))_{G_z} + \mathrm{i}a((\nabla\psi, \nabla\phi))_{G_z} - \mathrm{i}(((s^2 b_z - 2\mathrm{i}\dot{s}s)\psi, \phi))_{G_z} + \\ +(s^2\psi,\phi)_{\Omega_z(\tau_1)} - (s^2\psi,\phi)_{\Omega_z(\tau_0)}. \tag{6}$$

There exists a unique solution of the transformed problem and satisfies: if $\phi \in \tilde{H}_1(G_z)$ and if $G_z = G_z(\tau_0,\tau_1)$ then $B_{G_z}(z,\phi) = ((s^2 f_z, \phi))_{G_z}$, with $0 \leq \tau_0 \leq \tau_1 \leq T$. We will define the discrete analogue of previous equation:
Discrete problem. We seek $z_h \in V_h$ such that $z_h^0 = z^0$ and

$$B_{G_z^n}(z_h, \phi_h) = ((s^2 f_z, \phi_h))_{G_z^n}, \text{ for any } \phi_h \in V_{z_h}^n \text{ and any } n,\ 0 \leq n \leq N-1. \tag{7}$$

Theorem 02 *The discrete problem (7) admits a unique solution z_h, which satisfies the estimate*

$$(\min_{x,t}(s^2 b_{zI} - \dot{s}s) - \frac{1}{2})\|z_h\|^2_{G_z(0,t^n)} \leq \frac{1}{2}|s_0 z^0|^2_{\Omega_z^0} + \frac{1}{2}\|s^2 f_z\|^2_{G_z(0,t^n)}. \tag{8}$$

Discontinuous in time (Space-Time) finite elements for the transformed problem

Let $0 = t^0 < t^1 \cdots < t^N = T$ a discretization of $[0,T]$, $k_n := t^{n+1} - t^n$, $I_n = (t^n, t^{n+1}]$. For simplicity we define $\Omega_z =: \Omega$ (Ω is the domain after the change of variables) then $\widetilde{G_z^n} = \overline{\Omega} \times I_n$, $\Omega_z^n = \Omega$, with Ω independent of t. We consider a triangulation T_{hn} of Ω that satisfies [4]: **(H1)** T_{hn} is regular: (i) There is a constant σ such that $\forall k \in \cup_{h_n} T_{hn}$, $\frac{h_k}{\varrho_k} \leq \sigma$, with ϱ_k the largest diameter of spheres in k. (ii) The quantity $h_n = \max_{k \in T_{hn}} h_k$, with $h_k = D(k) =$ diameter of k, is near zero. **(H2)** The finite element family (k, P_k, Σ_k), with $k \in T_{hn}$, is affine for any h. **(H3)** All finite elements (k, P_k, Σ_k), $k \in \cup_{h_n} T_{hn}$ are of class $\mathcal{C}^0$ [4]. We also assume [4] that: **(H4)** There exist integers $r - 1 \geq 0$, $l \geq 0$, $l \leq r-1$ such that $P_{r-1}(\widehat{k}) \subset \widehat{P} \subset H^i(\widehat{k})$, and, $H^r(\widehat{k}) \hookrightarrow \mathcal{C}^s(\widehat{k})$, where s is the largest order of derivatives that appears as degree of freedom

(in set $\widehat{\Sigma}$), where $P_{r-1}(\widehat{k})$ are the polynomials defined in $\widehat{k}$, of order at most $r-1$. Let $S_h^n := \{x \in H_0^1(\Omega) : x| \in P_{r-1}(K), K \in T_{hn}\}$ (of class $\mathcal{C}^0(\widehat{k})$) with $s=0,\ l=1,\ 0 \le m \le 1$ where P_{r-1} is the space of polynomials of order at most $r-1$. The space S_h^n has finite dimension in every I_n and satisfies $(H1)-(H4)$. We define for q positive integer, the space $V_{hk} = V_{hk}(q)$ of piecewise polynomial functions $\phi : \Omega \times (0,T] \to C : \phi|\Omega \times I_n = \sum_{j=0}^{q-1} t^j \tilde{x}_j(y), \tilde{x}_j \in S_h^n$. For any n, let $V_{hk}^n := \{\phi|\Omega \times I_n : \phi \in V_{hk}\}$. The functions of V_{hk} are elements of S_h^n for any $t \in I_n$, and for any $x \in \Omega$ piecewise polynomial functions of order at most $q-1$ possibly discontinuous at points t^n for $n = 0, \ldots, N-1$, [6]. We assume that the solution z of the problem is q times continuously differentiable in t in any $\overline{I_i}$. If $S_h^{i-1} \neq S_h^i$ for finite number of times c_d, independent of N, then we can prove the next Theorem:

Theorem 03 *If z is the solution of the transformed problem, z_h the solution of discrete problem (7) and $v_h \in V_{z_h}$ then for $h := \max\limits_{i \le n-1} h_i, \quad k := \max\limits_{i \le n-1} k_i$ holds that*

$$\begin{aligned} &c\|z - z_h\|_{G_z(0,t^n)} + \tfrac{1}{2}\sum_{i=0}^{n-1} |s_i(z_h^{i+0} - z_h^i)|_{\Omega_z^i} + \tfrac{1}{8}\max_{0 \le j \le n} |s_j(z^j - z_h^j)|_{\Omega_z^j} \le \\ &\le c(1+c_d)\{h^r + k^{q-1}\} + c(N-c_d)(h^r k + k^q) \le c\{h^r + k^{q-1}\}. \end{aligned} \tag{9}$$

We notice that Theorem 03 can be applied when $S_h^n := \{x \in H_0^1(\Omega) : x|K \in P_{r_n-1}(K), K \in T_{hn}\}$ where P_{r_n-1} is the space of polynomials of order at most $r_n - 1$. Theorem 03, in the present case, can be extended using $r = \min\limits_n r_n, \quad q = \min\limits_n q_n$. Consequently, in every $\Omega \times I_n$, one can use different degree of polynomial approximation in time and in space variables.

Acknowledgements

This work was supported by a Pythagoras grant to the University of Athens, co-funded by the Greek Ministry of Education and the E.U. European Social Fund.

References

1. G. Akrivis, C. Makridakis, Galerkin time-stepping methods for nonlinear parabolic equations. ESAIM: Math. Mod. and Num. Anal. **38** (2004) 261–289
2. D. C. Antonopoulou, Theory and Numerical Analysis of Parabolic Approximations, Thesis of Science. University of Athens, Department of Mathematics. (2006)
3. K. S. Bey, J. T. Oden, hp-version discontinuous Galerkin methods for hyperbolic conservation laws. Comput. Methods Appl. Mech. Engrg. **133** (1996) 259–286
4. P. G. Ciarlet, The Finite Element Method for Elliptic Problems. North-Holland, Amsterdam. (1978)
5. P. Jamet, Galerkin-type approximations which are discontinuous in time for parabolic equations in a variable domain. SIAM J. Numer. Anal. **15** (1978) 913–928

6. O. Karakashian, C. Makridakis. A space-time finite element method for the Nonlinear Schrödinger Equation: The Discontinuous Galerkin Method. Math. Comp. **67** (1998) 479–499
7. D. Lee, S. T. McDaniel, Ocean acoustic propagation by finite difference methods. Comput. Math. Appl. **14** (1987) 305–423
8. W. H. Reed, T. R. Hill, Triangular mesh methods for the neutron transport equation. Tech. Report LA-UR-73-479, Los Alamos Scientific Laboratory. (1973)
9. F. D. Tappert, The parabolic approximation method, Wave Propagation and Underwater Acoustics. J.B. Keller and J.S. Papadakis, eds., Lecture Notes in Phys. **70**, Springer-Verlag, Berlin (1977) 224–287

A Paraxial Approach for Electromagnetic PIC Codes in Highly Relativistic Beams

Franck Assous[1] and Felix Tsipis[2]

[1] Department of Comp. Sc. & Maths., College of Judea and Samaria
Ariel 44837, Israel
[2] Department of Maths & Stat., Bar-Ilan University
Ramat-Gan 52900, Israel.

Abstract. Solving the Vlasov-Maxwell problem can lead to very expensive computations. To construct a simpler model, Laval et al. [5] proposed to exploit the *paraxial* property of the charged particle beams, i.e the particles of the beam remain close to an optical axis. They so constructed a paraxial approximation and performed its mathematical analysis. In this contribution, we examine some recent developments: using a paraxial axisymmetric model, and coupling it with the Vlasov equation, one contructs a Particle In Cell (PIC) code, in the case of highly relativistic beams. We provide numerical results to illustrate the efficiency of this approach.

Introduction

Charged particle beams simulations require to develop models appropriate for numerical experiments, such as the Vlasov-Maxwell system of equations (cf. [4]). This model, even if it is unavoidable in many situations [2], [1], leads to very expensive computations. We consider here the transport of a bunch of highly relativistic charged particles in the interior of a perfectly conducting tube. As we are in the same physical assumptions as Laval et. al [5], we start from their approach to investigate a paraxial axisymmetric situation. First, we recall the Vlasov-Maxwell model. Then, by introducing an *ad hoc* scaling, we develop a paraxial Vlasov-Maxwell model, based on asymptotic expansions. Finally, numerical approximations are used to construct a PIC code, from which numerical results illustrate the efficiency of the method.

From Vlasov-Maxwell to a paraxial axisymmetric model

Consider a beam of charged particles with a mass m and a charge q which moves inside a perfectly conducting cylindrical tube, the z-axis being the axis of the tube. Suppose that the beam is confined in a neighborhood of the z-axis, which is chosen as the optical axis of the beam. Each particle can be characterized by its position $\mathbf{x} = (x, y, z)$ and its velocity $\mathbf{v} = (v_x, v_y, v_z)$ in the phase space $(\mathbf{x}, \mathbf{v})$. Assuming that the beam is relativistic and noncollisional, the motion of these

particles can be described in terms of particle distribution function $f(\mathbf{x}, \mathbf{p}, t)$ by the relativistic Vlasov equation

$$\frac{\partial f}{\partial t} + \mathbf{v} \cdot \nabla_x f + \mathbf{F} \cdot \nabla_{\mathbf{p}} f = 0\,, \tag{1}$$

where the momentum $\mathbf{p}$ verifies $\mathbf{p} = \gamma m \mathbf{v}, \quad \gamma = (1 - \frac{\mathbf{v}^2}{c^2})^{-1/2}$. In equation (1), $\mathbf{F}$ denotes the electromagnetic Lorentz force given by

$$\mathbf{F} = q(\mathbf{E} + \mathbf{v} \times \mathbf{B})\,, \tag{2}$$

that describes how an electromagnetic field $\mathbf{E}(\mathbf{x}, t)$ and $\mathbf{B}(\mathbf{x}, t)$ acts on a particle with a velocity $\mathbf{v}$. This electromagnetic field satisfies the Maxwell equations in the vacuum

$$\frac{1}{c^2}\frac{\partial \mathbf{E}}{\partial t} - \nabla \times \mathbf{B} = -\mu_0 \mathbf{J}, \qquad \nabla \cdot \mathbf{E} = \frac{1}{\varepsilon_0}\rho, \tag{3}$$

$$\frac{\partial \mathbf{B}}{\partial t} + \nabla \times \mathbf{E} = 0, \qquad \nabla \cdot \mathbf{B} = 0\,. \tag{4}$$

The charge and the current densities ρ and $\mathbf{J}$ are obtained from the distribution function f

$$\rho = q \int f d\mathbf{p}, \qquad \mathbf{J} = q \int \mathbf{v} f d\mathbf{p}. \tag{5}$$

Following [5], we assume that the beam is highly relativistic i.e., satisfies $\gamma >> 1$. Since $\mathbf{v} \simeq c$ for any particle in the beam, we rewrite the Vlasov-Maxwell equations (1-4) in the beam frame, which moves along the z-axis with the light velocity c. Hence we set $\zeta = ct - v_z, \quad v_\zeta = c - v_z$. Then, to derive a paraxial model, we first introduce a scaling of the equations. As in [5], we assume that the dimensions of the beam are small compared to the longitudinal length of the device, the longitudinal particle velocities v_z are close to the light velocity c, whereas the transverse particle velocities are small compared to c. Hence we can introduce the transverse characteristic velocity of the particles $\overline{v}$, and define a small parameter η, $\eta = \frac{\overline{v}}{c} \ll 1$. We thus obtain (cf [3]) a Vlasov-Maxwell equations expressed in dimensionless variables, where appear powers of the small parameter η. The next step consists in developing asymptotic expansions of all these quantitites ($f, \mathbf{E}, \mathbf{B}, \mathbf{F}$, etc.) in powers of the small parameter η. It is proved in [5] that the resulting new model is an approximation exact up to the order 3 in η. In this paper, we focus on the axisymmetric case, which is well adapted to our problem. Using the coordinates (r, θ, ζ) (with obvious notations), the electric field is now denoted (E_r, E_θ, E_z), the magnetic one (B_r, B_θ, B_z). Hence, the paraxial model of ultrarelativistic Maxwell equations is written

$$E_r = cB_\theta = \frac{1}{\varepsilon_0\, r}\int_0^r \rho s\, ds \text{ and } E_\theta = B_r = 0$$

for the zero-order fields. For the first-order ones, we have

$$\frac{\partial E_z}{\partial r} = \frac{\partial B_\theta}{\partial t} \text{ with } E_{z_{(r=R)}} = 0 \text{ and } \frac{\partial B_z}{\partial r} = \mu_0 J_\theta \text{ with } \int_0^R B_z r dr = 0.$$

Finally, the second-order pseudo-fields $\mathcal{E}_r = E_r - cB_\theta$ and $\mathcal{E}_\theta = E_\theta + cB_r$ verify

$$\mathcal{E}_r = \frac{1}{r}\int_0^r (\mu_0 c J_\zeta - \frac{1}{c}\frac{\partial E_z}{\partial t}) s\, ds \text{ and } \mathcal{E}_\theta = -\frac{1}{r}\int_0^r \frac{\partial B_z}{\partial t} s\, ds.$$

We approximate these equations with specific numerical schemes based on a finite-difference approach. The order of the computations is induced by the asymptotic expansion. Hence, the zero order fields have to be first computed, and are necessary to obtain the first order quantities etc. More details can be found in [3].

Starting from the Vlasov equation (1), we now derive its axisymmetric counterpart. Denoting by $\mathbf{x} = (r, z)$, $\mathbf{p} = (p_r, p_\theta, p_z)$, we assume that the particle distribution function $f(\mathbf{x}, \mathbf{p}, t)$ is independent of θ. According to the particle method, $rf(\mathbf{x}, \mathbf{p}, t)$ is approximated in the phase space $(\mathbf{x}, \mathbf{p})$ by (w_k being the weight of the particle k)

$$rf(\mathbf{x}, \mathbf{p}, t) = \sum_k w_k \delta(\mathbf{x} - \mathbf{x}_k(t))\delta(\mathbf{p} - \mathbf{p}_k(t)), \tag{6}$$

This gives the following approximations for $\mathbf{J}$ and ρ

$$r\rho(\mathbf{x}, t) = q\sum_k w_k \delta(\mathbf{x} - \mathbf{x}_k(t)), \quad r\mathbf{J}(\mathbf{x}, t) = q\sum_k w_k \mathbf{v}_k(t)\delta(\mathbf{x} - \mathbf{x}_k(t)). \tag{7}$$

Using (6), the solution to the Vlasov equation is equivalent to solve for each particle k, the system for the positions and for the momentum respectively

$$\begin{cases} \dfrac{\partial r}{\partial t} = \dfrac{1}{\gamma m} p_r, \\ \dfrac{\partial z}{\partial t} = \dfrac{1}{\gamma m} p_z, \end{cases} \qquad \begin{cases} \dfrac{\partial p_r}{\partial t} = \dfrac{1}{\gamma m r} p_\theta^2 + F_r, \\ \dfrac{\partial p_\theta}{\partial t} = -\dfrac{1}{\gamma m r} p_r p_\theta + F_\theta, \\ \dfrac{\partial p_z}{\partial t} = F_z \end{cases}$$

where the paraxial electromagnetic force $\mathbf{F}$ satisfies

$$\begin{aligned} F_r &= q(\mathcal{E}_r^s + v_\theta^n B_z^s + v_\zeta B_\theta^s), \\ F_\theta &= q(\mathcal{E}_\theta^s - v_r^n B_z^s - v_\zeta B_r^s), \\ F_z &= q(E_z^s + v_r B_\theta^s - v_\theta B_r^s). \end{aligned}$$

Numerical results

As a numerical example, consider a bunch of particles emitted with velocities such that the paraxial assumptions are verified. According to stability condition [4], more than 10 particles are placed in each cell, with the same weight and a charge following $w = \frac{\mathbf{J}\Delta t}{Ne}$, ($\mathbf{J}$ the total current to be emitted, and N the particle number). Figures 1 and 2 show respectively the self-consistent electric radial field E_r, and the charge density ρ obtained after 50 time steps of simulation with the resulting PIC paraxial code.

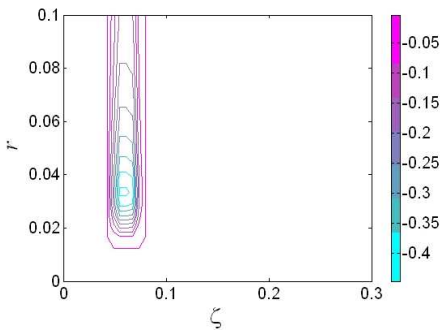


Fig. 1. E_r component (50 Δt).

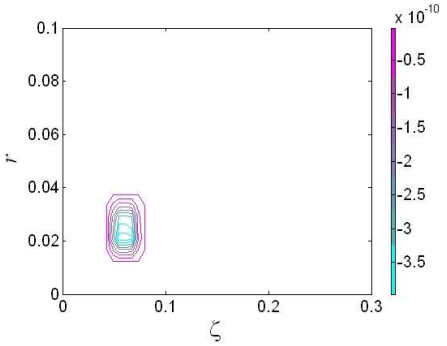


Fig. 2. charge density ρ (150 Δt).

Conclusion

PIC code for highly relativistic beam has been developed. It has been constructed from a paraxial approximation of the Vlasov-Maxwell equations in an axisymmetric geometry. We construct numerical schemes based on finite differences methods for the Maxwell equations, and we develop a well-adapted particle method for the Vlasov equation. Numerical results were presented to illustrate the feasibility and the accuracy of this approach.

References

1. Assous, F. Degond, P., Segre, J.: A particle-tracking method for 3D electromagnetic PIC codes on unstructured meshes. Com. Phys. Comm. Sci. **72** 513-562, (1992).
2. F. Assous, P. Degond, E. Heintzé, P.A. Raviart and J. Segré, On a finite element method for solving the three dimensional Maxwell equations, *J. Comput. Physics*, 109(2), 1993, 222-237.
3. Assous F., Tsipis F.: Numerical paraxial approximation for highly relativistic intense beams. IASTED 2006.
4. C.K. Birdsall and A.B. Langdon, *Plasmas Physics via Computer Simulation* (New York: Mac.Graw-Hill, 1985).
5. Laval, G., Mas-Gallic and S., Raviart, P. A.: Paraxial approximation of ultrarelativistic intense beams. Numer. Math. **69(1)**, 33-60 (1994).

Polynomial filtered Lanczos iterations with applications in electronic structure calculations

C. Bekas[1] and Y. Saad[2]

[1] IBM Research, Zurich Research Laboratory
bek@zurich.ibm.com
[2] Computer Science & Engineering Dept.
University of Minnesota Twin Cities,
saad@cs.umn.edu

Abstract. The most expensive part of all electronic structure calculations based on Density Functional Theory, lies in the computation of an invariant subspace associated with some of the smallest eigenvalues of a discretized Hamiltonian operator. We show that polynomial filtered Lanczos iterations combined with selective reorthogonalization schemes can yield a powerful tool for this demanding problem.

Introduction

Ab initio electronic structure calculations, in the framework of Density Functional Theory (DFT) [3, 4], have proven remarkably accurate in providing a wealth of information concerning several important physical properties of complex materials. However, DFT calculations are extremely demanding and have stretched our computational capabilities to their very limits. Therefore, advances in better computational techniques and algorithms receive much of attention in this very active field of research.

The core problem in DFT calculations is the solution of the Kohn-Sham equations

$$H_\rho \Psi_\rho = E\Psi_\rho, \tag{1}$$

where ρ is the charge density of the electrons distribution, H_ρ is the Kohn-Sham Hamiltonian operator, Ψ_ρ are the wavefunctions and E is the energy of the system. Observe that this is a nonlinear eigenvalue problem, since the Hamiltonian and the wavefunctions depend upon each other through the charge density ρ. The last decades have seen many methods that attempt to efficiently solve equation (1). Many of these methods utilize some sort of iteration which aims at improving some initially selected wavefunctions so that at the end of the iteration the approximate energy E is as small as possible, or in other words the solution of equation (1) is self-consistent. The computational complexity of practical algorithms for this problem stems mainly from two factors:

Discretization The Hamiltonian operator and thus the wavefunctions have to be discretized in some suitably selected basis. This typically results in a very large discretized Hamiltonian (possibly in the order of millions of degrees of freedom). Consequently, standard eigensolvers for dense matrices, such as the ones in `LAPACK`[3], are ruled out in this case. Thus, iterative techniques must be utilized instead, which only require applications of the discretized Hamiltonian operator on a likewise discretized function (i.e. vector).

Orthogonality The wavefunctions Ψ_ρ are orthogonal among each other. Thus, the approximate discretized wavefunctions must likewise form an orthogonal basis. This has the consequence that all iterative methods that aim to solve the nonlinear eigenproblem (1) must maintain orthogonality among the approximate wavefunctions at each iteration. Maintaining orthogonality will of course come at a heavy cost, as it scales as the cube of the total number of valence electrons in the system.

Large scale eigenvalue calculations in DFT

Let us consider a molecular system with N valence electrons. Furthermore, let $\psi_i, i = 1, \ldots, N$ be the eigenvectors associated with the N smallest eigenvalues of the discretized Hamiltonian H. Then, the charge density $\rho(j)$ at the $j-$th point in the discretized domain space is given as:

$$\rho(j) = \sum_{i=1}^{N} |\psi_i(j)|^2. \tag{2}$$

The Hamiltonian H is a symmetric (or Hermitian) very large matrix that is either sparse if the wavefunctions are directly discretized in real space[4] or it allows a fast matrix-vector multiplication, by means of the `FFT` transform, should the wavefunctions be discretized in Fourier space[5].

Thus, it is natural to consider the Lanczos iteration in order to approximate eigenvectors that correspond to the N algebraically smallest (leftmost) eigenvalues of H. There are two main problems in successfully deploying Lanczos for this task.

Convergence The number of required eigenvectors is equal to the number of valence electrons N, which grows very quickly especially for molecular systems with many atoms. Thus, while very small simulations already involve roughly hundreds of valence electrons, several thousands are typically involved for medium to large scale simulations. However, it is well known that even though Lanczos converges quickly for exterior eigenvalues it requires many more iterations to

[3] `http://www.netlib.org/lapack`
[4] see for example `http://www.ices.utexas.edu/parsec/`
[5] See for example `http://www.cpmd.org`

approximate eigenvalues that reach deep into the interior of the spectrum. In particular, consider the Lanczos factorization

$$HV_m = V_m T_m + \beta_{m+1} v_{m+1} e_m^\top, \tag{3}$$

where V_m is an orthonormal basis for the Krylov subspace

$$\mathcal{K}_m(H, v_1) = \text{span}\{v_1, Av_1, A^2 v_1, \ldots, A^{m-1} v_1\},$$

and T_m is a symmetric tridiagonal matrix that is the restriction of the Hamiltonian H on this subspace. The basis V_m and matrix T_m are calculated by means of the Lanczos algorithm that relies on a three term recurrence among the columns of V_m starting from a unit norm vector v_1. At each step k, the last vector v_k is multiplied by the matrix H and the result is orthogonalized against all previous basis vectors. The special structure of the problem allows the Lanczos algorithm to achieve this orthogonalization with a three term recurrence. The eigenvalues of the Hamiltonian are approximated by those of the matrix T_m. It is easy to see that if the starting vector v_1 had no components towards the direction of unwanted eigenvectors (these would be the ones that correspond to the largest eigenvalues in our case), then the wanted eigenvalues would converge faster. However, round-off will always put components of the largest eigenvectors into the Lanczos basis. To see this consider the expansion of the $A^{m-1}v_1$ onto the eigenvectors u_i of matrix H:

$$A^{m-1} v_1 = A^{m-1} \sum_i \xi_i u_i = \lambda_i^{m-1} \xi_i u_i, \tag{4}$$

where λ_i are the eigenvalues of the Hamiltonian H. Thus, even very small components ξ_i towards the largest eigenvalues can be greatly amplified (by the factor λ_i^{m-1}).

In [1] we propose to work with a filtered Hamiltonian $\mathcal{H} = p(H)$, where p is a carefully selected polynomial. Following the approach in [6] we can employ a variant of the Conjugate Residual iteration (see [5]) that is selected to keep the degree of the polynomial p as small as possible (i.e. smaller than 10). It is key to observe that the eigenvectors of $p(H)$ will be identical to the eigenvectors of the Hamiltonian H. This also reflects the property that the charge density ρ is invariant under orthogonal transformation of the wavefunctions. The polynomial $p(H)$ is applied in order to map the largest eigenvalues of H to 0 while mapping the desired leftmost eigenvalues of H to 1. Thus, the effect of round-off is minimized. Instead of multiplying the vector v_k with the Hamiltonian at each step of the Lanczos algorithm we perform the operation $p(H)v_k$. This induces an additional cost of d matrix-vector products, where d is the degree of the polynomial p. However, we have shown that this additional cost is offset by the gains obtained from requiring a much smaller Lanczos basis V_m (that is a smaller m) in order for the eigenvalues deep into the spectrum converge.

Loss of orthogonality A second important advantage of a smaller Lanczos basis V_m is the reduced computational cost for reorthogonalization of the Lanczos vectors. Although in exact arithmetic the Lanczos algorithm is able to keep

Table 1. Performance comparison of Filtered Lanczos, Lanczos with partial reorthogonalization and the popular ARPACK package for $N = 248,350,496$ eigenvalues of a $Ge_{99}H_{100}$ cluster. MV is the total number of matvecs, RTH is the number of reorthogonalizations, RES is the number of restarts, MEM is the required memory (MBytes) and t is time in seconds (on a single MIPS R10000 cpu).

	F. Lanczos				Partial Lanczos				ARPACK			
N	MV	RTH	MEM	t	MV	RTH	MEM	t	MV	RES	MEM	t
248	5194 (550)	102	396	2379	3150	109	2268	2746	3342	20	357	16454
350	8794 (950)	178	684	4648	4570	184	3289	5982	5283	24	504	37371
496	12934 (1410)	270	1015	8374	6550	302	4715	13714	6836	22	714	67020

the columns of V_m orthonormal by a simple three term recurrence, in practice the basis vectors quickly loose orthogonality. This fact requires the Lanczos vectors to be reorthogonalized by means of a Gram-Schmidt process. In the most common practice, when the next vector v_{k+1} is calculated at step k, it is reorthogonalized against all previous vectors. This will induce a cubic cost m^3 relative to the length m of the Lanczos basis, thus a shorter basis significantly reduces the overall cost. An important improvement is selective (or partial) reorthogonalization [7], in which the current vector v_{k+1} is reorthogonalized only when it is needed. It is possible to monitor the loss of orthogonality by means of a simple three term recurrence. In [2] it was shown that such a technique can be successfully adapted for the calculation of charge density with significant gains over the standard approach.

Conclusions The filtered Lanczos iteration and the efficient selective reorthogonalization techniques have been the basic ingredients for creating a highly efficient method (which we denote by F. Lanczos in Table 1) for calculating charge densities in DFT calculations.

References

1. C. Bekas, E. Kokiopoulou, and Y. Saad. Computation of large invariant subspaces using polynomial filtered lanczos iterations with applications in density functional theory. *Under submission*, 2006.
2. C. Bekas, Y. Saad, M. L. Tiago, and J. R. Chelikowsky. Computing charge densities with partially reorthogonalized Lanczos. *Comp. Ph. Com.*, 171(3):175–186, 2005.
3. P. Hohenberg and W. Kohn. Inomogeneous electron gas. *Phys. Rev.*, 136(3B):B864–B871, 1965.
4. W. Kohn and L. J. Sham. Self-consistent equations including exchange and correlation effects. *Phys. Rev.*, 140(4A):A1133–A1138, 1965.
5. Y. Saad. *Iterative Methods for Sparse Linear Systems*. SIAM, Philadelphia, 2003.
6. Y. Saad. Filtered conjugate residual-type algorithms with applications. *SIAM J. Mat. Anal. Appl.*, 28:845–870, 2006.
7. H. D. Simon. Analysis of the symmetric lanczos algorithm with reorthogonalization methods. *Linear Algebra Appl.*, 61:101–132, 1984.

Semi-Lagrangian semi-implicit time-splitting scheme for a regional model of the atmosphere

Andrei Bourchtein and Ludmila Bourchtein

Institute of Physics and Mathematics, Pelotas State University, Brazil
burstein@terra.com.br

Abstract. Semi-Lagrangian semi-implicit (SLSI) method is currently one of the most popular approaches for numerical solution of the atmosphere dynamics equations. In this research we apply splitting techniques in the context of a two-time-level SLSI scheme in order to simplify the treatment of the slow physical modes and optimize the solution of the elliptic equations related to implicit part of the scheme. The performed numerical experiments show the accuracy and computational efficiency of the scheme.

Introduction

Semi-Lagrangian semi-implicit approach is currently one of the most efficient methods for numerical solution of the hydrothermodynamic equations used in weather prediction and atmospheric modeling . Since the proof of its extended stability and accuracy in the 80's, the SLSI method is being used in an increasing range of atmospheric models aimed for atmospheric motions of different space and time scales [2, 3, 7, 8]. In the last years, two-time-level versions of this method are being used in different atmospheric centers, because they allows to choose larger time steps than three-time-level ones and achieve the same accuracy almost doubling the efficiency [1, 3, 5, 6].

Although SLSI schemes have been shown to be quite efficient, there are some computationally expensive parts of calculations, which can be treated in a more optimal manner. In this research we consider an alternative computation for slow gravity modes and solution of 3D elliptic problems for implicitly treated linear terms. For explicit and simple approximation of slow gravity waves, the SLSI algorithm is split into two successive steps: in the first step, all terms are treated explicitly, and in the second step, an implicit time discretization for the fastest waves and more accurate approximation for the most energy valuable terms are introduced. The first step requires simple computations for the entire spectrum of processes described by primitive equations, but the stability criterion is very restrictive. The second step improves the overall stability at a reduced computational cost, because only the terms responsible for the fastest processes are involved in these calculations. This kind of splitting is achieved by vertical decoupling, which transforms the linearized hydrostatic equations to a set of decoupled barotropic modes with different equivalent depths. In this way, 3D

elliptic problem is reduced to a set of 2D systems and only a few of these systems related to the fastest barotropic modes should be solved in order to ensure a large time step. Performed evaluations of computational efficiency and forecast accuracy confirm validity of applied techniques of scale separation and show reduced computational cost of the constructed scheme.

Time discretization

To introduce the SLSI splitting scheme design, let us consider the linearized prognostic equations of the hydrostatic atmosphere on a tangent f-plane [3,4]:

$$\frac{du}{dt} = fv - p_x, \frac{dv}{dt} = -fu - p_y, L\frac{dp}{dt} = \frac{R^2T_0}{c_p}(u_x + v_y). \quad (1)$$

Here t, x, y, σ are independent variables, representing the time, horizontal Cartesian coordinates and vertical pressure coordinate, respectively, and u, v, p are the unknown functions, representing the horizontal velocity components and generalized pressure function, respectively. The last is defined as $p = gz + RT_0 \ln p_s$, where z is the height of a pressure surface and p_s is the pressure at the Earth surface. Besides, the following thermodynamic parameters are used: g is the gravitational acceleration, R is the gas constant of dry air, $T_0 = const$ is the reference temperature profile, c_p is the specific heat at constant pressure, and f is the mean value of the Coriolis parameter. The operator notation is quite standard: the subscripts t, x, y, σ denote the partial derivatives with respect to indicated variable and the symbols $\frac{d}{dt}$ and L denote the individual 3D derivative induced by constant advection vector (a, b, c) and vertical coupling operator, respectively:

$$\frac{d\phi}{dt} = \phi_t + a\phi_x + b\phi_y + c\phi_\sigma, \quad L\phi = (\sigma^2\phi_\sigma)_\sigma. \quad (2)$$

Semi-Lagrangian forward-backward time approximation of equations (1) can be written in the form:

$$\frac{\tilde{u}^\tau - u}{\tau} = f\frac{\tilde{v}^\tau + v}{2} - \tilde{p}^\tau_x, \quad \frac{\tilde{v}^\tau - v}{\tau} = -f\frac{\tilde{u}^\tau + u}{2} - \tilde{p}^\tau_y, \quad L\frac{\tilde{p}^\tau - p}{\tau} = \frac{R^2T_0}{c_p}(u_x + v_y). \quad (3)$$

Here τ is the time step, the functions ϕ^τ denote the values at the arrival point of the 3D trajectory at the new time level $t_{n+1} = (n+1)\tau$, and ϕ denote the values at the departure point at the current time level $t_n = n\tau$. For the linearized model, the trajectory equations of air particles $dx/dt = a$, $dy/dt = b$, $d\sigma/dt = c$, $t \in [t_n, t_{n+1}]$ are readily solved for the coordinates of the arrival points assuming that the departure points are chosen to be the points of a spatial grid.

The forward-backward time step is computationally very simple, because the formulas (3) are actually explicit in time. However, the stability criterion is very rigid: $\tau \le \sqrt{2}h_g/c_g$, where h_g is the mesh size of spatial grid used for gravity terms and $c_g = 350m/s$ is the maximum velocity of gravity waves in the system (1). For example, on staggered grid C with the main mesh size $h = 50km$, the

minimum gravity mesh size is $h_g = h/2 = 25km$, and the maximum allowable time step is about $\tau = 1.5min$, which is too small as compared with accuracy requirement of $\tau_{phys} = 60min$ defined by principal synoptic atmospheric motions.

The spectral analysis of the primitive equations (1) shows that they support both fast and slow gravity waves as well the slow synoptic processes which carry the principal part of the atmosphere energy [3, 4]. The separation of the fast and slow gravity waves can be achieved by vertical decoupling using the first eigenfunctions of the operator L:

$$\phi = \sum \Phi_i \psi_i, \quad L\psi_i = \lambda_i \psi_i, \quad i = 1, ..., n. \tag{4}$$

Then the equations (3) can be written separately for amplitudes U_i, V_i, P_i of a single vertical mode (from now on the index i is omitted for simplicity):

$$\frac{\tilde{U}^\tau - U}{\tau} = f\frac{\tilde{V}^\tau + V}{2} - \tilde{P}^\tau_x, \frac{\tilde{V}^\tau - V}{\tau} = -f\frac{\tilde{U}^\tau + U}{2} - \tilde{P}^\tau_y, \frac{\tilde{P}^\tau - P}{\tau} = -c_g^2(U_x + V_y), \tag{5}$$

where $c_g = \sqrt{-\frac{R^2 T_0}{\lambda c_p}}$ is the gravity wave velocity for individual vertical mode. These velocities decrease with i and have the first characteristic values $c_{g1} = 350m/s, c_{g2} = 210m/s, c_{g3} = 120m/s, c_{g4} = 80m/s, c_{g5} = 55m/s$ in such a way that only the first four-five vertical modes contain the fast gravity waves. To improve stability of the scheme (3), only the fast gravity waves are corrected using the following equations:

$$\frac{U^\tau - \tilde{U}^\tau}{\tau} = f\frac{V^\tau - \tilde{V}^\tau}{2} - \frac{P^\tau_x - 2\tilde{P}^\tau_x + P_x}{2}, \frac{V^\tau - \tilde{V}^\tau}{\tau} = -f\frac{U^\tau - \tilde{U}^\tau}{2} - \frac{P^\tau_y - 2\tilde{P}^\tau_y + P_y}{2},$$

$$\frac{P^\tau - \tilde{P}^\tau}{\tau} = -c_g^2\frac{(U^\tau_x + V^\tau_y) - (U_x + V_y)}{2}. \tag{6}$$

The last system is reduced to 2D Helmholtz equation for the pressure amplitude:

$$P^\tau_{xx} + P^\tau_{yy} - \frac{\tau^2 c_g^2}{4 + \tau^2 f^2} P^\tau = F, \tag{7}$$

with F containing the previously found functions, and it is solved by efficient multigrid method. For specified P^τ, respective velocity amplitudes U^τ, V^τ are found from the first two explicit formulas in (6). After corrections for the fast vertical modes, the physical functions u^τ, v^τ, p^τ are found using formulas (4).

Numerical experiments

To evaluate efficiency and accuracy of the developed two-time-level SLSI splitting scheme (SLSIS in Table 1) applied to nonlinear equations of the hydrostatic atmosphere, 24-h forecasts were calculated and compared with those obtained by forward-backward scheme (SLFB in Table 1) and standard non-splitting SLSI scheme (SLSI in Table 1).

The computational domain consisting of 100x100 uniform horizontal grid with mesh size $h = 50km$ and 30 vertical levels was centered at Porto Alegre city $(30^0S, 52^0W)$. The initial and boundary conditions were obtained from objective analysis and global forecasts of National Centers for Environmental Prediction (NCEP). The stability analysis showed that the constructed scheme allows the use of time steps up to one hour that is close to physical requirement on accuracy of the numerical solution.

Table 1 shows two standard measures of the forecast skill for the geopotential height at the surface level of 500hPa: the root-mean-square (RMS) differences between 24-h forecasts and analysis (in meters), and the correlation coefficient (COR) between predicted and observed tendencies (non-dimensional). In the last line, the used time step τ in minutes (chosen in accordance with stability condition) and CPU time in the fraction of one SLFB forecast run are presented.

Table 1. The RMS, COR, time step τ and CPU time for three semi-implicit schemes

scheme	SLFB	SLSI	SLSIS
RMS	23	21	21
COR	0.91	0.92	0.92
τ / CPU	1.5 / 1	60 / 0.27	60 / 0.19

These evaluations show the efficiency and accuracy of the constructed two-time-level SLSI splitting scheme.

References

1. Benoit R., Desgagne M., Pellerin P., Chartier Y., Desjardins S.: The Canadian MC2: a semi-Lagrangian, semi-implicit wideband atmospheric model suited for finescale process studies and simulation. Mon. Wea. Rev. **125** (1997) 2382-2415.
2. Bourchtein A., Bourchtein L.: Semi-Lagrangian semi-implicit time-splitting scheme for the shallow water equations. Int. J. Numer. Meth. Fluids **54** (2007) 453-471.
3. Durran D.: Numerical Methods for Wave Equations in Geophysical Fluid Dynamics, Springer-Verlag, New York (1999).
4. Holton J.R.: An Introduction to Dynamic Meteorology, Academic Press, San Diego (2004).
5. Hortal M.: The development and testing of a new two-time-level semi-Lagrangian scheme (SETTLS) in the ECMWF forecast model. Q. J. R. Meteor. Soc. **128** (2002) 1671-1687.
6. Kiehl J.T., Hack J.J., Bonan G.B., Boville B.A., Williamson D.L., Rasch P.J.: The National Center for Atmospheric Research Community Climate Model: CCM3. J. Climate **11** (1998) 1131-1149.
7. Lauritzen P.H., Kaas E., Machenhauer B.: A mass-conservative semi-implicit semi-Lagrangian limited-area shallow-water model on the sphere. Mon. Wea. Rev. **134** (2006) 1205-1221.
8. Rosatti G., Cesari D., Bonaventura L.: Semi-implicit semi-Lagrangian modelling for environmental problems on staggered Cartesian grids with cut cells. J. Comp. Phys. **204** (2005) 353-377.

Comparision of different spatial grids for numerical schemes of geophysical fluid dynamics

Ludmila Bourchtein and Andrei Bourchtein

Institute of Physics and Mathematics, Pelotas State University, Brazil
bourchtein@terra.com.br

Abstract. Different processes of the atmosphere and ocean dynamics are of a large scale, which implies a formulation of the respective mathematical models in spherical coordinates to take into account the earth curvature. Due to complexity of these mathematical models only a few particular analytical solutions are known with a little application to practice. Therefore, approximate solutions are found by applying numerical methods. When high accuracy solutions are need in large areas, the most uniform representation of spherical domains is required because that ensures the highest accuracy and stability of numerical schemes. In such cases, the original geographical coordinates are not suitable for construction of computational grids because of strong variation of mesh size with latitude and the pole singularity. The most used approach for planar representation of spherical domains is the conformal mapping, including classical stereographic, conic and cylindrical projections. The important properties of the conformal mappings as applied to grid generation for the models of geophysical fluid dynamics are conservation of a simpler form of the primitive partial differential equations, locally isotropic treatment of derivatives and smoothness of the variation of physical mesh size.
In our recent studies we have solved some variational problems of conformal mappings of spherical domains. In particular, for circular spherical domains we have found the most uniform conformal mappings and evaluated advantages of its employment comparing with traditional mappings. In this study, we design computational grids based on different conformal mappings for rectangular computational domains. The employed conformal mappings include the tangent and secant stereographic, conic and cylindrical projections both polar and oblique. Generated grids are used in the context of the numerical schemes for forecasting the actual atmospheric fields of pressure and wind over the chosen spherical domains of different extent. The stability and accuracy properties of the explicit and semi-implicit schemes are evaluated and obtained numerical results are compared with the analytical evaluations. It is found that the computational grids based on the analytically found most uniform grids for circular spherical domains ensure the most accurate and stable scheme for square or nearly square computational domains of different extent.

Deterministic and randomized column selection algorithms for matrices

Christos Boutsidis and Petros Drineas

Computer Science Department, Rensselaer Polytechnic Institute, Troy, NY, 12180, USA,

Abstract. Given a matrix $A \in R^{m \times n}$ ($m \geq n$) and an integer k ($k \ll n$) we discuss deterministic and randomized algorithms for selecting the k "most linearly independent" columns of A. After summarizing previous deterministic and randomized algorithms for this task, we present a hybrid approach. First, we employ a randomized algorithm presented in [7] to select $c = O(k \log k)$ columns of A and then we employ the deterministic algorithm of [9] to pick exactly k columns from the c columns that were kept after the first step. We provide novel provable bounds for the singular values of the matrix containing the selected columns.

Introduction

Given an $m \times n$ ($m \geq n$) matrix A we study the problem of selecting the $k \ll n$ "most linearly independent" columns of A. Formally, given $A \in R^{m \times n}$ and an integer k ($k \ll n$), we select k columns from A and form the matrix $C \in R^{m \times k}$ such that:

- $\sigma_j(C)$, $j = 1 \ldots k$ is maximized ($\sigma_j(C)$ is the j-th singular value of C),
- $|det(C)|$ is maximized ($det(C)$ is the determinant of C),
- $vol(C)$ is maximized ($vol(C)$ is the volume spanned by the columns of C).

Intuitively, we seek the k columns of A forming a matrix $C \in R^{m \times k}$ such as C is "as non-singular as possible". Prior results in [9, 11, 6, 10], suggest that all the above objectives are (approximately) equivalent. Additionally, identifying the "most linearly independent" columns of a matrix leads to the computation of the numerical rank of the matrix [1], the solution of rank deficient least squares problems [2], and the construction of a near-optimal low rank matrix approximation [3]. See [9, 5] and references therein for further motivation on this problem.

Notation: For the remainder of the paper $A \in R^{m \times n}$ is the input matrix and $k \ll n$ denotes the number of columns that will be chosen. Given $X \in R^{m \times n}$, $X^+ \in R^{n \times m}$ denotes the pseudoinverse of X and $V_k \in R^{n \times k}$ the matrix whose columns are the top k right singular vectors of X.

Deterministic column selection

Golub in [8] presented the first algorithm (called pivoted QR factorization) to address the aforementioned objectives. It is a deterministic, greedy approach that works as follows: given a set of already selected columns, the algorithm chooses as next column the one that is farthest away in the Euclidean norm from the subspace spanned by the already chosen columns. There are no provable bounds for this algorithm; however, it works well in practise. The first algorithm that provided provable bounds for the singular values of $C \in R^{m \times k}$ was proposed by Chan in [1]. It guarantees that

$$\sigma_k(A) \geq \sigma_k(C) \geq \frac{\sigma_k(A)}{\sqrt{n(n-k+1)2^{n-k}}}. \tag{1}$$

The left bound is trivial, unlike the other one, which is useful only if k is very close to n. For smaller values of k the theoretical bound is very weak; however, the algorithm works quite well in practise provided a sharper bound than the theoretical one. For extensions of this algorithm with the same bounds see [3, 4].

Hong and Pan [10] provided the first existence proof of today's state-of-the-art bounds (see eqn. (2)). Chandrasekaran and Ipsen in [5] and then Pan and Tang in [12] presented (different) efficient algorithms that achieve these bounds. Gu and Eisenstat in [9] presented an existence proof and the first algorithm that achieves the bound of eqn. (2) for all top k singular values of C and the corresponding k top singular values of A:

$$\sigma_j(A) \geq \sigma_j(C) \geq \frac{\sigma_j(A)}{\sqrt{1+k(n-k)}}, \tag{2}$$

for $j = 1 \ldots k$. See [6] for yet another algorithm that achieves the same bound. The complexity of all these algorithms is $O(mn^2)$.

Randomized column selection

In [7] Drineas et al presented a randomized algorithm for selecting columns from matrices. The goal there was to construct efficient, column-based, low rank matrix approximation, i.e., to choose columns to form a matrix C that minimizes the error

$$\left\|A - CC^{+}A)\right\|_F. \tag{3}$$

Given the rank parameter k, an error parameter $\epsilon \in (0, 1]$, and a failure probability δ, [7] chooses $c = O(\frac{k \log k \log \frac{1}{\delta}}{\epsilon^2})$ columns of A to form the $m \times c$ matrix C such as with probability at least $1 - \delta$

$$\left\|A - CC^{+}A)\right\|_F \leq (1+\epsilon) \left\|A - A_k\right\|_F. \tag{4}$$

In this work we investigate whether a carefully rescaled version of the matrix $C \in R^{m \times c}$ satisfies bounds similar to the ones in eqn. (2) and we prove the following lemma.

Lemma 1. *Given matrix A ($m \geq n$), a rank parameter $k \ll n$, an error parameter $\epsilon \in (0,1]$, and a failure probability δ, form the matrix $C \in R^{m \times c}$ ($c = O(\frac{k \log k \log \frac{1}{\delta}}{\epsilon^2})$) as described in [7] and rescale it appropriately. Then, with probability at least $1-\delta$, for all $j = 1 \ldots k$,*

$$\epsilon(n-k)^{\frac{1}{4}}\sigma_{k+1}(A) + (1+\epsilon)\sigma_j(A) \geq \sigma_j(CD) \geq (1-\epsilon)\sigma_j(A). \quad (5)$$

Here D is a $c \times c$ diagonal rescaling matrix.

A hybrid approach

Table 1. Our hybrid approach

Input: Matrix $A \in R^{m \times n}$, integer $k \ll n$, error parameter $\epsilon \in (0,1]$, failure probability δ

Output: Matrix $C \in R^{m \times k}$ with columns of A, diagonal $k \times k$ rescaling matrix D.

Randomized step: randomly select $c = O(\frac{k \log k \log \frac{1}{\delta}}{\epsilon^2})$ columns of A and form $T \in R^{m \times c}$.

for $j = 1 : n$

(a) assign a score at the j-th column of A: $p_j = \frac{\|V_k(j,:)\|^2}{k}$

(b) select the j-th column with probability $\min(1, cp_j)$

(c) if the j-th column is chosen, keep the rescaling factor $\sqrt{\frac{1}{\min\{1,cp_j\}}}$

end

Deterministic step: deterministically select exactly k columns from T and form C using the algorithm of [9]. Assign the corresponding rescaling factors to the diagonal of D

Table 1 presents our approach to select the k "most linearly independent" columns from a matrix A. The output of the algorithm is a set of k rescaled columns of A. (Notice that rescaling the selected columns does not change the subspace spanned by these columns, and that in problems such as the subset selection and the low rank matrix approximation [3], it is in principle feasible to rescale the columns without any affecting the final output.) Our main result is summarized in the following theorem.

Theorem 1. *Given a matrix $A \in R^{m \times n}$ ($m \geq n$) and an integer k ($k \ll n$), we can form an $m \times k$ matrix C and a diagonal $k \times k$ rescaling matrix D using the hybrid algorithm of Table 1 such that with probability at least $1-\delta$, for all $j = 1 \ldots k$,*

$$\epsilon(n-k)^{\frac{1}{4}}\sigma_{k+1}(A) + (1+\epsilon)\sigma_j(A) \geq \sigma_j(CD) \geq \frac{(1-\epsilon)\sigma_j(A)}{\sqrt{k(c-k)+1}}. \quad (6)$$

The running time of our hybrid approach is $O(mn^2)$.

References

1. Chan, T. F. Rank revealing QR factorizations. *Linear Algebra and Its Applications 88/89* (1987), 67–82.
2. Chan, T. F., and Hansen, P. C. Computing truncated SVD least squares solutions by rank revealing QR factorizations. *SIAM J. Sci. Statist. Comput. 11* (1990), 519–530.
3. Chan, T. F., and Hansen, P. C. Some applications of the rank revealing QR factorization. *SIAM Journal on Scientific and Statistical Computing 13* (1992), 727–741.
4. Chan, T. F., and Hansen, P. C. Low-rank revealing QR factorizations. *Numerical Linear Algebra with Applications 1* (1994), 33–44.
5. Chandrasekaran, S., and Ipsen, I. C. F. On rank-revealing factorizations. *SIAM J. Matrix Anal. Appl. 15* (1994), 592–622.
6. de Hooga, F., and Mattheijb, R. Subset selection for matrices. *Linear Algebra Appl. 422* (2007), 349–359.
7. Drineas, P., Mahoney, M. W., and Muthukrishnan, S. Subspace sampling and relative-error matrix approximation: Column-based methods. In *APPROX-RANDOM* (2006), pp. 316–326.
8. Golub, G. H. Numerical methods for solving linear least squares problems. *Numer. Math. 7* (1965), 206–216.
9. Gu, M., and Eisenstat, S. C. Efficient algorithms for computing a strong rank-revealing QR-factorization. *SIAM Journal on Scientific Computing 17* (1996), 848–869.
10. Hong, Y. P., and Pan, C. T. Rank-revealing QR factorizations and the singular value decomposition. *Mathematics of Computation 58* (1992), 213–232.
11. Pan, C. T. On the existence and computation of rank-revealing LU factorizations. *Linear Algebra Appl. 316* (2000), 199–222.
12. Pan, C. T., and Tang, P. T. P. Bounds on singular values revealed by QR factorizations. *BIT Numerical Mathematics 39* (1999), 740–756.

Spline collocation for parabolic partial differential equations

Christina C. Christara

Department of Computer Science, University of Toronto
Toronto, Ontario M5S 3G4, Canada
ccc@cs.utoronto.ca
Collaborators: Tong Chen, Dang Duy Minh

Introduction

Collocation is a simple to implement discretization technique for differential equations' problems that gives an approximation to the solution over the whole domain of the problem. Spline collocation has been primarily used for the solution of Boundary Value Problems (BVPs) for ODEs or PDEs, and shown to be an effective method. Two types of spline collocation methods have been developed, the extrapolated (a.k.a. modified) and the deferred-correction methods, both giving rise to optimal order of convergence of the approximation to the solution. These methods have been extended recently to non-uniform grids and integrated with adaptive techniques.

In this paper, we consider quadratic and cubic spline collocation formulated as deferred-correction methods for the space discretization of parabolic PDEs. We focus primarily on the case of one space dimension, but several of the results in this paper can be naturally extended in two or more dimensions. The time discretization is handled by finite difference techniques. Deferred-correction methods normally require the solution of two linear systems per timestep. We discuss ways to avoid the solution of the second linear system, thus improving the efficiency of the methods, without sacrificing accuracy and stability. We incorporate adaptive techniques in the space dimension, while maintaining the efficient formulation of the methods. We present results that demonstrate the stability properties of the resulting methods. We also apply the adaptive spline collocation methods to the American option pricing problem, formulated as a free-boundary PDE problem. The numerical experiments indicate that the proposed methods correctly capture the behaviour of the problem, including the discontinuities due to the initial conditions and the free boundary.

Problem and Discretization

We are concerned with the numerical solution of the parabolic PDE problem

$$u_t - \mathbf{L}u = g \quad \text{in} \quad \Omega \times (O,T) \tag{1}$$

$$\mathbf{B}u = \gamma \quad \text{on} \quad \partial\Omega \times (0,T) \tag{2}$$

$$u = g_0 \quad \text{on} \quad \partial\Omega \cup \Omega \tag{3}$$

where $\mathbf{L}u(x) \equiv p(x)u''(x)+q(x)u'(x)+r(x)u(x)$ is a linear second-order operator, $\mathbf{B}u(x) \equiv \alpha(x)u(x)+\beta(x)u'(x)$ a boundary operator, $p, q, r, g, \alpha, \beta, \gamma, g_0$ are given, $\Omega \equiv (\omega_1, \omega_2)$ is the spatial domain, $(0, T)$ with $T > 0$ is the time domain, and $u(x)$ is the unknown function.

We apply a standard Finite Difference (FD) discretization of (1) in the time dimension

$$(\mathbf{I}-\lambda h_t\mathbf{L})U^j(x) = (\mathbf{I}+(1-\lambda)h_t\mathbf{L})U^{j-1}(x)+h_t(\lambda g^j(x)+(1-\lambda)g^{j-1}(x)),\ x \in \Omega, \tag{4}$$

where $U^j(x) \equiv U(x, t_j)$ is the approximation to $u(x, t_j)$ at the jth timestep, with $t_j = jh_t$ and h_t the chosen time stepsize. The values $\lambda = 1/2$ and $\lambda = 1$ give rise to the standard Crank-Nicolson (CN) and fully-implicit methods, respectively. We can view the above time-stepping technique as equivalent to solving a BVP at each time step.

For the space discretization, we apply collocation, based on quadratic or cubic splines. Let $\Delta^j \equiv \{x_0 \equiv \omega_1 < x_1 < \cdots < x_n \equiv \omega_2\}$ be the partition of Ω at time t_j and $U^j(x) = \sum_i \theta_i^j \phi_i^j(x)$ be the spline approximation to $u(x, t_j)$ written in terms of appropriate spline basis functions $\phi_i^j(x)$. Let also D^j be the set of collocation points at time t_j. The standard spline collocation approximation $U^j(x)$ is computed by forcing it to satisfy (4) for $x \in D^j$, and the boundary conditions arising from (2), that is, $\mathbf{B}U^j(x) = \gamma$ at $x = \omega_1, \omega_2$.

This spline collocation approximation turns out to be second-order, that is, non-optimal. In order to develop optimal spline collocation methods for Problem (1)-(3), we develop perturbations of $\mathbf{L}$ and $\mathbf{B}$ similar to those used to obtain optimal spline collocation methods for BVPs, see for example [1, 4, 6]. For simplicity, we assume $\mathbf{B}u \equiv u$, so that no perturbation is needed for $\mathbf{B}$. Thus we have the *extrapolated* or *one-step* method (1QSC-CN) in which u_Δ^j is determined by the equations

$$(\mathbf{I} - \lambda h_t(\mathbf{L} + \mathbf{P_L}))u_\Delta^j(x) = (\mathbf{I} + (1 - \lambda)h_t(\mathbf{L} + \mathbf{P_L}))u_\Delta^{j-1}(x) + h_t(\lambda g^j(x) + (1 - \lambda)g^{j-1}(x)),\ x \in D^j, \tag{5}$$

and the boundary conditions $u_\Delta^j(x) = \gamma$ at $x = \omega_1, \omega_2$. We also have the *deferred-correction* or *two-step* method (2QSC-CN) that first computes a second-order approximation U^j by

$$(\mathbf{I}-\lambda h_t\mathbf{L})U^j(x) = (\mathbf{I}+(1-\lambda)h_t\mathbf{L})U^{j-1}(x)+h_t(\lambda g^j(x)+(1-\lambda)g^{j-1}(x)),\ x \in D^j, \tag{6}$$

then an optimal order approximation u_Δ^j by

$$(\mathbf{I} - \lambda h_t\mathbf{L})u_\Delta^j(x) = (\mathbf{I} + (1 - \lambda)h_t(\mathbf{L} + \mathbf{P_L}))u_\Delta^{j-1}(x) + h_t\lambda\mathbf{P_L}U^j(x) + h_t(\lambda g^j(x) + (1 - \lambda)g^{j-1}(x)),\ x \in D^j. \tag{7}$$

Both U^j and u_Δ^j also satisfy the boundary conditions (2). When $\lambda = 1/2$ the methods 1QSC-CN and 2QSC-CN give rise to discretization errors of $O(h_t^2+h^4)$, locally at the gridpoints and midpoints of the space partition, where h is the (maximum) space partition stepsize.

Each set of equations (5), (6) and (7) results in a linear system to be solved. Method 1QSC-CN solves one linear system per timestep, while 2QSC-CN solves two but sparser linear systems. We propose a method that solves one linear system per timestep with the same matrix as that arising from 2QSC-CN.

The method QSC-CN is an one-step, still non-extrapolated method, that computes u_Δ^j by the equations

$$(\mathbf{I} - \lambda h_t \mathbf{L}) u_\Delta^j(x) = (\mathbf{I} + (1-\lambda) h_t \mathbf{L} + h_t \mathbf{P_L})) u_\Delta^{j-1}(x) + h_t(\lambda g^j(x) + (1-\lambda) g^{j-1}(x)),\ x \in D^j, \quad (8)$$

and by the boundary conditions $u_\Delta^j(x) = \gamma$ at $x = \omega_1, \omega_2$. One can view this method as combining (6) and (7) into one equation and treating the term $h_t \lambda \mathbf{P_L} U^j$ of (7) explicitly, i.e. substituting u_Δ^{j-1} in place of U^j.

Stability properties and improvements

In this section, we summarize the stability properties of the QSC-CN method (8). Let

$$A\theta^j = B\theta^{j-1} + \mathbf{g}^j \quad (9)$$

be the matrix problem arising from (8), and let $R = A^{-1}B$ be the iteration matrix. As expected, treating the term $h_t \lambda \mathbf{P_L} U^j$ of (7) explicitly, has negative effects in the stability properties of QSC-CN. More specifically, for the model PDE problem $u_t = p u_{xx}$ with homogeneous Dirichlet boundary conditions, discretized on a uniform partition, we have $\rho(R) < 1$, if $\sigma \leq 5.06$, where $\sigma = p\frac{h_t}{h^2}$. This means that we have a stepsize restriction to preserve stability. On the other hand, the 2QSC-CN method (i.e. equations (6) and (7)) does not have a stepsize restriction, as is expected for a CN-based (or implicit) method.

We examined the source of the instability of QSC-CN, and we found that it is the perturbation term $\mathbf{P_L} u_\Delta^{j-1}$ of (8) corresponding to the first and last midpoints (collocation points) that is responsible for the fact that some eigenvalues of R may become larger than 1 in magnitude. We examined some remedies to this issue, namely altering the perturbation term $\mathbf{P_L} u_\Delta^{j-1}$ at the first and last midpoints. One remedy approximates the term by an $O(h)$ approximation obtained by the perturbation terms of the nearby midpoints (QSC-CN1), another sets the term so that the eigenvalues of R satisfy certain desirable formulae (QSC-CN2), and a third omits this perturbation completely (QSC-CN0).

Numerical experiments show that, all these remedies result in methods without stability restrictions, and that the third remedy (resulting in method QSC-CN0) gives rise to the smallest $\rho(R)$, and slightly more accurate approximations, even more accurate than 2QSC-CN. Similar results where obtained for cubic splines. Note that [1], for extrapolated cubic spline methods, also alters the perturbation corresponding to collocation points near the boundary.

The case of convection-dominated problems needs to be dealt carefully, since quadratic splines require a perturbation for the term qu_x of $\mathbf{L}$ in order to reach the optimal order of convergence, while cubic splines do not. For a model problem

of the form $u_t = pu_{xx} + qu_x$ with homogeneous Dirichlet boundary conditions, discretized on a uniform partition, QSC-CN0 is stable if $h_t \leq c\frac{p}{q^2}$, with $c \approx 25$, while the respective cubic spline method, CSC-CN0, does not have a stability restriction.

Adaptive mesh methods

Recently, quadratic and cubic spline collocation methods with optimal order of convergence have been developed on non-uniform grids and integrated with adaptive mesh techniques [3, 2]. We incorporate adaptive mesh techniques in the space discretization of parabolic PDEs. More specifically, at each timestep, we incorporate the Algorithm PlaceMap in [2], that uses the error equidistribution principle to move the partition points in order to obtain a better error distribution, and thus a smaller error.

To proceed from t_{j-1} to t_j, we solve (8) using the partition Δ^{j-1} at t_{j-1}. If PlaceMap needs to redistribute the partition points, a new partition Δ^j is computed and (8) is re-solved using interpolated values from the previous timestep in the right hand side of (8). The algorithm for timestepping from t_{j-1} to t_j using an adaptive mesh technique is summarized as follows:
1. Let $\Delta^j = \Delta^{j-1}$
2. Compute $u^j_{\Delta^j}$ by solving (8)
3. Apply PlaceMap to possibly obtain a new Δ^j
4. If Δ^j remains the same, proceed to line 7, else
5. Interpolate $u^{j-1}_{\Delta^{j-1}}$ to obtain $u^{j-1}_{\Delta^j}$
6. Compute $u^j_{\Delta^j}$ by solving (8)
7. Proceed to step j
For slowly evolving functions, lines 6 and 7 in the above algorithm can be combined in
6, 7. Interpolate $u^j_{\Delta^{j-1}}$ to obtain $u^j_{\Delta^j}$

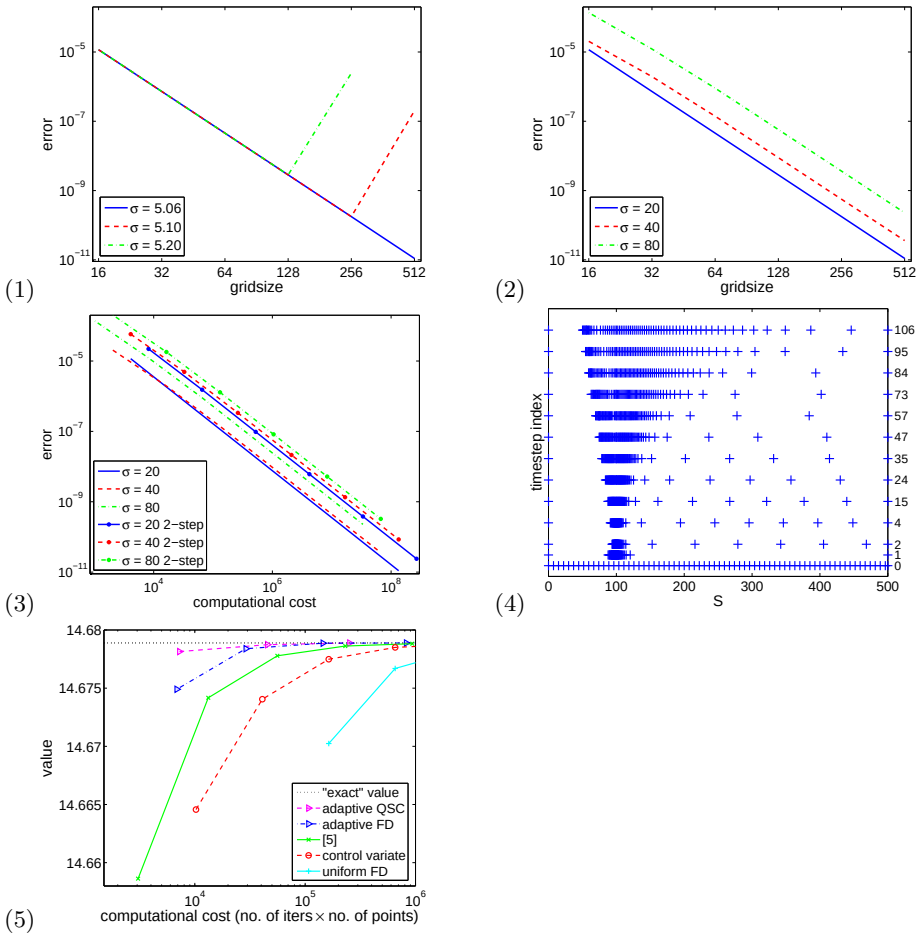
Numerical results

In Figures 1 and 2, we present graphically indicative results that verify the unconditional stability of QSC-CN0 and the conditional stability of QSC-CN, for a model problem of the form $u_t = pu_{xx}$, with $u(x,t) = \sin(\pi x)\exp(-t)$. In Figure 3, we see that QSC-CN0 is at least twice more efficient than 2QSC-CN.

We applied the adaptive mesh QSC-CN method to the American put option pricing problem. We model the problem using the Black-Scholes PDE with appropriate initial and boundary conditions, together with additional conditions that define the free boundary for this problem. We solve the free-boundary problem by a penalty method applied to the nonlinear PDE $v_t - \mathbf{L}v = \hat{\rho}\max\{f - v, 0\}$, where $\mathbf{L}v \equiv \frac{\hat{\sigma}^2 S^2}{2} v_{SS} + \hat{r}Sv_S - \hat{r}v$, with $\hat{\sigma}$ being the volatility, $\hat{r}$ the risk-free rate, $f = \max\{E - S, 0\}$ the payoff function (also initial state), E the strike price, S the asset price that plays the role of the space variable, and $\hat{\rho}$ a large number (penalty parameter). In the example chosen, $E = 100$, $\hat{\sigma} = 0.80$, $\hat{r} = 0.10$, $T = 0.25$, and $\hat{\rho} = 10^7$.

In Figure 4, we show the location of the partition points at selected timesteps, as computed by the adaptive algorithm. We start with a uniform grid (as if we do not know how the solution behaves). At the first timestep, the points are concentrated around the strike. As the time evolves, the points spread to cover the interval between the free boundary (which moves from E to the left) and E, with concentration around the free boundary. Almost no points are needed to the left of the free boundary (where the solution is linear) and few points are needed towards the right end of the interval, where the solution is almost linear.

Figure 11 shows that the QSC-CN adaptive mesh method outperforms the adaptive FD method (our implementation), the non-uniform FD method of [5], as well as other methods. The "exact" value was computed by the data in [5] and extrapolation.



References

1. Archer, D. An $O(h^4)$ cubic spline collocation method for quasilinear parabolic equations. *SIAM J. Numer. Anal. 14*, 4 (1977), 620–637.
2. Christara, C. C., and Ng, K. S. Adaptive techniques for spline collocation. *Computing 76*, 3 (2006), 259 – 277.
3. Christara, C. C., and Ng, K. S. Optimal quadratic and cubic spline collocation on nonuniform partitions. *Computing 76*, 3 (2006), 227 – 257.
4. Daniel, J. W., and Swartz, B. K. Extrapolated collocation for two-point boundary-value problems using cubic splines. *J. Inst. Maths Applics 16* (1975), 161–174.
5. Forsyth, P. A., and Vetzal, K. Quadratic convergence for valuing American options using a penalty method. *SIAM J. Sci. Comput. 23*, 6 (2002), 2095–2122.
6. Houstis, E. N., Christara, C. C., and Rice, J. R. Quadratic-spline collocation methods for two-point boundary value problems. *Internat. J. Numer. Methods Engrg. 26* (1988), 935–952.

The singular function boundary integral method for two- and three-dimensional elliptic boundary value problems with boundary singularities

Evgenia Christodoulou, Miltiades Elliotis, Christos Xenophontos, and Georgios Georgiou

Department of Mathematics and Statistics, University of Cyprus, P.O. Box 20537, 1678 Nicosia, Cyprus
georgios@ucy.ac.cy http://www.ucy.ac.cy/∼georgios

Abstract. The Singular Function Boundary Integral Method (SFBIM) for two-dimensional elliptic boundary value problems with boundary singularities is reviewed. In this method, the solution is approximated by the leading terms of the local asymptotic solution expansion which are also used to weight the governing PDE in the Galerkin sense. The singular coefficients, i.e. the coefficients of the asymptotic expansion of the solution, are unknowns to be calculated. By means of the divergence theorem the discretized equations are reduced to boundary integrals. The Dirichlet boundary conditions are weakly enforced by means of Lagrange multipliers, the values of which are calculated together with the singular coefficients. For two-dimensional Laplacian problems, we have shown that the approximate singular coefficients converge to the true ones at an exponential rate, as the number of singular functions increases. This is demonstrated via several applications, including ones involving the biharmonic operator which can be viewed as an extension of the theory.

Introduction

Planar elliptic boundary value problems with boundary singularities have been extensively studied in the last few decades. Many different methods have been proposed for the solution of such problems, ranging from special mesh-refinement schemes to sophisticated techniques that incorporate, directly or indirectly, the form of the local asymptotic expansion, which is known in many occasions. The local solution, centered at the singular point, in polar coordinates (r, θ) is of the general form:

$$u(r,\theta) \;=\; \sum_{j=1}^{\infty} a_j \, r^{\mu_j} \;\; f_j(\theta)\,, \tag{1}$$

where μ_j are the eigenvalues and f_j are the eigenfunctions of the problem, which are uniquely determined by the geometry and the boundary conditions along the boundaries sharing the singular point. The singular coefficients α_j, also known as generalized stress intensity factors or flux intensity factors, are determined by

the boundary conditions in the remaining part of the boundary. In the past few years, Georgiou and co-workers [1,2] developed the Singular Function Boundary Integral Method (SFBIM), in which the unknown singular coefficients are calculated directly. The solution is approximated by the leading terms of the local asymptotic solution expansion and the Dirichlet boundary conditions are weakly enforced by means of Lagrange multipliers. The method has been tested on standard Laplacian and biharmonic problems, yielding extremely accurate estimates of the leading singular coefficients, and exhibiting exponential convergence with respect to the number of singular functions. In the present paper, the SFBIM is reviewed and its convergence is discussed.

The singular function boundary integral method (SFBIM)

We consider a general Laplacian problem with a boundary singularity: Find u such that:

$$\nabla^2 u \;=\; 0 \qquad\qquad \text{in} \;\; \Omega\,, \tag{2}$$

with

$$\left.\begin{array}{lll} \dfrac{\partial u}{\partial n} \;=\; 0 & \text{on} & S_1 \\ u \;=\; 0 & \text{on} & S_2 \\ u \;=\; f(r,\theta) & \text{on} & S_3 \\ \dfrac{\partial u}{\partial n} \;=\; g(r,\theta) & \text{on} & S_4 \end{array}\right\}\,, \tag{3}$$

where $\partial\Omega{=}S_1 \cup S_2 \cup S_3 \cup S_4$, Ω has a smooth boundary with the exception of a boundary singularity at the corner O, formed by the straight boundary segments S_1 and S_2. In the remaining parts of the boundary, either Dirichlet or Neumann boundary conditions apply and the given functions f and g are such that no other boundary singularity is present.

In general, the asymptotic expansion of the solution is given by expansion (1). The SFBIM is based on the approximation of the solution by the leading terms of the local solution expansion:

$$\bar{u} \;=\; \sum_{i=1}^{N_\alpha} \bar{\alpha}_i \, W^i \tag{4}$$

where N_α is the number of singular functions used, which are defined by $W^i \equiv r^{\mu_i} f_i(\theta)$. Note that this approximation is valid only if Ω is a subset of the convergence domain of expansion (1). By applying Galerkin's principle and by double application of Green's second identity, the problem is discretized as follows:

$$\int_{\partial\Omega} W^i \, \frac{\partial \bar{u}}{\partial n} \, dS \;-\; \int_{\partial\Omega} \bar{u} \, \frac{\partial W^i}{\partial n} \, dS \;+\; \int_{\Omega} \bar{u} \, \nabla^2 W^i \, dV = 0, \quad i = 1, 2, \ldots, N_\alpha\,. \tag{5}$$

Since the singular functions are harmonic and exactly satisfy the boundary conditions along S_1 and S_2, the dimension of the problem is reduced by one and for

$i = 1, 2, \ldots, N_\alpha$, we have:

$$\int_{S_3} \left(\frac{\partial \bar{u}}{\partial n} W^i \, - \, \bar{u} \, \frac{\partial W^i}{\partial n} \right) dS + \int_{S_4} \left(W^i \, \frac{\partial \bar{u}}{\partial n} \, - \, \bar{u} \, \frac{\partial W^i}{\partial n} \right) dS \, = \, 0 \ . \quad (6)$$

It should be noted that the integrands in Eq. (6) are non-singular and all integrations are carried out far from the boundaries causing the singularity. To impose the Neumann condition along S_4, we simply substitute the normal derivative by the known function g. The Dirichlet condition along S_3 is imposed by means of a Lagrange multiplier function, λ, replacing the normal derivative. The function λ is expanded in terms of standard, polynomial basis functions M^j,

$$\lambda \, = \, \frac{\partial \bar{u}}{\partial n} \, = \sum_{j=1}^{N_\lambda} \lambda_j \, M^j, \quad (7)$$

where N_λ represents the total number of the unknown discrete Lagrange multipliers (or, equivalently, the total number of Lagrange-multiplier nodes) along S_3. The basis functions M^j are used to weight the Dirichlet condition along the corresponding boundary segment S_3. We thus obtain the following system of $N_\alpha + N_\lambda$ discretized equations:

$$\int_{S_3} \left(\lambda \, W^i \, - \, \bar{u} \, \frac{\partial W^i}{\partial n} \right) dS \, - \, \int_{S_4} \bar{u} \, \frac{\partial W^i}{\partial n} \, dS \, = \, - \int_{S_4} W^i \, g(r, \theta) \, dS \, , \quad (8)$$

$$\int_{S_3} \bar{u} \, M^j \, dS \, = \, \int_{S_3} f(r, \theta) \, M^j \, dS, \quad (9)$$

where $i = 1, 2, \ldots, N_a$ and $j = 1, 2, \ldots, N_\lambda$. It is easily shown that the system of Eqs. (8) and (9) is symmetric and nonsingular, provided $N_a > N_\lambda$. The "optimal" relationship between these two parameters will be discussed in the conference.

The theoretical convergence of the SFBIM in the case of Laplacian problems is discussed in Ref. [2]. Numerical results will be presented for test Laplacian problems and biharmonic elasticity problems. The formulation of the SFBIM for a certain 3-D Laplace problem with a straight-edge singularity is currently investigated. The extraction of the leading singular functions for this problem is based on the work of Yosibash et al. [3].

References

1. G.C. Georgiou, L.G. Olson, and Y. Smyrlis, A singular function boundary integral method for the Laplace equation, *Commun. Numer. Meth. Eng.* **12** (1996) 127-134.
2. C. Xenophontos, M. Elliotis, and G. Georgiou, The singular function boundary integral method for elliptic problems with singularities, *SIAM J. Sc. Comp.* **28** (2006) 517-532.
3. Z. Yosibash, R. Actis, and B. Szabó, Extracting edge flux intensity factors for the Laplacian, *Int. J. Numer. Methods Engin.* **53** (2002) 225-242.

A Symbolic-Numeric Software Package for the Computation of the GCD of Several Polynomials

Dimitrios Christou[1], Nicos Karcanias[1] and Marilena Mitrouli[2]

[1] Control Engineering Research Centre, School of Engineering and Mathematical Sciences, City University, Northampton Square, EC1V 0HB, London, U.K.
[2] Department of Mathematics, University of Athens, Panepistemiopolis 15784, Athens, Greece.
dchrist@math.uoa.gr, N.Karcanias@city.ac.uk, mmitroul@math.uoa.gr

Abstract. This survey is intended to present a package of algorithms for the computation of exact or approximate GCDs of sets of several polynomials and the evaluation of the quality of the produced solutions. These algorithms are designed to operate in symbolic-numeric computational environments. The key of their effectiveness is the appropriate selection of the right type of operations (symbolic or numeric) for the individual parts of the algorithms. Symbolic processing is used to improve on the conditioning of the input data and handle an ill-conditioned subproblem and numeric tools are used in accelerating certain parts of an algorithm. A sort description of the basic algorithms of the package is presented by using the symbolic-numeric programming code of Maple.

Introduction

The interaction of different type of computations (symbolic-numerical) is challenging for the development of new algorithms and has become an interesting area of research in the last decade. The presented software package for the computation of the GCD of several polynomials includes the following algorithms:

1. The **ERES** algorithm, which is a matrix based algorithm for the computation of the GCD of polynomials [2, 9].
2. The **ResultantMatrix** algorithm, which constructs a Resultant type matrix for more than two polynomials [5].
3. The **PSVD1** algorithm, which computes the partial singular value decomposition [13] for rank-1 matrices.
4. The **Strength** algorithm, which evaluates the quality of an approximate GCD [7].

The algorithms have been implemented and thoroughly tested in the programming environment of Maple and we shall present them in a matrix based formulation by using the notation of the programming code of Maple and routines of the LinearAlgebra package, which is included in Maple 8 and later versions.

Description and implementation of the algorithms

The Hybrid ERES algorithm

The main algorithm that we use for the computation of the GCD is the ERES algorithm [2, 9].

```
Hybrid_Eres_GCD := proc ( Pmn :: set(polynom), et::float )

INPUT  : et  = Tolerance for the termination
               criterion of the algorithm.
OUTPUT : gcd = Vector of the GCD coefficients.

  Pm := BasisMatrix( Pmn );
  P  := convert( Pm, rational );
  k  := 1;
  While k > 0 do                          # MAIN ITERATIVE PROCEDURE
      r, q := Dimensions( P );
      di := RowDegree( P, i=1..r );
      RowReorder( P );

      If max( di ) = min( di ) then
         Pf := convert( P, float );
         Normalize( Pf );
         rho, sigma, tol, w := PSVD1( Pf, et )
         If rho=1 then                    # TERMINATION CRITERION
            If hastype( Pm, float ) then
               gcd := w;                  # GCD FROM SVD
            else
               gcd := P[1,1..q];          # GCD FROM THE MATRIX
            end if;
            break;
         end if;
      end if;

      Scale( P );
      GaussianElimination( P );
      Shifting( P );
      Shrink( P );

  end do;
  gcd;
end proc;
```

Description of subroutines :

`BasisMatrix( Pmn )`	: Creates the initial $m \times n + 1$ basis matrix of the given set Pmn from the coefficients of its polynomials.
`RowDegree(P,i=1..r)`	: Specifies the degree $d_i,\ i = 1..r$ of each polynomial row of the matrix P.
`RowReorder( P )`	: Reorder matrix P : $d_{i-1} \leq d_i$ for all $i = 2..r$.
`Normalize( Pf )`	: Normalizes the rows of Pf using the Euclidean norm.
`Scale( P )`	: Scales the the matrix P such that P[1,1] > P[i,1] for all $i = 1..r$.
`Shifting( P )`	: Apply the shifting operation on every row of P,
`Shrink( P )`	: Deletes the zero rows and columns of P.

The PSVD1 algorithm : The PSVD1 algorithm is based on the methodology of the partial singular value decomposition [12, 13]. It is actually a variation of the classical singular value decomposition method [4, 1], especially developed for the efficient computation of the unique singular value and its right singular vector of an approximate ε_t-rank 1 matrix. The PSVD1 algorithm is a quick and effective tool for the detection of an approximate rank 1 matrix. It can increase the performance of other methods, such as the ERES method, and can be implemented easily in any software programming environment.

```
PSVD1 := proc( A::Matrix, theta::float )

INPUT:  A     = m x n matrix with real floating-point data.
        theta = a small positive bound such that
                 sigma_1=<...=<sigma_{k-1}=<theta<=sigma_{k}
                 where k = min{m,n}.

OUTPUT: rho   = Indicator of the rank of the matrix.
        sigma = Maximum singular value.
        w     = Respective right singular vector.
        tol   = Recommended tolerance for a potential
                 rank-1 matrix.
```

The Strength of an approximate GCD

The quality of a given approximate GCD known as the *strength* of the approximate GCD is evaluated by the following algorithm. A rigorous definition of the approximate GCD has been given recently [7] that allows the computation of the strength of approximation and sets up a framework for computing the *optimal approximate GCD*. This approach is based on recent results on the representation of the GCD of many polynomials in terms of the factorisation of the generalised resultant and a Toeplitz matrix representation of the GCD [5].

The procedure Strength

```
minstrength := proc( Pmn::set(polynom), GCD::polynom )

INPUT:  Pmn = set of m univariate polynomials.
        GCD = a polynomial which is given as a GCD
              from a GCD algorithm.

OUTPUT: St  = strength of the given GCD.
```

The strength of a given approximation can be computed by the routine : Optimization[Minimize] in Maple.

References

1. Barlow, J.L.: More accurate bidiagonal reduction for computing the singular value decomposition. SIAM J. Matrix Anal. Appl. **23** No. 3 (2002) 761–798.
2. Christou, D., Mitrouli, M.: Estimation of the Greatest Common Divisor of many polynomials using hybrid computations performed by the ERES method. Appl. Num. Anal. and Comp. Math. **2** No. 3 (2005), 293–305.
3. Corless, R.M., Gianni, P.M., Trager, B.M., WattS, M.: The Singular Value Decomposition for Polynomial Systems. Proc. ISSAC'95 Quebec Canada (1995) 195–207
4. Datta, B.N.: Numerical Linear Algebra and Applications. Brooks/Cole Publishing Company - ITP (1995)
5. Fatouros, S., Karcanias, N.: Resultant properties of the GCD of many polynomials and a factorisation representation of GCD. Int. Journ. Control **76** No. 16 (2003), 1666–1683
6. Karcanias, N.: Invariance properties and characterisation of the greatest common divisor of a set of polynomials. Int. Journ. Control **46** (1987) 1751–1760
7. Karcanias, N., Fatouros, S., Mitrouli, M., Halikias, G.H.: Approximate greatest common divisor of many polynomials, generalised resultants and strength of approximation. Computers & Mathematics with Applications **51** Issue 12 (2006) 1817–1830
8. Karcanias, N., Mitrouli, M.: Approximate algebraic computations of algebraic invariants, Symbolic methods in control systems analysis and design. IEE Control Engin. Series **56** (1999) 162–168
9. Mitrouli, M., Karcanias, N.: Computation of the GCD of polynomials using Gaussian transformation and shifting. Int. Journ. Control **58** (1993) 211–228
10. Noda, M.T., Sasaki, T.: Approximate GCD and its applications to ill-conditioned algebraic equations. Jour. of Comp. and Appl. Math. **38** (1991) 335–351
11. Rupprecht D.: An algorithm for computing certified approximate GCD of n univariate polynomials. J. of Pure and Applied Algebra **139** (1999) 255–284
12. Van Huffel, S., Vandewalle, J.: An efficient and reliable algorithm for computing the singular subspace of a matrix, associated with its smallest singular values. Journ. of Comp. and Applied Math. **19** (1987) 313–330
13. Van Huffel, S.: Partial Singular Value Decomposition Algorithm Journ. of Comp. and Applied Math. **33** (1990) 105–112

Linear versus nonlinear models for evaluation of brain connectivity from neurophysiological time series

Laura Cimponeriu[1], Maria Stavrinou[2]
Liviu Moraru[2], and Anastasios Bezerianos[2]

[1] University of Potsdam, Am Neuen Palais,
PF 601553, D-14415 Potsdam
[2] University of Patras, Dept. of Medical Physics, School of Medicine,
26500 University Campus, Rio

Abstract. The aim of the present study is to test two approaches to inferring connectivity among sources of neural oscillations. The first approach is based on linear frequency domain analysis of interdependencies in multivariate time series using the measures of partial coherence and partial directed coherence. The second approach is based on nonlinear dynamic models of coupled oscillator systems. The results of experimental data analysis show that, even though the two approaches are conceptually different, they provide similar estimates of the degree of mutual engagement of the active brain areas during the considered task. Inconsistency in what regards the directional aspects of the inferred interactions might reflect differential sensitivity of measures to particular aspects of coupling.

Introduction

How to infer brain connectivity from experimental data? The question has received a great deal of attention from both experimental and theoretical neuroscience communities, since brain function and its computational properties appear to be a direct consequence of its circuitry. Nowadays, technologically advances in measurement of brain activity based on a variety of neuroimaging methods enable the acquisition of massive multimodal data which form the basis of investigation of brain network structure and dynamics.

A number of methodological approaches have been utilized to recover the underlying network of interactions from multisite recordings of brain activity [5]. The structural and functional aspects of reconstructed networks have been further studied using graph theoretical concepts and methods. However, the topology of the inferred brain functional networks may depend on the specific way interaction is inferred from the data.

The aim of the present study is twofold: 1). to compare and contrast two approaches for inferring interaction from multivariate data and 2). to apply them to the problem of inferring functional connectivity among sources of neural oscillations engaged in real and imagined human movements.

Partial Directed Coherence (PDC)

To study connectivity in the frequency domain, particularly relevant to neural signals analysis, the partial directed coherence (PDC) [3] has been introduced within the multivariate autoregressive (MAR) modeling framework. For a stationary time series of state vectors $s_t \in \Re^D$ (adjusted to have mean zero), the MAR process of order p is defined by $S_t = \sum_{k=1}^{p} A_k s_{t-k} + E_t$ where the matrices $A_k \in \Re^{D \times D}$ are the coefficient matrices and the noise term E_t (error-term) is an D-dimensional iid sequence with mean zero and covariance matrix Σ. Autoregressive coefficients $a_{(i,j)(k)}$, $i, j = 1, ..., D$ represent the linear interaction effect of $s(t-k)$ onto $s(t)$. Several numerical methods are available for estimating the parameters of the model [1]. The spectral matrix of a MAR process is defined by $S(f) = H(f)\Sigma H^{H(f)}$ where the subscript $(.)^H$ denotes the Hermitian transpose and $H(f) = [I - A(f)]^{-1} = [\bar{A}(f)]^{-1}$. $A(f)$ is given by the Fourier transform of the coefficients, $A_{ij}(f) = \sum_{k=1}^{p} a_{k,ij} e^{-i2\pi kf}$. The inverse spectral matrix can be used to derive the partial spectral density matrix: $PS_{ij}(f) = \sum_k \bar{A}^*_{ki}(f)\bar{A}_{kj}(f) / [\sum_k \bar{A}^*_{ki}(f)\bar{A}_{ki}(f)) \sum_k \bar{A}^*_{kj}(f)\bar{A}_{kj}(f)]^{1/2}$

It is then possible to define *partial directed coherence* (PDC) [3] as $\pi_{ij}(f) = \bar{A}_{ij}(f) / \sqrt{\sum_{k=1}^{N} \bar{A}^*_{ki}(f)\bar{A}_{kj}(f)}$. The PDC ($\pi_{ij}(f)$) measures the linear influence from s_j to s_i, whereupon common effects produced by all other signals are excluded. PDC measure is normalized, $\pi^2_{ij} \in [0, 1]$. Data non-stationarity and observational noise limit interpretation of MAR-based inferred interactions in terms of genuine causal effects [2]

Phase modeling analysis (PMC)

The second approach to inferring interaction among processes, restricted to oscillatory systems is based on phase dynamics modeling. The important theoretical idea put forth by phase models is that weak interaction of self-sustained oscillators affects only their phases, whereas the amplitudes can be considered as unchanged [6]. The reduced description of weakly coupled oscillators takes the form

$$\begin{aligned} \dot{\phi}_i &= \omega_i + f_1(\phi_i, \phi_j) + \xi_i, \\ \dot{\phi}_j &= \omega_j + f_2(\phi_i, \phi_j) + \xi_j, \end{aligned} \tag{1}$$

where $\omega_{i,j}$ are frequencies of uncoupled systems, $\phi_{i,j}$ are the two phase variables, and functions $f_{i,j}$, 2π-periodic with respect to their arguments, describe the coupling. The irregular terms $\xi_{i,j}$ correspond to perturbations to the phase dynamics due to noise and/or the chaotic nature of amplitudes. Aside from large interest in phase models for explaining cooperative behavior and synchronization phenomena in large ensemble of oscillators, recently it has been showed that they can be used for inferring features of interaction (strength, directionality and time delay) from data [7],[4]. The strength of coupling between two oscillators i and j can be quantified by the mean phase coherence index, defined as $\rho_{i,j} = | < e^{(i(n\phi_i - m\phi_j))} > |$, where n, m are integers and average is taken

over time. The index of synchronization is normalized, with $\rho_{i,j} = 1$ reflecting perfect locking, and $\rho_{i,j} = 0$ indicating a uniform distribution of the relative phase. Estimation of asymmetric relations between oscillators can be obtained by reconstructing the coupling functions $f_{i,j}(.)$ from data. Using a Fourier expansion, as the most convenient way to handle periodic functions, $f_{i,j}(.)$ are approximated by $\bar{F}(\phi_i, \phi_j) = \sum_{m,n} A_{m,n} e^{(i(n\phi_i + m\phi_j))}$. Coefficients $A_{m,n}$ can be estimated by least-square regression. Further, integrative measures of influence of one oscillator on its counterpart can be derived [7]. Here, we quantify the strength of the directional coupling from oscillator j to i by coefficients $PC_{i,j} = \sum_{m \neq 0, n=0} |A_{m,n}|^2$.

Experimental data analysis

The experimental data come from EEG measurements of ERP (event related potentials) from healthy subjects performing an auditorily paced finger tapping task (inter-stimulus interval of 1.5s), in real and imaginary situation. The data

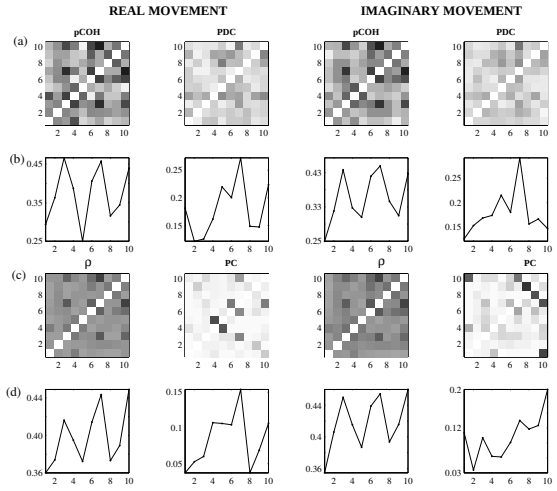


Fig. 1. PDC and PMC -based estimation of cortical connectivity. (a),(c) shows the connectivity matrices (significant coupling coefficients lie in $[0, 1]$), whereas (b),(d) shows the strength of each node (electrode site).

consists of 100 single trial measurements at 10 electrodes sites located over the brain areas involved in motor function. Details on the experimental protocol and data pre-processing can be found in [8]. Our interest was in reconstructing the functional connectivity in the beta frequency range, widely observed in sensorimotor cortex in relation to motor behavior in humans. Assuming the ERP

data as realizations of the same (multivariate) stochastic process, the information from all the trials has been used in PDC and PMC approaches to increase the statistical significance of the fitted model parameters. Structural properties of the inferred undirected (by partial coherence (pCOH) and ρ) and directed (by PDC and PC) networks are quantified by coarse-grained measurement using the node strength given by $s_i = (1/10)\sum_i w_{ji}$, where w_{ji} denotes the coupling strength. To note, that for directed networks s_i defined above represents the out-strength of the node. Fig.1 summarizes the results of data analysis.

Discussions and conclusions

The results of EEG data analysis show that, even though the two approaches have different theoretical bases, the two methods provide similar estimates of the degree of mutual engagement of the active brain areas during both (real and imagery) experimental conditions. In addition, the results support the hypothesis that the functional connectivity over the contralateral hemisphere during finger tapping is preserved in imagery. The discrepancy between estimated directional influences among the co-activated brain areas can be attributed to the differential sensitivity of the measures to particular aspects of interaction.

Acknowledgments

We thank the European Social Fund (ESF), Operational Program for Educational and Vocational Training II (EPEAEK II), and particularly the Program PYTHAGORAS II, for funding the above work.

References

1. Akaike H., A new look at the statistical model identification, IEEE Transaction on Automatic Control AC 19 (6) (1974) 716723.
2. Baccala,L.A. and Sameshima,K. Comments on Is partial coherence a viable technique for identifying generators of neural oscillations? Biol. Cybern., 95, (2006) 135141.
3. Baccala, L. A., and Sameshima, K. Partial directed coherence: A new concept in neural structure determination. Biological Cybernetics, 84(6), (2001) 463-474.
4. Cimponeriu L., Rosenblum M.G., and Pikovsky A.S., Estimation of delay in coupling from time series, Physical Review E, 70, (2004) 046213.
5. David O, Cosmelli D, Friston KJ. Evaluation of different measures of functional connectivity using a neural mass model.Neuroimage. 2004 Feb;21(2):659-73.
6. Kuramoto Y., Chemical Oscillations, Waves and Turbulence., 1984, Springer, Berlin.
7. Rosenblum M.G. and Pikovsky A.S., Detecting direction of coupling in interacting oscillators, Physical Review E, 64, (2001) 045202.
8. Stavrinou M. L., Moraru L., Cimponeriu L., Della Penna S., and Bezerianos A., Evaluation of cortical connectivity during real and imagined rhythmic finger tapping. Brain Topography, 19, 3, (2007) 1-9 Online First.

Separation of local extrema of least squares piecewise monotonic data fits

I.C. Demetriou

Unit of Mathematics & Informatics, Department of Economics,
University of Athens, 8 Pesmazoglou street, Athens 105 59.
demetri@econ.uoa.gr

Abstract. If function measurements include uncorrelated errors, then piecewise monotonic trends may be captured by a least squares fit to the data so that the sequence of the first differences of the components of the fit includes at most $k-1$ sign changes. Then the best fit is composed of at most k monotonic sections, alternately increasing and decreasing, whereas the positions of its extrema are calculated automatically by the optimization calculation. We prove that as the data increase, the positions of the corresponding extrema of the best fit increase as well. One strong corollary is that the local maxima of a best fit with $k-1$ monotonic sections are separated by the local maxima of the best fit with k monotonic sections, and local minima separate similarly. Interesting applications of these results may be found in global and local analyses of the data, in extrema forecasting concerning future events and in developing fast procedures for piecewise monotonic data fitting.

Modified SCRS method based on residual vector of BiCR method

Seiji Fujino[1], Yusuke Onoue[2], and Kuniyoshi Abe[3]

[1] Research Institute for Information Technology, Kyushu University
[2] Graduate School of Information Science and Electrical Engineering, Kyushu University
[3] Faculty of Economics and Information, Gifu Shotoku Gakuen University

Abstract. Conjugate Gradient Squared (CGS) method[2] is an attractive iterative method for the solution of a linear system of equations with nonsymmetric coefficient matrix[1]. However, the popularity of CGS method has diminished over time except for the context of device simulations becuase of instability of convergence rate[2]. Therefore some versions of CGS method which have stability of convergence as compared with CGS method have been proposed. However, the amount of computational costs per one iteration of some versions of CGS method are increased. Moreover some versions of CGS method badly affect its convergence rate and limit its attainable accuracy in some situations. In this article, we propose product types of BiCR iterative methods i.e., CRS[1], S(Stabilized) CRS and M(Modified) SCRS methods, whose residual vector is based on that of the original BiCR method.

SCRS method

In general, Lanczos polynomial $R_n(\lambda)$ satisfies the next alternative reccurence

$$R_0(\lambda) = 1, P_0(\lambda) = 1, \tag{1}$$
$$R_{n+1}(\lambda) = R_n(\lambda) - \alpha_n \lambda P_n(\lambda), \tag{2}$$
$$P_{n+1}(\lambda) = R_{n+1}(\lambda) + \beta_n P_n(\lambda), \quad n = 0, 1, \ldots \tag{3}$$

using auxiliary polynomial $P_n(\lambda)$. The residual $\boldsymbol{r}_n^{\mathrm{SCRS}}$ of S (Stabilized) CRS method is defined as a form of product of polynomial $R_n(\lambda)$ and accelerated polynomial $H_n(\lambda)$ as

$$\boldsymbol{r}_n^{\mathrm{SCRS}} = H_n(\lambda) R_n(\lambda) \boldsymbol{r}_0.$$

The accelerated polynomial $H_n(\lambda)$ satisfies the following reccurence

$$H_0(\lambda) = 1, \tag{4}$$
$$H_{n+1}(\lambda) = R_n(\lambda) - \omega_n \lambda P_n(\lambda), \quad n = 0, 1, \ldots. \tag{5}$$

,where parameter ω_n is decided from minimization of 2-norm of the residual vector $\boldsymbol{r}_{n+1}^{\mathrm{SCRS}}$. Next we update the product of polynomial of $H_{n+1}(\lambda) R_{n+1}(\lambda)$.

When we utilize reccurences (1)-(3) and the accelerated polynomials $H_{n+1}(\lambda)$,

$$H_{n+1}(\lambda)R_{n+1}(\lambda) = R_n(\lambda)R_n(\lambda) - \alpha_n\lambda R_n(\lambda)P_n(\lambda) - \omega_n\lambda R_{n+1}(\lambda)P_n(\lambda)$$

is gained, and the residual vector $\boldsymbol{r}_n^{\mathrm{SCRS}}$ can be updated by reccurence

$$\boldsymbol{r}_{n+1}^{\mathrm{SCRS}} = \boldsymbol{r}_n^{\mathrm{CRS}} - \alpha_n A\boldsymbol{e}_n - \omega_n A\boldsymbol{h}_n. \tag{6}$$

However, computational cost of multiplication of matrix and vector can be evaluated as three times per one iteration. Therefore, for reduction of computational cost we calculate $A\boldsymbol{h}_n$ as

$$A\boldsymbol{h}_n = A(\boldsymbol{e}_n + \boldsymbol{h}_n) - A\boldsymbol{e}_n. \tag{7}$$

Moreover, by defining auxiliary vector $\boldsymbol{q}_n := A\boldsymbol{p}_n$, the next reccurences

$$\boldsymbol{q}_{n+1} = A\boldsymbol{e}_{n+1} + \beta_n(A\boldsymbol{h}_n + \beta_n\boldsymbol{q}_n), \quad \boldsymbol{h}_{n+1} = \boldsymbol{e}_{n+1} - \alpha_{n+1}\boldsymbol{q}_{n+1} \tag{8}$$

are also adopted for reduction of the computational cost. As a result, the computational cost of multiplication of matrix and vector of SCRS method reduces to two times per one iteration. Parameter ω_n can be decided as

$$\omega_n = \frac{(A\boldsymbol{h}_n, \boldsymbol{r}_n^{\mathrm{CRS}} - \alpha_n A\boldsymbol{e}_n)}{(A\boldsymbol{h}_n, A\boldsymbol{h}_n)} \tag{9}$$

from minimization of 2-norm $||\boldsymbol{r}_{n+1}^{\mathrm{SCRS}}||_2$ of the residual vector $\boldsymbol{r}_{n+1}^{\mathrm{SCRS}}$. The solution vectors $\boldsymbol{x}_{n+1}$ and $\boldsymbol{x}_{n+1}^{\mathrm{CRS}}$ can be calculated using the next reccurence, respectively.

$$\boldsymbol{x}_{n+1} = \boldsymbol{x}_n^{\mathrm{CRS}} + \alpha_n\boldsymbol{e}_n + \omega_n\boldsymbol{h}_n, \quad \boldsymbol{x}_{n+1}^{\mathrm{CRS}} = \boldsymbol{x}_n^{\mathrm{CRS}} + \alpha_n(\boldsymbol{e}_n + \boldsymbol{h}_n). \tag{10}$$

Parameters α_n and β_n can be decided as

$$\alpha_n = \frac{(R_n(A)R_n(A)\boldsymbol{r}_0, A^T\boldsymbol{r}_0^*)}{(AR_n(A)P_n(A)\boldsymbol{r}_0, A^T\boldsymbol{r}_0^*)} = \frac{(\boldsymbol{r}_n^{\mathrm{CRS}}, A^T\boldsymbol{r}_0^*)}{(A\boldsymbol{e}_n, A^T\boldsymbol{r}_0^*)}. \tag{11}$$

from the following bi-orthogonal conditions:

$$R_n(A)\boldsymbol{r}_0 \ \perp \ A^T K_n(A^T; \boldsymbol{r}_0^*), \tag{12}$$

$$AP_n(A)\boldsymbol{r}_0 \ \perp \ A^T K_n(A^T; \boldsymbol{r}_0^*). \tag{13}$$

Similarly, parameter β_n is also written as

$$\beta_n = \frac{(AR_{n+1}(A)\boldsymbol{r}_0, R_{n+1}(A^T)\boldsymbol{r}_0^*)}{(AR_n(A)\boldsymbol{r}_0, R_n(A^T)\boldsymbol{r}_0^*)} = \frac{\alpha_n\omega_{n-1}}{\omega_n\alpha_{n-1}}\frac{(A\boldsymbol{r}_{n+1}^{\mathrm{SCRS}}, \boldsymbol{r}_0^*)}{(A\boldsymbol{r}_n^{\mathrm{SCRS}}, \boldsymbol{r}_0^*)}. \tag{14}$$

Furthermore for reducing computational cost of multiplication of matrix and vector, parameter β_n is calculated actually with transpose matrix of A as

$$\beta_n = \frac{\rho_1}{\rho_0}\frac{(\boldsymbol{r}_{n+1}^{\mathrm{SCRS}}, A^T\boldsymbol{r}_0^*)}{(\boldsymbol{r}_n^{\mathrm{SCRS}}, A^T\boldsymbol{r}_0^*)}, \tag{15}$$

using $\rho_0 := \alpha_{n-1}/\omega_{n-1}$ and $\rho_1 := \alpha_n/\omega_n$. At last we must note that we calculate $A^T \boldsymbol{r}_0^*$ once only before the iteration loop of iterative methods.

Algorithm 1: SCRS

Let $\boldsymbol{x}_0$ be an initial guess,
put $\boldsymbol{r}_0 = \boldsymbol{b} - A\boldsymbol{x}_0$,
Choose $\boldsymbol{r}_0^*$ such that $(\boldsymbol{r}_0, \boldsymbol{r}_0^*) \neq \boldsymbol{0}$,
Set $\beta_{-1} = 0,\ \boldsymbol{x}_0^{\mathrm{CRS}} = \boldsymbol{x}_0$,
For $n = 0, 1, \ldots,$ Do

$$\boldsymbol{q}_n = A\boldsymbol{e}_n + \beta_{n-1}(A\boldsymbol{h}_{n-1} + \beta_{n-1}\boldsymbol{q}_{n-1}) \tag{16}$$

$$\alpha_n = \frac{(\boldsymbol{r}_n^{\mathrm{CRS}}, \boldsymbol{r}_0^*)}{(A\boldsymbol{e}_n, \boldsymbol{r}_0^*)} \tag{17}$$

$$\boldsymbol{h}_n = \boldsymbol{e}_n - \alpha_n \boldsymbol{q}_n \tag{18}$$

$$A\boldsymbol{h}_n = A(\boldsymbol{e}_n + \boldsymbol{h}_n) - A\boldsymbol{e}_n \tag{19}$$

$$\omega_n = (A\boldsymbol{h}_n, \boldsymbol{r}_n^{\mathrm{CRS}} - \alpha_n A\boldsymbol{e}_n)/(A\boldsymbol{h}_n, A\boldsymbol{h}_n) \tag{20}$$

$$\rho_1 = \frac{\alpha_n}{\omega_n} \tag{21}$$

$$\boldsymbol{r}_{n+1}^{\mathrm{SCRS}} = \boldsymbol{r}_n^{\mathrm{CRS}} - \alpha_n A\boldsymbol{e}_n - \omega_n A\boldsymbol{h}_n \tag{22}$$

if $||\boldsymbol{r}_{n+1}^{\mathrm{SCRS}}||_2/||\boldsymbol{r}_0||_2 \leq \epsilon$ then (23)

$$\boldsymbol{x}_{n+1} = \boldsymbol{x}_n^{\mathrm{CRS}} + \alpha_n \boldsymbol{e}_n + \omega_n \boldsymbol{h}_n \tag{24}$$

end if and stop

$$\boldsymbol{r}_{n+1}^{\mathrm{CRS}} = \boldsymbol{r}_n^{\mathrm{CRS}} - \alpha_n A(\boldsymbol{e}_n + \boldsymbol{h}_n) \tag{25}$$

$$\boldsymbol{x}_{n+1}^{\mathrm{CRS}} = \boldsymbol{x}_n^{\mathrm{CRS}} + \alpha_n(\boldsymbol{e}_n + \boldsymbol{h}_n) \tag{26}$$

$$\beta_n = \frac{\rho_1}{\rho_0} \frac{(\boldsymbol{r}_{n+1}^{\mathrm{SCRS}}, \boldsymbol{r}_0^*)}{(\boldsymbol{r}_n^{\mathrm{SCRS}}, \boldsymbol{r}_0^*)} \tag{27}$$

$$\rho_0 = \rho_1 \tag{28}$$

$$\boldsymbol{e}_{n+1} = \boldsymbol{r}_{n+1}^{\mathrm{CRS}} + \beta_n \boldsymbol{h}_n \tag{29}$$

End Do

As well as the residual vector of SCRS method, residual vector $\boldsymbol{r}_n^{\mathrm{MSCRS}}$ of M (Modified) SCRS method is defined as follows:

$$\boldsymbol{r}_n^{\mathrm{MSCRS}} := H_n(A)R_n(A)\boldsymbol{r}_0 = \boldsymbol{r}_n^{\mathrm{CRS}} - \alpha_{n-1}A\boldsymbol{e}_{n-1} - \omega_{n-1}A\boldsymbol{h}_{n-1}. \tag{30}$$

Here, we introduce newly the following associate residual vector $\boldsymbol{a\_r}_n (:= H_n(A)R_{n-1}(A)\boldsymbol{r}_0)$ with the polynomial $R_{n-1}(A)$ at the $(n-1)$step of iteration loop as

$$H_n(A)R_{n-1}(A)\boldsymbol{r}_0 = (R_{n-1}(A) - \omega_{n-1}AP_{n-1}(A))R_{n-1}(A)\boldsymbol{r}_0 = \boldsymbol{r}_{n-1}^{\mathrm{CRS}} - \omega_{n-1}A\boldsymbol{e}_{n-1} \tag{31}$$

The parameter ω_{n-1} is decided from minimization of 2-norm $||\boldsymbol{r}_{n-1}^{\mathrm{CRS}} - \omega_{n-1}A\boldsymbol{e}_{n-1}||_2$ as

$$\omega_{n-1} = (\boldsymbol{r}_{n-1}^{\mathrm{CRS}}, A\boldsymbol{e}_{n-1})/(A\boldsymbol{e}_{n-1}, A\boldsymbol{e}_{n-1}). \tag{32}$$

Algorithm 2: Modified SCRS

Let $\boldsymbol{x}_0$ be an initial guess,
put $\boldsymbol{r}_0 = \boldsymbol{b} - A\boldsymbol{x}_0,\ \boldsymbol{r}_0^{\mathrm{MSCRS}} = \boldsymbol{r}_0$,
Choose $\boldsymbol{r}_0^*$ such that $(\boldsymbol{r}_0, \boldsymbol{r}_0^*) \neq \boldsymbol{0}$,
Set $\beta_{-1} = 0,\ \mathrm{sign} = 0,\ \boldsymbol{x}_0^{\mathrm{CRS}} = \boldsymbol{x}_0$,
For $n = 0, 1, \ldots,$ Do
$\boldsymbol{q}_n = A\boldsymbol{e}_n + \beta_{n-1}(A\boldsymbol{h}_{n-1} + \beta_{n-1}\boldsymbol{q}_{n-1})$
$\alpha_n = \dfrac{(\boldsymbol{r}_n^{\mathrm{CRS}}, A^T\boldsymbol{r}_0^*)}{(A\boldsymbol{e}_n, A^T\boldsymbol{r}_0^*)}$
$\boldsymbol{h}_n = \boldsymbol{e}_n - \alpha_n\boldsymbol{q}_n$
$A\boldsymbol{h}_n = A(\boldsymbol{e}_n + \boldsymbol{h}_n) - A\boldsymbol{e}_n$
$\boldsymbol{r}_{n+1}^{\mathrm{CRS}} = \boldsymbol{r}_n^{\mathrm{CRS}} - \alpha_n A(\boldsymbol{e}_n + \boldsymbol{h}_n)$
if $||\boldsymbol{r}_{n+1}^{\mathrm{CRS}}||_2/||\boldsymbol{r}_0||_2 \leq \delta$ then (Condition M1)
sign = 1
end if
if sign = 1 then (Condition M2)
$\omega_n = \dfrac{(\boldsymbol{r}_n^{\mathrm{CRS}}, A\boldsymbol{e}_n)}{(A\boldsymbol{e}_n, A\boldsymbol{e}_n)}$
$\boldsymbol{r}_{n+1}^{\mathrm{MSCRS}} = \boldsymbol{r}_n^{\mathrm{CRS}} - \alpha_n A\boldsymbol{e}_n - \omega_n A\boldsymbol{h}_n$
if $||\boldsymbol{r}_{n+1}^{\mathrm{MSCRS}}||_2/||\boldsymbol{r}_0^{\mathrm{MSCRS}}||_2 \leq \epsilon$ then
$\boldsymbol{x}_{n+1}^{\mathrm{MSCRS}} = \boldsymbol{x}_n^{\mathrm{CRS}} + \alpha_n\boldsymbol{e}_n + \omega_n\boldsymbol{h}_n$
end if and stop
end if
$\boldsymbol{x}_{n+1}^{\mathrm{CRS}} = \boldsymbol{x}_n^{\mathrm{CRS}} + \alpha_n(\boldsymbol{e}_n + \boldsymbol{h}_n)$
$\beta_n = \dfrac{(\boldsymbol{r}_{n+1}^{\mathrm{CRS}}, A^T\boldsymbol{r}_0^*)}{(\boldsymbol{r}_n^{\mathrm{CRS}}, A^T\boldsymbol{r}_0^*)}$
$\boldsymbol{e}_{n+1} = \boldsymbol{r}_{n+1}^{\mathrm{CRS}} + \beta_n\boldsymbol{h}_n$
End Do

The numerical results in Table 1 verify stability of convergence and efficiency of the SCRS and MSCRS methods as well as CRS method. SCRS and MSCRS methods outperform greatly compared with the CGS method.

References

1. Abe, K., Sogabe, T., Fujino, S., and Zhang, S.-L., A product-type Krylov subspace method based on Conjugate Residual method for nonsymmetric coefficient matrices, Transaction of IPSJ, Vol.48 No.SIG8(ACS18), 2007, pp.11-21.
2. Sonneveld, P., A Fast Lanczos-type Solver for Nonsymmetric Linear Systems, SIAM J. Sci. Stat. Comput., 10(1989), pp.36-52.

Table 1. Convergence property of iterative methods with accelerated ILU(0) preconditioner when 31 parameters γ are varied from 1.0 up to 1.3 at the equi-interval of 0.01 for seven matrices in the field of fluid analysis of Florida sparse matrix collection.

matrix	method	successful cases	ave. itr.	ave. time [sec.]	ave. true res. [$\log_{10}$]	ratio of time
EX10HS	CGS	15	2326	2.11	−7.14	1.000
	CRS	16	193	**0.19**	−7.04	**0.090**
	SCRS	16	320	0.32	−7.04	0.152
	MSCRS	14	184	**0.19**	−7.03	**0.090**
EX19	CGS	15	2351	10.70	−7.11	1.000
	CRS	19	299	1.44	−7.10	0.135
	SCRS	19	180	**0.92**	−7.09	**0.086**
	MSCRS	18	203	1.03	−7.07	0.096
FIDAP003	CGS	14	1924	1.44	−7.16	1.000
	CRS	19	326	0.26	−7.17	0.181
	SCRS	15	111	**0.10**	−7.03	**0.069**
	MSCRS	15	128	0.12	−7.04	0.083
FIDAP007	CGS	23	385	0.30	−7.34	1.000
	CRS	24	83	0.08	−7.10	0.267
	SCRS	22	49	**0.06**	−7.10	**0.200**
	MSCRS	23	78	0.08	−7.11	0.267
FIDAP010	CGS	7	5156	4.43	−7.15	1.000
	CRS	15	326	0.30	−7.11	0.068
	SCRS	14	298	**0.28**	−7.05	**0.063**
	MSCRS	11	341	0.32	−7.02	0.072
FIDAPM37	CGS	16	2084	21.80	−7.16	1.000
	CRS	17	733	7.97	−7.02	0.366
	SCRS	13	448	**5.19**	−7.02	**0.238**
	MSCRS	14	867	9.50	−6.98	0.436
LI	CGS	11	2951	82.51	−7.20	1.000
	CRS	12	90	**3.35**	−7.03	**0.041**
	SCRS	12	98	3.64	−7.15	0.044
	MSCRS	12	99	3.63	−7.08	0.044

Modeling the MagnetoEncephaloGram (MEG) of Epileptic Patients Using Genetic Programming

Efstratios Georgopoulos[1] Adam Adamopoulos[2] and Spiridon Likothanassis[3]

[1] Techological Educational Institute of Kalamata, Kalamata, Hellas
[2] Medical Physics Laboratory, Department of Medicine,
Democritus University of Thrace, 681 00, Alexandroupolis, Hellas
[3] Pattern Recognition Laboratory, Department of Computer Engineering and Informatics, University of Patras, 265 00, Patras, Hellas
efgeo@teikal.gr, adam@med.duth.gr, likothan@cti.gr

Introduction

A very interesting task in the field of signal analysis is the modeling of real world signals. That means finding the mathematical relationship that governs the observed variables measuring a system. Most conventional modeling techniques begin by selecting size and shape of the mathematical model, which is usually a linear model. In the sequence they use some methods to calculate the values of certain coefficients and constants required by the particular model in order to achieve the best possible fit between the observed data and the model. However the most important issue in modeling a system is finding the right size and shape of the mathematical model itself. Koza [9], [10] suggested that finding the functional form of a mathematical model can be considered as being equivalent to searching a space of possible computer programs (they can be viewed as mathematical models) for the particular computer program (model) which produces the desired output for given inputs. That is one is searching for the computer program (model) whose behavior best fits the observed data. The fittest individual computer programs can be found using the Genetic Programming evolutionary technique. In particular, the problem of modeling requires finding a mathematical function, in symbolic form, which fits given numeric data points representing some observed system. Finding such an empirical model for a system can be used also in forecasting future values of the state variables of the system.
In this paper we focus on the modeling of MEG recordings of epileptic patients. MEG recordings were obtained using a Superconductive QUantum Interference Device (SQUID) which is installed in Medical Physics Laboratory, in the General Hospital of Alexandroupolis, Hellas, model NEUROMAG-122 provided by 4-D Imaging. SQUIDs are very sensitive magnetometers, capable to detect and record biomagnetic fields of the order of 10^{-15} T, generated in the human brain due to electrical microcurrents at neural cellular level [6]. MEG recordings were digitized and stored for off-line analysis. Considering the MEG as timeseries, the problem of MEG modeling can be considered as a problem of finding a mathematical relationship that associates the value of the MEG at time t to values

of MEG at previous time intervals, $t-1$, $t-2$, and so on. GP methods are utilized, in order to optimize the nonlinear model fitting to MEG measurements. Specifically, in this paper we use a variation of traditional Genetic Programming, namely Linear Genetic Programming (LGP). LGP is a particular subset of GP wherein computer programs in population are represented as a sequence of instructions from imperative programming language or machine language. The graph-based data flow that results from a multiple usage of register contents and the existence of structurally noneffective code (introns) are two main differences to more common tree-based genetic programming (TGP) [8]. To our knowledge, this is the first time that GP is used for such kind of signal modeling. Until now there were a lot of efforts to model and predict the MEG of patients and healthy subjects but none of these efforts uses GP. So, in [3], [4], [5], and [11], an Evolutionary Neural Network with Multiple Extended Kalman Algorithm was used in order to model and forecast the behavior of MEG signals of patients suffering from epilepsy. MEG modeling and prediction, if successful, could provide information on the complexity of the underlying brain dynamics in epilepsy or any other normal or pathological condition of the Central Nervous System. This information could be of clinical interest [1].

Methods

For the MEG signal modeling the following preparations according to [10] were done:

(1) Set of terminals: As terminals were used the input variables of the data sets that were constructed using the MEG recordings, and a number of random constants. Since the MEG recording can be considered as a time series, then the problem of MEG modeling can be considered as a problem of finding a mathematical relationship that associates the value of the MEG at time t (the output) with values of MEG at previous time intervals, $t-1, t-2$, and so on. Since it isn't known how many previous time intervals are necessary in order to model better MEG a number of different previous time intervals were used varying from 2 to 7.
(2) Set of primitive functions: As functions were used the classical mathematical functions of addition, subtraction, multiplication, division (a protected version), absolute value, square root, exponential, sine and cosine.
(3) Fitness measure: As fitness measure it was used the Mean Square Error between the desired output (the observed MEG recordings) and the real output of the model.
(4) Parameters for controlling the run: as population size, M, it was used the 500 individuals, while the maximum number of generations to be run, G, it was set equal to 300.
(5) Method for designating a result and criterion for terminating a run: As method of result designation for a run we choose to designate the best

individual obtained in any generation of the population during the run (i.e., the best-so-far individual) as the result of the run. While, as a termination criterion it was used the maximum number of 300 generations that genetic programming was left to run.

In all the experiments we used the same parameter values, for comparison reasons. The MEG recordings were organized in 6 data sets according to the number of previous time intervals used (they vary from 3 to 7). In the sequence, every data set was split to three other data sets, namely training set consisting of 200 patterns, validation set consisting of 200 patterns and test set consisting of 1600 patterns. Training set was used for the training of the individuals (computer programs - models) of the population, the validation set were used to exhibit the generalization performance of the individuals, while the test set was used at the end of the whole evolutionary process in order to appraise the performance of the produced model (the output of Genetic Programming technique) on unseen data. In order to evaluate the performance of the produced computer programs three well-known error measures, the Normalized Root Mean Squared Error (NRMSE), the Correlation Coefficient (CC) and the Mean Relative Error (MRE) were used.

Table 1. Performance of the produced models on the Test Set

inputs	$NRMSE$	CC	MRE	MSE
2	0.0735	0.9973	0.3603	$3.6032 \cdot 10^{-1}$
3	0.0293	0.9996	0.1337	$5.9385 \cdot 10^{-4}$
4	0.0288	0.9996	0.1313	$5.7302 \cdot 10^{-4}$
5	0.0307	0.9995	0.1410	$6.5096 \cdot 10^{-4}$
6	0.0215	0.9998	0.0851	$3.1991 \cdot 10^{-4}$
7	0.0259	0.9997	0.1166	$4.6285 \cdot 10^{-4}$

Results and Discussion

Table 1 depicts the performance of the produced computer programs (models) on the test set for the six different cases of number of inputs used. The obtained results are in accordance to the corresponding ones obtained by non-linear analysis and chaotic methods for the analysis of the epileptic MEG [2], [7]. In that works, signal processing methods based on Complexity Theory and the Theory of chaotic dynamics of nonlinear systems were applied on MEG signals for the purpose of a better physical understanding of the underlying processes in epileptic brain dynamics. From the view of the Complexity Theory, low-dimensional non-linear dynamics were revealed to undergo the MEG of epileptic patients and the existence of low-dimensional strange attractors in the dynamics of brain function in epilepsy was justified. In addition, in [1] MEG signals of epileptic

patients were modeled using NARMAX (Nonlineal ARMA with external noise) methods and strong evidence were obtained that the nonlinear coefficients are rather weak compared to the linear coefficients of the obtained models. These results can be explained only considering the high level of synchronization and rhythmicity that appears in brain function of epileptic patients. Thus, the hypothesis that epileptic behavior is due to highly synchronized neural dynamics seems also to be supported by the present work. It is in our intentions to investigate much further the ability of GP to produce models of biological systems like the MEG and many others. For this reason we have already started to build a GP environment specialized in modeling problems.

References

1. Adamopoulos, A.V.: Intelligent Adaptive Modeling and Identification of MEG of Epileptic patients. WSEAS Transactions on Biology and Biomedicine **3(2)** (2006) 69-76
2. Adamopoulos, A. and Anninos, P.: Application of chaotic dynamics data analysis methods on the MEG of epileptic and parkinsonian patients, in order to evaluate their improvement using external magnetic fields. Biophysical Journal **64(2)** Part 2 (1993), Abstracts of 37th Annual Meeting of the Biophysical Society
3. Adamopoulos, A., Anninos, P., Likothanassis, S., Georgopoulos, E.: On the Predictability of MEG of Epileptic Patients using RBF Networks Evolved with Genetic Algorithms. Proceedings of BIOSIGNAL'98, Brno, Czech Republic, June 23-25, 1998.
4. Adamopoulos, A., Georgopoulos, E., Likothanassis, S., Anninos, P. : Forecasting the Magnetoengaphalogram (MEG) of Epileptic Patient using Genetically Optimized Neural Networks. Proceedings of Genetic and Evolutionary Computation Conference (GECCO'99), Orlando, Florida USA, July 14-17, 1999.
5. Adamopoulos A., Georgopoulos E. and Likothanassis S.: Evolutionary Neural Networks for Timeseries Prediction. WSEAS Transactions on Biology and Biomedicine **1(1)** (2004) 137-147
6. Anninos, P., Jacobson, J., Tsagas, N., Adamopoulos, A.: Spatiotemporal Stationarity of Epileptic Focal Activity Evaluated by Analyzing MagnetoEncephaloGraphic (MEG) data and the Theoretical Implications. Panminerva Medica **39** (1997) 189-201
7. Anninos, P.A., Kotini, A., Adamopoulos, A., Tsagas, N.: The use of nonlinear analysis for differentiating brain biomagnetic activity in epileptic patients before and after magnetic stimulation. Hadronic Journal Supplement **14** (1999) 1-26
8. Banzhaf W.P., Nordin P., Keller R.E., Francone F.D.: Genetic programming: An introduction. Kaufmann, San Mateo CA, 1998
9. Koza, J.R.: A genetic approach to econometric modeling. Sixth World Congress of the Econometric Society, Barcelona, Spain. August 27, 1990
10. Koza, J.R.: Genetic programming: on the programming of computers by means of natural selection. MIT Press, Cambridge MA, 1992
11. Likothanassis, S., Georgopoulos, E., Adamopoulos, A.: Structure Determination and Training of Neural Networks using Evolution Programs. Neural, Parallel and Scientific Computations Journal **8(1)** (2000) 29-48

Scalable preconditioners for hp-version discontinuous Galerkin finite element methods

Emmanuil H. Georgoulis[1] and Daniel Loghin[2]

[1] Department of Mathematics, University of Leicester, University Road, Leicester LE1 7RH, United Kingdom, Emmanuil.Georgoulis@mcs.le.ac.uk
[2] School of Mathematics, University of Birmingham, Edgbaston, Birmingham BS15 2TT, United Kingdom, d.loghin@bham.ac.uk

Abstract. Discontinuous Galerkin methods raise new challenges with regard to the solution of the ensuing linear system. Due to the nature of the discretisation, the problems can become very quickly very large, particularly when the degree of the polynomial approximation is also increased. In the case of general elliptic problems, useful iterative methods can be designed by taking into account the finite element formulation. In this work we devised a preconditioner based on equivalence to the norm inherited from the finite element space. The preconditioner is employed together with a 3-term GMRES routine in order to maintain storage at a minimum. The resulting solver was applied to the case of DGFEM discretizations of problems with non-negative characteristic form. Theoretical results were derived to explain the convergence behaviour.

Summary

Recent years have seen an increasing interest in a class of non-conforming finite element approximations of elliptic boundary-value problems with hyperbolic nature, usually referred to as *discontinuous Galerkin finite element methods*. Various families of discontinuous Galerkin finite element methods (DGFEMs) have been proposed, particularly for the numerical solution of convection-diffusion problems, due to the many attractive properties they exhibit. In particular, DGFEMs admit good stability properties, they offer flexibility in the mesh design (meshes containing hanging nodes are admissible) and in the imposition of boundary conditions (Dirichlet boundary conditions are weakly imposed), and they are increasingly popular in the context of *hp*-adaptive algorithms. The increase in popularity for DGFEMs has created a corresponding demand for developing linear solvers.

Existing approaches to solving systems arising in DGFEMs include domain decomposition, either non-overlapping [10], [11] or overlapping [21] and multigrid [18], [4]. Another favoured approach consists in reformulating the problem as a system of PDEs which is then solved using a mixed finite element method for which block-preconditioners can be devised [27], [22], [20].

Remarkably, preconditioned Krylov methods feature rarely and have been mostly employed for the case of time-dependent problems, whether using time-

discontinuous discretizations [8], or space-discontinuous ones [3]. One of the reasons for this notable absence may be explained through the ill-conditioning the resulting linear systems suffer from [5] coupled with a need for preconditioner design.

In this work, we extend results available for the case of standard finite element methods to the *hp*-discontinuous case. In particular, it is now known that the norms associated with the finite element analysis of convection-diffusion problems can be employed as preconditioners. Here, we extend this idea to the case of *hp*-version DGFEMs who are known to given rise to ill-conditioned stiffness matrices with block structure

More precisely, we construct scalable preconditioning strategies (in the sense that the number of iterations in a preconditioned GMRES algorithm is independent of the mesh size and the local elemental polynomial degree) for the *hp*-version DGFEM. Moreover, the preconditioner is employed together with a 3-term GMRES routine in order to maintain storage at a minimum. The resulting solver was applied to the case of DGFEM discretizations of the wide family of boundary-value problems with non-negative characteristic form. Theoretical results were derived to explain the convergence behaviour accompanied by numerical experiments that are in complete agreement with the theoretical predictions.

We shall also present some recent results and extensions of the ideas found in Georgoulis and Loghin [15] when anisotropic meshes are employed to resolve possible boundary layers. Some of the presentation will be inspired by the work [15], but recent development and extensions of these results will be presented.

Numerical Example

The validity of our theoretical results providing scalable (i.e., GMRES iteration counts independent of the mesh-size and the local polynomial degrees) preconditionining strategies for DGFEM-discretisations of convection-diffusion problems is illustrated by the following numerical experiment.

We solved

$$-\epsilon\Delta u + \mathbf{b}\cdot\nabla u = f \quad \text{for } (x,y)\in(0,1)^2,$$

with $\mathbf{b} = (1,1)$, subject to a Dirichlet boundary condition, which, along with the forcing function f, is chosen so that the analytical solution is

$$u(x,y) = x + y(1-x) + \frac{\mathrm{e}^{-\frac{1}{\epsilon}} - \mathrm{e}^{-\frac{(1-x)(1-y)}{\epsilon}}}{1-\mathrm{e}^{-\frac{1}{\epsilon}}}.$$

This problem was considered in [19] (Example 3). The solution exhibits boundary layer behaviour along $x = 1$ and $y = 1$, and the layers become steeper as $\epsilon \to 0$.

We solved the problem for a range of ϵ. Discretisations for a range of uniform h (meshsize) and p (degree of polynomial approximation) were employed. We denote by n the number of unknowns of the the linear system. The results are presented Table 1.

Table 1. GMRES iterations for DGFEM-discretisation of the convection diffusion problem using the proposed preconditioning strategy (left table) and the corresponding results using the standard ILU(10^{-2})-preconditioning strategy(right table).

p	h	n	$\epsilon = 0.5$	$\epsilon = 0.1$	$\epsilon = 0.05$	$\epsilon = 0.01$
1	0.04	2,500	7	15	22	77
	0.02	10,000	7	15	22	80
	0.01	40,000	7	14	22	80
2	0.04	5,625	7	14	22	80
	0.02	22,500	6	14	22	80
	0.01	90,000	6	14	21	78
3	0.04	10,000	6	14	22	79
	0.02	40,000	6	14	22	78
	0.01	160,000	6	13	21	78

p	n	$\epsilon = 0.5$	$\epsilon = 0.1$	$\epsilon = 0.01$
1	2,500	12	13	7
	10,000	36	40	29
	40,000	124	117	69
2	5,625	18	17	12
	22,500	61	59	60
	90,000	235	231	137
3	10,000	39	29	23
	40,000	112	114	100
	160,000	> 300	> 300	> 300

As predicted by theory, the number of iterations is independent of discretisation parameters.

References

1. M. Arioli, D. Loghin, and A. J. Wathen, *Stopping criteria for iterations in finite element methods.*, Numer. Math., 99 (2005), pp. 381–410.
2. D. N. Arnold, *An interior penalty finite element method with discontinuous elements*, SIAM J. Numer. Anal., 19 (1982), pp. 742–760.
3. F. Bassi and S. Rebay, *GMRES discontinuous Galerkin solution of the compressible Navier-Stokes equations*, in Discontinuous Galerkin methods (Newport, RI, 1999), vol. 11 of Lect. Notes Comput. Sci. Eng., Springer, Berlin, 2000, pp. 197–208.
4. S. C. Brenner and J. Zhao, *Convergence of multigrid algorithms for interior penalty methods*, Appl. Numer. Anal. Comput. Math., 2 (2005), pp. 3–18.
5. P. Castillo, *Performance of discontinuous Galerkin methods for elliptic PDEs*, SIAM J. Sci. Comput., 24 (2002), pp. 524–547 (electronic).
6. P. Concus and G. H. Golub, *A generalized conjugate gradient method for nonsymmetric systems of linear equations*, in Proc. Second Internat. Symp. on Computing Methods in Applied Sciences and Engineering, R. Glowinski and J. L. Lions, eds., vol. 134 of Lecture Notes in Economics and Mathematical Systems, Berlin, 1976, Springer Verlag.
7. H. C. Elman, *Iterative methods for large sparse non-symmetric systems of linear equations*, PhD thesis, Yale University, New Haven, 1982.
8. D. J. Estep and R. W. Freund, *Using Krylov-subspace iterations in discontinuous Galerkin methods for nonlinear reaction-diffusion systems*, in Discontinuous Galerkin Methods, B. Cockburn, G. E. Karniadakis, and C.-W. Shu, eds., Springer-Verlag, Berlin, 2000, pp. 327–335.
9. V. Faber, T. A. Manteuffel, and S. V. Parter, *On the theory of equivalent operators and application to the numerical solution of uniformly elliptic partial differential equations*, Adv. Appl. Math., 11 (1990), pp. 109–163.

10. X. Feng and O. A. Karakashian, *Two-level additive Schwarz methods for a discontinuous Galerkin approximation of second order elliptic problems*, SIAM J. Numer. Anal., 39 (2001), pp. 1343–1365 (electronic).
11. ———, *Two-level non-overlapping Schwarz preconditioners for a discontinuous Galerkin approximation of the biharmonic equation*, J. Sci. Comput., 22/23 (2005), pp. 289–314.
12. E. H. Georgoulis, *Discontinuous Galerkin methods on shape-regular and anisotropic meshes*, D.Phil. Thesis, University of Oxford, (2003).
13. ———, *hp-version interior penalty discontinuous Galerkin finite element methods on anisotropic meshes.*, International Journal of Numerical Analysis and Modeling, 3 (2006), pp. 52–79.
14. E. H. Georgoulis and A. Lasis, *A note on the design of hp-version interior penalty discontinuous Galerkin finite element methods for degenerate problems.*, IMA J. Numer. Anal., 26 (2006), pp. 381–390.
15. E. H. Georgoulis and D. Loghin, *Norm preconditioners for discontinuous hp-finite element methods*. SIAM J. of Sci. Comput., in review 2007.
16. E. H. Georgoulis and D. Loghin, *A multilevel domain decomposition method for discontinuous Galerkin methods*. in preparation, 2007.
17. E. H. Georgoulis and E. Süli, *Optimal error estimates for the hp–version interior penalty discontinuous Galerkin finite element method*, IMA J. Numer. Anal., 25 (2005), pp. 205–220.
18. J. Gopalakrishnan and G. Kanschat, *A multilevel discontinuous Galerkin method*, Numer. Math., 95 (2003), pp. 527–550.
19. P. Houston, C. Schwab, and E. Süli, *Discontinuous hp-finite element methods for advection-diffusion-reaction problems*, SIAM J. Numer. Anal., 39 (2002), pp. 2133–2163 (electronic).
20. G. Kanschat, *Preconditioning methods for local discontinuous Galerkin discretizations*, SIAM J. Sci. Comput., 25 (2003), pp. 815–831.
21. C. Lasser and A. Toselli, *An overlapping domain decomposition preconditioner for a class of discontinuous Galerkin approximations of advection-diffusion problems*, Math. Comp., 72 (2003), pp. 1215–1238.
22. R. D. Lazarov, S. Z. Tomov, and P. S. Vassilevski, *Interior penalty discontinuous approximations of elliptic problems*, Comput. Methods Appl. Math., 1 (2001), pp. 367–382.
23. D. Loghin, D. Ruiz, and A. Touhami, *Adaptive preconditioners for nonlinear systems of equations*, J. Comp. Applied Math., (2006), pp. 362–374.
24. D. Loghin and A. J. Wathen, *Analysis of block preconditioners for saddle-point problems*, SIAM J. Sci. Comput., 25 (2004), pp. 2029–2049.
25. O. A. Oleĭnik and E. V. Radkevič, *Second order equations with nonnegative characteristic form*, Plenum Press, New York, 1973. Translated from the Russian by Paul C. Fife.
26. B. Rivière, M. F. Wheeler, and V. Girault, *Improved energy estimates for interior penalty, constrained and discontinuous Galerkin methods for elliptic problems. I*, Comput. Geosci., 3 (1999), pp. 337–360.
27. T. Rusten, P. S. Vassilevski, and R. Winther, *Interior penalty preconditioners for mixed finite element approximations of elliptic problems*, Math. Comp., 65 (1996), pp. 447–466.
28. Y. Saad, *Iterative Methods for Sparse Linear Systems*, PWS Publishing Co., Boston, 1996.

Perturbed Newton method for Unconstrained Optimization

T.N. Grapsa[1], G.E. Antonelou[1], and A.E. Kostopoulos[1]

Division of Computational Mathematics & Informatics, Department of Mathematics, University of Patras, GR–261.10 Patras, Greece,
grapsa@math.upatras.gr, antonelougeorgia@gmail.com, arkostop@math.upatras.gr

Abstract. A perturbed Newton method for unconstrained optimization problem is proposed in this paper. Proper perturbed parameters are defined at each iteration to perturb the Hessian matrix in Newton's method in order generally to accelerate it. The idea behind the choice of these parameters is to use information of a properly selected component of the gradient of the objective function in addition to information of some norm of a column-vector of the Hessian matrix. The quadratic convergence of the produced new algorithm can be proved. Preliminary results from the application of the new method on various well known test functions are promising.

Introduction

Consider the general unconstrained optimization problem

$$\min_{x\in\mathbb{R}^n} f(x), \tag{1}$$

where $f : \mathbb{R}^n \to \mathbb{R}$ is assumed to be a continuously differentiable function and $x = (x_1, x_2, \ldots, x_n)^T \in \mathbb{R}^n$. As it is known, all the local minimizers of the objective function f are stationary points. Thus, at these points the gradient $\nabla f(x) = G(x) = (g_1(x), g_2(x), \ldots, g_n(x))^T$ vanishes :

$$\nabla f(x) = G(x) = 0 \tag{2}$$

If the Hessian matrix $H(x)$ is symmetric and positive definite then solving the problem (1) is equivalent to solving the problem (2). There are several numerical methods for solving this problem [1, 3–6]. Among others, Newton's algorithm, given by : $x_{p+1} = x_p - H(x_p)^{-1}G(x_p), p = 0, 1, 2, \ldots$, has been shown as a successful algorithm for solving optimization problems. For this reason, we remain in Newton's method and try to accelerate it, without increasing the computational cost per iteration, by applying a new strategy on it. In particular, the goal of our study is to develop a robust perturbed Hessian matrix to substitute the trivial one in Newton's method. In this sense, the new method may be viewed as a new quasi-Newton method. Moreover, it is known that the perturbation of the Hessian matrix in Newton's method is part and parcel with the rotation of tangent

planes at the gradient surfaces. Thus, the new perturbed Hessian matrix, from a geometric point of view, may be considered as a rotation of tangent planes at the gradient surfaces which results in removing the current approximation point closer to the solution one. Therefore, an efficient Hessian may join to accelerate Newton's method and at the same time enlarge the convergence area [2]. The challenge that arises now is how to select properly such parameters to modify Newton's Hessian. Inspired by the importance of a properly selected component of the gradient of the objective function and of some norm of a column-vector of the Hessian matrix, we produce the perturbed parameters accordingly to such information. Thus, an efficient perturbed Newton algorithm is proposed.

Moreover, the proposed algorithm can also be considered as a new Newton method applied to a proper system that is equivalent to the original one given by (2) and thus its quadratic convergence is evident.

We now present all the above more formally. So, we consider the mapping $W = (w_1, w_2, \ldots, w_n)^T : D \subset \mathbb{R}^n \to \mathbb{R}^n$, with

$$w_i(x) = g_i(x) + \sum_{j=1}^{n} t_j x_j, i = 1, 2, \ldots, n \tag{3}$$

where $T = (t_1, t_2, \ldots, t_n)^T$ is the vector of the perturbed parameters $t_j, j = 1, 2, \ldots, n$ and the inner product $\langle x, T\rangle = 0$, i.e. the $\sum_{j=1}^{n} t_j x_j = 0$. If $x_p = (x_1^p, x_2^p, \ldots, x_n^p)^T$ is the approximation of the solution at the p-th iteration, $p = 0, 1, \ldots$, $G'(x) = H(x)$ is the Jacobian matrix of G and $G'_{ij} = H_{ij} = (\partial_j g_1, \partial_j g_2, \cdots, \partial_j g_n)^T$ is the j-th column-vector of the Jacobian matrix of G, then the perturbed parameters are given by

$$t_j = \begin{cases} \frac{g_j(x_p)}{||G'_{ij}(x_p)||_2^2}, & \text{for } j = 1, \ldots, n-1 \\ -\frac{\sum_{k=1}^{n-1} t_k x_k^p}{x_n^p}, & \text{for } j = n \text{ and } x_n^p \neq 0 \end{cases} \tag{4}$$

The application of Newton's method in the new system of the equations $w_i(x) = 0, i = 1, 2, \ldots, n$, results in our new iterative scheme given by

$$x_{p+1} = x_p - W'(x_p)^{-1} W(x_p), \qquad p = 0, 1, 2, \ldots \tag{5}$$

where, because of (3), the Jacobian matrix of the new function $W(x)$ is given by

$$W'(x_p) = G'(x_p) + \Xi = H(x_p) + \Xi, \quad p = 0, 1, 2, \ldots \tag{6}$$

where $G'(x_p)$ is the Jacobian matrix of $G(x)$ at the current point x_p, and Ξ is the rank-1 $n \times n$ matrix with $\Xi_{ij} = t_j, j = 1, \ldots, n$, for all $i = 1, \ldots, n$.

Due to the iterative form (5), our algorithm is a new Newton's method and therefore its quadratic convergence is evident.

Taking into account the original system (2), the relation $w_i(x_p) = g_i(x_p), i = 1, \ldots, n$, $p = 0, 1, \ldots$ (due to (3) and the fact that $\langle x, T\rangle = 0$) and (6), our iterative scheme (5) may also be written in the form

$$x_{p+1} = x_p - (G'(x_p) + \Xi)^{-1} G(x_p) = x_p - (H(x_p) + \Xi)^{-1} G(x_p), \quad p = 0, 1, \ldots \tag{7}$$

In the form (7) our new method can also be considered as a new quasi-Newton method. Also, with an appropriate procedure similar to the one presented in [3], a new perturbation to the proposed Hessian matrix $W'(x)$ may be applied in order to transform it into a symmetric one as well as into a diagonal one. The quadratic convergence of the last procedure may also be proved.

Numerical Results

The proposed algorithm has been implemented using a new Fortran program, named PN_OPT (Perturbed Newton OPTimization). PN_OPT was compared with well known test functions. The preliminary results are quite satisfactory. The reported parameters in the Tables 1-6 indicate : IT the total number of iterations required to obtain x^*, FE the total number of function evaluations(including derivatives) and D indicates divergence or nonconvergence.

First, we give the Tables 1, 2 and 3 where we compare our new method given by (7) with the well known methods: Armijo's steepest descent method, Fletcher-Reeves[FR], Polak-Ribiere[PR] and Broyden-Fletcher-Goldfarb-Shanno[BFGS] with our new method Perturbed Newton's Method[PN_OPT]. Next, we give Table 4 comparing (7) with the trivial Newton method. In all cases the accuracy is $eps = 10^{-12}$. Due to lack of space, the interested reader is encouraged to see the full version of the paper [7] for more examples.

Table 1. Brown's Almost Linear Function (see Example 1)

			Armijo		FR		PR		$BFGS$		PN_OPT	
x_1^0	x_2^0	x_3^0	IT	FE	IT	FE	IT	FE	IT	FE	IT	FE
0	0	3	221	1612	27	389	9	143	9	131	1	12
-1	0	3	233	1691	34	491	42	612	12	180	6	72
0.8	0.7	-2	193	1446	70	996	23	336	14	216	10	120

Table 2. Freudenstein & Roth's Function (see Example 2)

		Armijo		FR		PR		$BFGS$		PN_OPT	
x_1^0	x_2^0	IT	FE	IT	FE	IT	FE	IT	FE	IT	FE
0.5	1000	1380	18770	D	D	D	D	D	D	33	198
4	-1000	1886	25597	D	D	D	D	D	D	34	204
12	2	2027	26886	70	1145	8	130	7	103	5	30

Example 1 (Brown's almost-linear function [5, 2]). In this case the objective function f is given by $f(x) = \sum_{i=1}^{3} {f_i}^2(x)$, where, in general, $f_i(x) = x_i + \sum_{j=1}^{n} x_j - (n+1), 1 \leq i < n$ and $f_n(x) = \prod_{j=1}^{n} x_j - 1$. In our case n=3.

Table 3. Rosenbrock's Function (see Example 3)

		Armijo		FR		PR		$BFGS$		PN_OPT	
x_1^0	x_2^0	IT	FE	IT	FE	IT	FE	IT	FE	IT	FE
10	10	18416	251611	310	7469	26	526	32	505	7	42
100	100	D	D	D	D	33	746	54	822	7	42
-2000	-2000	2542	35743	D	D	93	2466	173	2667	9	54

Example 2 (Freudenstein and Roth's function [5, 2]). In this example the objective function f is given by $f(x) = \sum_{i=1}^{2} {f_i}^2(x)$, where $f_1(x) = -13 + x_1 + ((5 - x_2)\, x_2 - 2)\, x_2$ and $f_2(x) = -29 + x_1 + ((x_2 + 1)\, x_2 - 14)\, x_2$.

Example 3 (Rosenbrock's function [5, 2]). In this example the objective function f is given by $f(x) = \sum_{i=1}^{2} {f_i}^2(x)$, where $f_1(x) = 10\left(x_2 - x_1^2\right)$ and $f_2(x) = 1 - x_1$.

Example 4 (Beale's function [5, 2]). In this example the objective function f is given by $f(x) = \sum_{i=1}^{2} {f_i}^2(x)$, where $f_1(x) = 1.5 - x_1(1 - x_2)$ and $f_2(x) = 2.25 - x_1(1 - x_2^2)$.

Table 4. Beale's Function (see Example 4)

		Newton		PN_OPT	
x_1^0	x_2^0	IT	FE	IT	FE
-2	-100	61	366	38	228
0.1	0.1	40	240	23	138
-1	20	51	306	42	252

References

1. J.E. Dennis Jr. and R.B. Schnabel, *Numerical Methods for Unconstrained Optimimization and Nonlinear Equations*, Prentice-Hall, Englewood Cliffs, NJ, 1983.
2. T.N. Grapsa, *A Quadratic Convergence quasi-Newton Method for Unconstrained Optimization*, accepted for presentation in the conference *" Optimization2007 "*, Porto, Portugal, 2007.
3. T.N. Grapsa, M.N. Vrahatis, *A dimension-reduction method for unconstrained optimization*, Journal of Computational and Applied Mathematics, 66 (1996) 239-253.
4. Jose Mario Martinez , *Practical quasi-Newton methods for solving nonlinear systems*, J.C.A.M., Issue 1-2, 124 (2000) 97-121.
5. B.J. More, B.S. Garbow and K.E. Hillstrom, *Testing Unconstrained Optimization Software*, ACM Trans. Math. Software 7 (1981) 17-41.
6. J.M. Ortega and W.C. Rheinbolt, *Iteration Solution of Nonlinear Equations in Several Variables*, (Academic Press, New York, 1970).
7. T.N. Grapsa, G.E. Antonelou and A.E. Kostopoulos, *Perturbed Newton method for Unconstrained Optimization*, `www.math.upatras.gr/~grapsa/NumAnOPT07.pdf`.

Optimal Stationary One- and Two-Parameter ADI Preconditioners for Conjugate Gradient Methods*

Michael Lapidakis**

Department of Mathematics
University of Crete
GR-714 09 Heraklion
Greece
mihalis@math.uoc.gr

Abstract. We determine optimal Stationary One- and Two-Paremeter Alternating Direction Implicit (ADI) Preconditioners for Conjugate Gradient Methods when applied for the solution of a model problem second order elliptic PDE. The PDE is approximated by 5− and 9−point stencils. It is proved that the problem of determining the best ADI Preconditioner is equivalent to that of determining the optimal Extrapolated(E) ADI method. Analytic expressions for the optimal acceleration and extrapolation parameters are found for both discretizations. Numerical examples show that the ADI preconditioners we propose are very competitive compared to other well-known ones.

* It is a joint work with A. Hadjidimos (hadjidim@inf.uth.gr). Part of the work of this author was funded by the Program Pythagoras of the Greek Ministry of Education. Department of Computer and Communication Engineering, University of Thessaly, 10 Iasonos Street, GR-383 33 Volos, Greece.

** The work of the author was done under a scholarship from the Greek State Scholarships Foundation.

The Optimum Preconditioned Simultaneous Displacement Method for 2-cyclic matrices

M.A. Louka, N.M. Missirlis * and F.I. Tzaferis

Department of Informatics and Telecommunications,
University of Athens,
Panepistimiopolis, 15784, Athens, Greece
{ mlouka, nmis, ftzaf }@di.uoa.gr
http://parallel.di.uoa.gr

Introduction

In [1], [4] the Preconditioned Simultaneous Displacement (PSD) iterative method, was introduced and studied for the numerical solution of the linear system

$$Au = b \tag{1}$$

where A is a $N \times N$ positive definite and consistently ordered matrix and u, b are $N \times 1$ vectors. PSD is a first order extrapolation of SSOR and as such it was shown to be asymptotically twice as fast as SSOR for the natural ordering [1]. In [4] various accelerated techniques were applied to PSD, which increased its rate of convergence by an order of magnitude. Comparisons with the SOR method proved that PSD combined with Semi-Iterative(SI) techniques is even better than SOR, in certain cases [4]. However, all these results were based on "good" (near the optimum) values of the involved parameters as, at the time, there was not known any functional relationship analogous to the SOR one connecting the eigenvalues of the PSD iteration matrix to the ones of the Jacobi matrix, except in the red-black case [5]. By the time that such a functional relationship was shown to exist for SSOR [10] it also exists for PSD and the theory concerning p-cyclic matrices (see e.g. [2], [8]) can be extended for the PSD method also. It is the purpose of the present work to proceed towards this direction and study the convergence of the PSD method in case the block Jacobi iteration matrix is weakly 2-cyclic [9] and possesses either real or imaginary eigenvalues.
Let us consider the linear system (1), where A is a $p \times p$ matrix partitioned into block form

$$A = \left[A_{i,1}, A_{i,2}, \ldots, A_{i,p} \right], \quad i = 1, 2, \cdots p, \quad p \geq 2 \tag{2}$$

and each diagonal $A_{i,i}$ is square and nonsingular. Let the coefficient matrix A be splitted as

$$A = D - C_L - C_U, \tag{3}$$

* This author's research was carried out while on leave of absence at the Department of Informatics, University of Cyprus, Cyprus.

so that $D := diag(A_{1,1}, A_{2,2}, \cdots, A_{p,p})$ and $-C_L,\ -C_U$ are the strictly lower and upper triangular parts of A, respectively. The associated block Jacobi matrix is defined by

$$B := L + U, \tag{4}$$

where $L = D^{-1}C_L,\ \ U = D^{-1}C_U$. The PSD method is given by the following scheme [1], [5]

$$u^{(n+1)} = \mathcal{D}_{\tau,\omega}u^{(n)} + \delta_{\tau,\omega}, \tag{5}$$

where

$$\mathcal{D}_{\tau,\omega} := I - \tau\mathcal{B}_\omega, \qquad \mathcal{B}_\omega := (I - \omega U)^{-1}(I - \omega L)^{-1}D^{-1}A \tag{6}$$

and

$$\delta_{\tau,\omega} := \tau(I - \omega U)^{-1}(I - \omega L)^{-1}D^{-1}b, \tag{7}$$

where $\tau \neq 0,\ \omega \in \Re$. Note that if $\tau = \omega(2 - \omega)$, then (5) reduces to the well known SSOR method. Let B be a weakly cyclic matrix of index p [9].
Working as in [10] we can prove the following

Theorem 1 *Assume that the block-partitioned matrix A of (2) is such that all diagonal submatrices $A_{i,i}$ are square and non-singular, $1 \leq i \leq p$, and B of (4) is a weakly cyclic of index p block matrix. If $\omega \neq 0, 2$, if $\lambda \neq 1$ is an eigenvalue of $\mathcal{B}_\omega$, and if μ satisfies*

$$(1 - \lambda)^p = (1 - \lambda\omega)^{p-2}[1 - \lambda\omega(2 - \omega)]\mu^p, \tag{8}$$

then μ is an eigenvalue of the block Jacobi matrix B. Conversely, if μ is an eigenvalue of B and if $\hat{\lambda} \neq 1$ satisfies (8), then $\hat{\lambda}$ is an eigenvalue of $\mathcal{B}_\omega$.

If ν is an eigenvalue of $\mathcal{D}_{\tau,\omega}$, then because of (6) we have

$$\nu = 1 - \tau\lambda. \tag{9}$$

Expressing (8) in terms of ν, yields

$$(\nu + \tau - 1)^p = \tau(\tau - \omega + \omega\nu)^{p-2}[\tau - \omega(2 - \omega)(1 - \nu)]\mu^p. \tag{10}$$

The above functional equation relates the eigenvalues of $D_{\tau,\omega}$ and those of the block Jacobi matrix B. The paper is organized as follows. In Section 3 we derive sufficient and necessary conditions for PSD to converge under the assumption that the eigenvalues of the Jacobi iteration matrix are either all real or all imaginary. Under the same assumptions we find the optimum values of the involved parameters in Section 4. In Section 5 we compare PSD with SSOR. Our numerical results are presented in Section 6. Finally, we discuss our conclusions in Section 5.

Final remarks and Conclusions

In this paper we studied the rate of convergence of the iterative methods PSD, SSOR and PJ under the assumptions that the coefficient matrix A is two-cyclic and the Jacobi iteration matrix B possesses either real or imaginary eigenvalues. Applying the results of [6], [7] we were able to find sufficient and necessary conditions for the aforementioned iterative schemes to converge. Additionally, we determine the optimum values of the parameters involved such that the methods attain their optimum rate of convergence. The conclusions from our analysis are: (i) PSD attains a faster rate of convergence than SSOR and PJ methods and (ii) the rate of convergence of the PSD method is similar to the Extrapolated Gauss-Seidel method if B possesses real eigenvalues and is equal to the ESOR one if B possesses imaginary eigenvalues.
The problem of using complex parameters τ, ω with $\sigma(B)$ belonging to a compact subset Σ of the complex plane $\mathbb{C}$ is open for the PSD method. This problem has been solved recently in [3] for the SOR method under the assumption that Σ is symmetric with respect to the origin. SSOR shares the same functional eigenvalue relationship with SOR, where now instead of ω we have $\hat{\omega} = \omega(2 - \omega)$, hence the theory in [3] applies to SSOR also. Since PSD is a first order extrapolation scheme of SSOR it follows that a similar theory will also hold. It is therefore conjectured that in case the outer boundary of Σ is not an ellipse, then Semi-Iterative PSD will be better than simple PSD and the optimum value of ω will be unity. Alternatively, if the outer boundary of Σ is an ellipse, symmetric with respect to the origin, then it is conjectured that PSD, as SOR (and SSOR) [3], will attain its maximum convergence for complex optimum ω, τ.

Acknowledgement

The project is co-funded by the European Social Fund and National Resources (EPEAK II) Pythagoras, grant no. 70/3/7418.

References

1. D. J. Evans, N. M. Missirlis, The preconditioned simultaneous displacement method (PSD method) for elliptic difference equations, Math. Comput. Simulation 22 (1980) 256-263.
2. A. Hadjidimos and M. Neumann, Precise Domains of Convergence for the block SSOR method associated with p-cyclic matrices, BIT 29 (1989) 311-320.
3. A. Hadjidimos and N. S. Stylianopoulos, Optimal Semi-Iterative methods for complex SOR with results from potential theory, Numer. Math. (2006) 35-52.
4. N. M. Missirlis and D. J. Evans, On the acceleration of the Preconditioned Simultaneous Displacement method, Math. Comput. Simulation 23, (1981) 191-198.
5. N. M. Missirlis and D. J. Evans, The Modified Preconditioned Simultaneous Displacement (MPSD) method, Math. Comput. Simulation 26, (1984) 257-262.
6. N. M. Missirlis, Convergence Theory of Extrapolated Iterative Methods for a Certain Class of Non-Symmetric Linear Systems, Numer. Math. 45,(1984) 447-458.

7. N. M. Missirlis, Iterative Methods for Sparse Linear Systems: Some recent developments, Sparsity and its Applications, ed. D. J. Evans, Cambridge Univ. Press, (1985) 113-135.
8. Y. G. Saridakis, On the Analysis of the Unsymmetric Overrelaxation Method when Applied to p-cyclic Matrices, Numer. Math. 49 (1986) 461-473.
9. R. S. Varga, Matrix Iterative Analysis, Prentice-Hall, Inc. Englewood Cliffs, N.J., (1962).
10. R.S. Varga, W. Niethammer, and D.-Y. Cai, p-cyclic matrices and the Symmetric Successive Overrelaxation method, Linear Algebra Appl. 58 (1984) 425-439.
11. D. M. Young, Iterative Solution of Lerge Linear Systems, Academic Press, New York, 1971.

Improved Newton's method without direct function evaluations

E.N. Malihoutsaki[1], I.A. Nikas[1], and T.N. Grapsa[1⋆]

[1] Division of Computational Mathematics and Informatics, Department of Mathematics, University of Patras, GR–26500 Patras, Greece,
malihoutsaki_eri@yahoo.gr, nikas@math.upatras.gr, grapsa@math.upatras.gr

Abstract. Several methods have been proposed to solve systems of nonlinear equations. Among them, Newton's method holds a prominent position. Recently, we have proposed a Newton's method to manage problems with inaccurate function values or problems with high computational cost. Due to the existence of many such problems in real-life applications and the promising results of the above method, we remain in this goal and introduce a new improved version of it. For this reason, we alter the above method such that, on one hand, to accelerate it and moreover to reduce its computational cost, by requiring even less information per iteration and, on the other hand, holding its important advantages. These are its quadratic convergence, its good behavior in singular and ill-conditioned cases of Jacobian matrix and, of course, its capability to be ideal for imprecise function problems. The efficiency of the new method is demonstrated by numerical applications.

Introduction

We consider nonlinear systems of equations

$$F(x) = 0 \tag{1}$$

where $F : \mathcal{D} \subset \mathbb{R}^n \to \mathbb{R}^n$ is continuously differentiable on an open neighborhood $\mathcal{D}^* \subset \mathcal{D}$ of a solution $x^* = (x_1^*, \ldots, x_n^*) \in \mathcal{D}$ of the system (1). We denote $F = (f_1, \ldots, f_n)$ and by $F'(x)$ the Jacobian matrix of F for all $x = (x_1, \ldots, x_n) \in \mathbb{R}^n$.

Newton's method is the most widely used algorithm for solving systems of nonlinear equations [3, 8, 9], given by:

$$x^{p+1} = x^p - F'(x^p)^{-1} F(x^p), \quad p = 0, 1, \ldots. \tag{2}$$

where $x^p = (x_1^p, x_2^p, \ldots, x_n^p)$ is the current approximation and x^{p+1} is the next approximation.

In Newton's method and its variants a main drawback is the computational cost due to the function and derivative evaluations. On the other hand, there are

⋆ Corresponding author: e-mail: grapsa@math.upatras.gr, Phone: +30 2610 997332, Fax: +30 2610 992965

many cases where exact function or derivative values are not available. So, it is important to find methods, which are free of function or derivative evaluations, in order to reduce the total computational cost or to work out problems with imprecise function or derivative values. There are several methods in bibliography, which are derivative free [2, 3]. However, we have detected only one method free of function values, but it is only for polynomial equations [1] and furthermore there are some other methods ideal for problems with imprecise function values [6, 7, 11].

We have recently proposed a method in [4] to make Newton's method ideal when accurate function values are not available or their computational cost is high. Thus, in [4] the function values in Newton's method are not directly evaluated from the corresponding functions f_i, $i = 1, 2, \ldots, n$, but they issue by some appropriate approximated expressions of them. To succeed it, we have taken advantage of the proper selected pivot points $x_{pivot}^{p,i} = (x_1^p, x_2^p, \ldots, x_{n-1}^p, x_n^{p,i}), \quad i = 0, 1, 2, \ldots, n, \quad p = 0, 1, \ldots$, which we have already defined in [4, 5], where randomly we have selected to differ from the current point x^p at the $n-$th component, and which are extracted via a sign-function-based technique [4–7, 10, 11]. The iterative form of this method, named *WFEN (Without Function Evaluations Newton)* method, has been given by

$$x^{p+1} = x^p - F'(x^p)^{-1}\partial_n f_i(x_{pivot}^{p,i})(x_n^p - x_n^{p,i}), i = 1, 2, \ldots, n, p = 0, 1, \ldots. \quad (3)$$

In this paper, due to the contribution of the above method we reissue to improve it, in the sense of reducing its computational cost and accelerating it. Thus, a new Newton method is proposed, also without requiring the explicit evaluation of functions and so that to be applicable to imprecise problems. The key point is to define new quantities to approximate function values in Newton's method with even less cost than the corresponding one in the previous work [4], given by the relation (3). Working in a similar way, as in [4], using proper Taylor's series and utilizing the proposed pivot points, we produce the new method, named *IWFEN (Improved Without Function Evaluations Newton)* method, given by:

$$x^{p+1} = x^p - F'(x^p)^{-1}\partial_n f_i(x^p)(x_n^p - x_n^{p,i}), i = 1, 2, \ldots, n, p = 0, 1, \ldots. \quad (4)$$

Note 1. Comparing the schemes (3) and (4), it is easy to notice that in (4) we need n less partial derivatives evaluations per iteration contrary to (3), because the $\partial_n f_i(x^p)$, $i = 1, 2, \ldots, n$, $p = 0, 1, \ldots$ have already been evaluated in the Jacobian matrix $F'(x^p)$. Moreover, it is important to point out that the new method, while it works well for imprecise problems, it also works well even in singular cases or in cases of an ill-conditioned Jacobian matrix.

From another point of view, we consider the mapping $L = (l_1, l_2, \ldots, l_n)^T : \mathcal{D} \subset \mathbb{R}^n \to \mathbb{R}^n$, by

$$l_i(x) = (x_n^p - x_n^{p,i})\partial_n f_i(x^p) + \sum_{j=1}^{n}(x_j - x_j^p)\partial_j f_i(x^p), i = 1, 2, \ldots, n, p = 0, 1, \ldots (5)$$

It may be proven, that $l_i(x^p) = \partial_n f_i(x^p)(x_n^p - x_n^{p,i})$, $i = 1, 2, \ldots, n$, $p = 1, 2, \ldots$ and since $L'(x) = F'(x)$ we have that $L'(x^p) = F'(x^p)$, $p = 1, 2, \ldots$. Hence, applying Newton's method to the approximated system $l_i(x) = 0$, our new method may also be written in the form

$$x^{p+1} = x^p - L'(x^p)^{-1} L(x^p), \quad p = 0, 1, \ldots. \tag{6}$$

According to the scheme (6), the new proposed method is a new Newton method on the function L and therefore, its quadratic convergence is evident.

Numerical Applications

We apply the proposed method to random problems, given in [4, 7], with nonsingular, singular and ill-conditioned Jacobian matrices. The algorithms have been implemented, using two FORTRAN programs: a program named *WFEN* for the scheme (3), which has been proposed in [4] and a new program named *IWFEN* for our new iterative scheme, given by (4).

The numerical results are quite satisfactory and the new method is similar or superior to Newton's one.

In Tables 1,2,3 we present the results obtained by Newton's method and the Schemes (3) and (4), with accuracy $\varepsilon = 10^{-14}$, for three examples, given in [4]. The *'IT'* indicates the number of the iterations, *'FE'* the number of the function evaluations (including derivatives), the *'AS'* the total number of algebraic signs for the computation of the $n-$th component, $x_n^{p,i}$, of the pivot points $x_{pivot}^{p,i}$, $i = 1, 2, \ldots, n$, $p = 0, 1, \ldots$ and *'r_i'* the root to which each method converges.

Table 1. Comparison between WFEN, IWFEN and Newton's method for Example 1

			Newton			*WFEN*				*IWFEN*			
x_1^0	x_2^0	x_3^0	*IT*	*FE*	r_i	*IT*	*FE*	*AS*	r_i	*IT*	*FE*	*AS*	r_i
0.4	0.5	0.5	53	636	r_2	20	240	600	r_2	20	180	600	r_2
-4	-2	1	33	396	r_2	33	396	990	r_2	33	297	990	r_2
-2	2	2	32	384	r_2	32	384	960	r_2	32	288	960	r_2

Table 2. Comparison between WFEN, IWFEN and Newton's method for Example 2

			Newton		*WFEN*			*IWFEN*		
x_1^0	x_2^0	x_3^0	*IT*	*FE*	*IT*	*FE*	*AS*	*IT*	*FE*	*AS*
2	2	15	387	4644	119	1428	3570	16	144	480
3	3	3	122	1464	19	228	570	19	171	570
3	3	15	612	7344	42	504	1260	18	162	540

Note 2. In examples 1 and 3, the number of iterations of the methods (3) and (4) is identical. This is because of the fact that the $\partial_n f_i$, $i = 1, 2, \ldots, n$ are independent from the x_n component and thus $\partial_n f_i(x^p) \equiv \partial_n f_i(x_{pivot}^{p,i})$, $i = 1, 2, \ldots, n$, $p = 0, 1, \ldots$, hence the methods, given by (3) and (4) are identical, but the new method cost less (n less partial derivatives evaluations per iteration).

Table 3. Comparison between WFEN, IWFEN and Newton's method for Example 3

					Newton			*WFEN*				*IWFEN*			
x_1^0	x_2^0	x_3^0	x_4^0	x_5^0	*IT*	*FE*	r_i	*IT*	*FE*	*AS*	r_i	*IT*	*FE*	*AS*	r_i
-8	-3	4	2	1.5	85	2550	r_3	8	240	400	r_3	8	200	400	r_3
-4	-4	4	2	1.5	80	2400	r_3	10	300	500	r_3	10	250	500	r_3
10	3	4	2	1.5	83	2490	r_3	12	360	600	r_3	12	300	600	r_3

References

1. W. Chen, *A Newton method without evaluation of nonlinear function values*, in http://xxx.lanl.gov/abs/cs.CE/ 9906011, 1999.
2. William La Cruz, Jose Mario Martinez and Marcos Raydan, *Spectral Residual Method Without Gradient Information For Solving Large-Scale Nonlinear Systems of Equations*, Mathematics of Computation, Vol.75, Num.255, pp.1429-1448, 2006
3. J.E. Dennis Jr. and R.B. Schnabel, *Numerical Methods for Unconstrained Optimization and Nonlinear Equations* (Prentice-Hall, Englewood Cliffs, NJ, 1983).
4. T.N. Grapsa and E.N. Malihoutsaki, *Newton's method without direct function evaluations*, accepted to be presented at the "Hellenic European Research on Computer Mathematics & its Applications - HERCMA 2007", Athens, Greece, 2007.
5. T.N. Grapsa, *Implementing the Initialization-Dependence and the Singularity Difficulties in Newton's Method*, working title, 2007.
6. T.N. Grapsa and M.N. Vrahatis, *The implicit function theorem for solving systems of nonlinear equations in* $\mathbb{R}^2$, Inter. J. Computer Math., 28(1989), pp.171–181.
7. T.N. Grapsa and M.N. Vrahatis, *A Dimension-Reducing Method for solving systems of nonlinear equations in* $\mathbb{R}^n$, Inter. J. Computer Math., 32(1990), pp.205–216.
8. J.M. Ortega and W.C. Rheinboldt, *Iterative Solution of Nonlinear Equations in Several Variables*, Academic Press, New York, 1970.
9. W.C. Rheinboldt, *Methods for Solving Systems of Nonlinear Equations*, SIAM, Philadelphia, 1974.
10. M.N. Vrahatis, *CHABIS: A mathematical software package for locating and evaluating roots of systems of nonlinear equations*, ACM Trans. Math. Software, 14(1988), pp.330–336.
11. M.N. Vrahatis and K.I. Iordanidis, *A rapid generalized method of bisection for solving systems of non–linear equations*, Numer. Math., 49(1986), pp.123–138.

On the growth problem for Hadamard matrices

Marilena Mitrouli

Department of Mathematics, University of Athens,
Panepistemiopolis 15784, Athens, Greece.
mmitroul@math.uoa.gr

Abstract. Traditionally, backward error analysis for Gaussian Elimination (GE), on a matrix $A = (a_{ij}^{(1)})$ is expressed in terms of the *g*rowth factor

$$g(n, A) = \frac{\max_{i,j,k} |a_{ij}^{(k)}|}{\max_{i,j} |a_{ij}^{(1)}|},$$

which involves all the elements $a_{ij}^{(k)}$, $k = 1, 2, \ldots, n$, that occur during the elimination. Matrices with the property that no row and column exchanges are needed during GE with complete pivoting are called *completely pivoted* (CP) or feasible. In other words, at each step of the elimination the element of largest magnitude (the "pivot") is located at the top left position of every appearing submatrix during the process. For a CP matrix A we have

$$g(n, A) = \frac{\max\{p_1, p_2, \ldots, pn\}}{|a_{11}^{(1)}|},$$

where $p_1, p_2, \ldots, pn$ are the pivots of A.
In 1968 Cryer conjectured that "$g(n, A) \leq n$, with equality iff A is a Hadamard matrix". In 1991 Gould discovered a 13×13 matrix for which the growth factor is 13.0205. Thus the first part of the conjecture was shown to be false. The second part of the conjecture concerning the growth factor of Hadamard matrices still remains open. Interesting problems remain, such as determining $\lim_{n\to\infty} g(n)/n$ and evaluating $g(n, A)$ for Hadamard matrices. The approach of the growth problem for Hadamard matrices, and for orthogonal matrices in general, from a statistical point of view is also under investigation.
In this talk we will present the overall progress that has been done on the growth problem for Hadamard matrices. Also we take a small step towards proving the Hadamard part of Cryer's conjecture by demostrating all 34 possible pivot patterns for a Hadamard matrix of order 16. The peculiarity of this problem lies in the fact that H-equivalent operations do not preserve pivots, i.e. the pivot pattern is not invariant of the H-equivalency. So, H-equivalent matrices do not have necessarily the same pivot pattern. Although Hadamard matrix problems might sound tantalizingly easy, they are non-trivial, because e.g. for the case of proving the pivot structures for H_{16}, a naive computer exhaustive search performing all posible H-equivalence operations would require $(16!)^4 \approx 10^{53}$ trials

A symmetric Boussinesq system of KdV-KdV type ⋆ ⋆⋆

D. E. Mitsotakis[1,2] and V. A. Dougalis[1,2]

[1] Department of Mathematics, University of Athens, 15784 Zographou, Greece
[2] Institute of Applied and Computational Mathematics, FO.R.T.H., P.O. Box 1527, 71110 Heraklion, Greece
dmitsot@math.uoa.gr,doug@math.uoa.gr

Abstract. We consider the L^2 conservative, symmetric coupled KdV system of Boussinesq equations

$$\begin{aligned} &\eta_t + u_x + \tfrac{1}{2}(\eta u)_x + \tfrac{1}{6}u_{xxx} = 0 \\ &u_t + \eta_x + \tfrac{1}{2}\eta\eta_x + \tfrac{3}{2}uu_x + \tfrac{1}{6}\eta_{xxx} = 0. \end{aligned} \tag{1}$$

This system describes the bidirectional propagation of small-amplitude long waves on the surface of water in a channel, when the surface tension is omitted. In this work we mainly describe traveling wave solutions of the above system. Some solutions appear to be homoclinic to small oscillatory profiles, i.e. symmetric, non-localized solitary waves with identical oscillatory tails on both sides of the central core; these are usually referred as nanopterons or generalized solitary waves. Using a stable and efficient fully discrete numerical method based on a standard Galerkin/finite element method combined with the two stage Gauss-Legendre implicit Runge-Kutta scheme, we study solutions of the initial-periodic boundary value problem for the system (1), the generation of radiating waves and the interactions between generalized solitary waves. Finally we present an optimal-order error estimate for the semidiscrete finite element scheme.

⋆ Expanded version of a talk presented at the Conference in Numerical Analysis 2007 (NumAn 2007) Recent Approaches to Numerical Analysis: Theory, Methods and Applications September 3-7, 2007 – Kalamata, Greece.

⋆⋆ This work was supported by a "Pythagoras" EPEAEK II grant to the Department of Mathematics, University of Athens, which was co-funded by the E.U. European Social Fund.

Re-scaling techniques for computing blowing-up solutions to 2^{nd} order differential equations

Nabil R. Nassif[1], Noha Makhoul Karam[2] and Yeran Soukiassian[3]

[1] Mathematics Department American University of Beirut, Beirut Lebanon
[2] IRISA, Campus Beaulieu, Université de Rennes I, Rennes, France
[3] Computer Science Department, American University of Beirut, Beirut Lebanon

Introduction

Consider the computation of solutions to 2^{nd} order ODEs of the form:

$$y^{''} - b|y^{'}|^{q-1}y^{'} + |y|^{p-1}y = 0,\ t > 0,\ y(0) = y_{1,0},\ y^{'}(0) = y_{2,0}. \tag{1}$$

For $p, q > 1$ this model describes the motion of a membrane element linked to a spring. The non-linear term in y is related to the rigidiy of the spring and that in y' models a "speed-up" of the phenomenon when $b > 0$ and a "slow-down" when $b < 0$. In this last case, the IVP is dissipative and the existence of the solution is global on $[0, \infty)$ ([1], [2]). Computationally, such case is not difficult to handle. However, when $b > 0$, the existence domain of the solution is finite, with the existence of a finite "blow-up time" $T_b > 0$, at which $y(t)$ and $y'(t)$ "explode", i.e. $\lim_{t\to T_b} |y(t)| = \lim_{t\to T_b} |y'(t)| = \infty$. Such situation can exhibit two types of explosive Oscillatory and Non-Oscillatory behaviors. For the case when $b = 1$, Souplet ([3], [4], [5]) considered the equation $y^{''} + |y|^{p-1}y = |y^{'}|^{q-1}y^{'}$, $t \geq 0, \forall\, p, q > 1$. They proved the existence of 2 critical values $q = p$ and $q = \frac{2p}{p+1}$ in the plane (p, q) with three distinct behaviors of the solution to (1): (1) $q \geq p$, explosion in a finite time, (2) $\frac{2p}{p+1} < q < p$, non-oscillatory finite time blow-up. (3) $1 < q \leq \frac{2p}{p+1}$, oscillatory finite time blow-up. In [6], Balabane, Jazar and Souplet have also studied the critical case where $q = \frac{2p}{p+1}$, $p > 1$, with b a positive number. In this work, we present a robust algorithm to efficiently compute the solutions to these singular problems, based on the idea of "sliced-time" computations introduced in [7]. Basic elements of this method are given in the simple case of the next section.

Rescaling for a case study: $y^{''} = y^p$, $p > 1$

Consider the initial value problem:

$$y^{''} = y^p,\ t > 0,\ p > 1, \tag{2}$$

with the initial conditions of (1), (2) can be reduced to a first order initial value problem: $y^{'} = F(y) = \sqrt{\frac{y^{p+1}(t)}{p+1} + \frac{y_{2,0}^2}{2} - \frac{y_{1,0}^{p+1}}{p+1}}$, $t > 0$, $y(0) = y_{1,0} \geq 0$.

As $F(y) = O(y^{\frac{p+1}{2}})$, one easily shows that this problem has a finite time blow-up T_b, such that $\lim_{t\to T_b} y(t) = +\infty$. Re-scaling techniques start by selecting a **coarse grid** that would subdivide the time interval $[0, T]$ of integration of the differential equation (2). Since the problem under study has a finite-time existence domain $[0, T_b)$, where a-priori T_b is unknown, we seek a subdivision of $[0, T_b)$ into an **infinite number of subintervals (slices)**:

$$[0, T_b) = \cup_{n=1}^{\infty}[T_{n-1}, T_n),\ \lim_{n\to\infty} T_n = T_b \text{ and } \lim_{n\to\infty}(T_n - T_{n-1}) = 0. \quad (3)$$

Letting $y_{1,n} = y(T_n)$, and $y_{2,n} = y'(T_n)$, then on the n^{th} slice $[T_{n-1}, T_n]$, we introduce the change of variables:

$$t = T_{n-1} + \beta_n s,\ y(t) = y_{1,n-1}(1 + z_{1,n}(s)),\ y'(t) = y_{2,n-1}(1 + z_{2,n}(s)). \quad (4)$$

The parameter β_n is selected on the basis of allowing the rescaled systems to become **"similar"** on all the slices. For simplicity of notations, we shall use $z_1 = z_{1,n}$ and $z_2 = z_{2,n}$ and find out that z_1 verifies:

$$\frac{d^2 z_1}{ds^2} = \beta_n^2 y_{1,n-1}^{p-1}(1 + z_1)^p,\ s > 0,\ z_1(0) = 0,\ z_1^{'}(0) = \beta_n \frac{y_{2,n-1}}{y_{1,n-1}}. \quad (5)$$

By selecting $\beta_n = \frac{1}{(y_{1,n-1})^{(p-1)/2}}$, (5) becomes

$$\frac{d^2 z_1}{ds^2} = (1 + z_1)^p,\ s > 0,\ z_1(0) = 0 \geq 0,\ z_1^{'}(0) = \omega_n = \frac{y_{2,n-1}}{(y_{1,n-1})^{(p+1)/2}}.$$

At that point, an **additional constraint** is needed: **the end of slice condition**. It allows determining the size of the slice $T_n - T_{n-1} = \beta_n s_n$. In this case, it is based on the observation that $y(T_n) = y_{1,n} = y_{1,n-1}(1 + z(s_n)) > y(T_{n-1}) = y_{1,n-1}$. Hence, the condition $z_1(s_n) = S$, where S is a "cutoff" value that "stops" the growth of $y(t)$ on $[T_{n-1}, T_n]$ (and $z(s)$ on $[0, s_n]$). Such restriction leads to: $y(T_n) = y_{1,n} = y_{1,n-1}(1 + S) = y_{1,0}(1 + S)^n, \beta_n = \frac{1}{(1+S)^{(n-1)(p-1)/2}}$ and the computation of (2) reduces into solving a sequence of "shooting problems", whereby on the n^{th} slice, one computes an initial value problem with a stopping criterion:

$$\frac{d^2 z_1}{ds^2} = (1 + z_1)^p,\ 0 < s \leq s_n,\ z_1(0) = 0,\ z_1^{'}(0) = \omega_n,\ z_1(s_n) = S \quad (6)$$

Note that in such problem, the initial and end values of $z_1(s)$ are preset to 0 and S respectively, while $z_1'(0)$ varies with n. One proves that the sequence of problems (6) are "similar", in the sense that, there exists constants c_0, c_1, d_0, d_1 such that: $\forall n,\ c_0 \leq \omega_n \leq c_1$, $\forall n,\ d_0 \leq s_n \leq d_1$. As a consequence the sequence $\{T_n\}$ associated with the similar problems (6) verifies: $\lim_{n\to\infty} \beta_n = 0$, $\lim_{n\to\infty} T_n = T_b$, and $d_0 g(S) \leq T_b \leq d_1 g(S)$, where $g(S) = \frac{(1+S)^{(p-1)/2}}{(1+S)^{(p-1)/2}-1}$. Numerically, we deal with (6) by changing it into a first order system of equations through the introduction of the variable $z_2(s)$, given by:
$y'(t) = y_{2,n-1}(1 + z_2(s))$. This yields with $z_1(0) = z_2(0) = 0$, $z_1(s_n) = S$:

$$\frac{dz_1}{ds} = \omega_n(1 + z_2),\ \frac{dz_2}{ds} = \frac{1}{\omega_n}(1 + z_1)^p,\ 0 < s \leq s_n. \quad (7)$$

A numerical solver on the n^{th} slice, $n \leq n_0$

Since $\beta_n \cong \frac{T_n - T_{n-1}}{T_1}$, note that for a given computational tolerance of ϵ_{Tol}, the total number of slices n_0, on which we solve (7) is reached when $\beta_{n_0} \leq \epsilon_{Tol} < \beta_{n_0-1}$. Since rescaling provides similar models on all the slices, a first advantage consists in implementing a scheme that uses a uniform mesh for numerical integration on each n^{th} slice $1 \leq n \leq n_0$. Specifically, let us first rewrite (7) in the form:

$$\frac{d\mathbf{z}}{ds} = \mathbf{g_n}(\mathbf{z}),\ 0 < s \leq s_n,\ z(0) = 0,\ z_1(s_n) = S. \tag{8}$$

We have chosen the standard 4^{th} order explicit Runge-Kutta method with mesh size τ. On the n^{th} slice, the method would yield: $\{\mathbf{Z}^{c,k} \cong \mathbf{z}(s^k)\}$, $s^k = k\tau|\mathbf{Z}_1^c(s_k)| \leq S, \forall k \leq l$. To reach the stopping criterion of (8), an adaptative procedure is adopted whereby the final time interval $[s^{l-1}, s^l]$ is iteratively modified until $\mathbf{Z}_1^{c,l} \cong S$. The computed s_n, is such that $s_n^c = s^l$. The mesh size τ, is found on the basis of solving (8) for $n = 1$ using the Runge-Kutta scheme for the given computational tolerance ϵ_{Tol}. One starts with $\tau_1 = \frac{1}{2}$ and refines until achieving a value τ_0 that fulfills the computational tolerance ϵ_{Tol}.

Results of Numerical Tests on $y'' = (y)^p$, $p > 1$, $y(0) = 1$ and $y'(0) = \sqrt{\frac{2}{p+1}}$.

In such case, the blow-up time is given by: $T_b = \int_0^\infty \frac{dy}{\sqrt{\frac{2}{p+1}(1+y)^{p+1}}} = \frac{\sqrt{2\ (p+1)}}{p-1}$. The following table gives the blow-up time T_b of the solution when $y(0) = 1$, $y'(0) = \sqrt{\frac{2}{p+1}}$, $p = 5$, precision $= \frac{1}{2}10^{-09}$ ($T_b^{exact} = 8.660254037844386e-001$).

Cutoff Value	*Number of Slices*	*Computed* T_b	*Relative Error*
1	17	8.660254109836718e-001	8.312958463015252e-009
3	9	8.660256537609148e-001	2.886479716334406e-007
10	6	8.660390890898021e-001	1.580242947111533e-005

Case of $y'' - b|y'|^{q-1}y' + |y|^{p-1}y = 0$, $q = \frac{2p}{p+1}$

On the basis of the subdivision (3) and the change of variables (4), one finds for (1), the equivalent form to (5):

$$\frac{d^2 z_1}{ds^2} = b\beta_n^{2-q}|y_{1,n-1}|^{q-1}|z_1'|^{q-1}z_1' - \beta_n^2|y_{1,n-1}|^{p-1}|1+z_1|^{p-1}(1+z_1),\ s > 0 \tag{9}$$

$z_1(0) = 0$, $z_1^{'}(0) = \beta_n \frac{y_{2,n-1}}{y_{1,n-1}}$. To make these slices similar, one needs to have the coefficients $\omega_n = \beta_n \frac{y_{2,n-1}}{y_{1,n-1}}$, $\gamma_{1,n} = \beta_n^2|y_{1,n-1}|^{p-1}$, $\gamma_{2,n} = \beta_n^{2-q}|y_{1,n-1}|^{q-1}$, uniformly bounded, independently from n. By selecting $\gamma_{1,n} = 1$, we prove a similarity result when $q = \frac{2p}{p+1}$, specifically: $\beta_n = \frac{1}{|y_{1,n-1}|^{(p-1)/2}}$ and in case of blow-up $\lim_{n\to\infty}\beta_n = 0$, $\gamma_{2,n} = 1$ and if the blow-up is non-oscillatory ($b \geq$

$b_1(p) = (p+1)(\frac{p+1}{2p})^{\frac{p}{p+1}})$, then ω_n has a constant sign and there exist positive constants c_1 and c_2 independent from n such that for large n, $c_1 \leq \omega_n \leq c_2$.
If the blow-up is oscillatory ($b < b_1(p) = (p+1)(\frac{p+1}{2p})^{\frac{p}{p+1}})$, then ω_n changes sign and for large n, $\omega_n \leq c_3$, where c_3 is independent from n.
As for the computational approach, we transform (9) into a first-order system, by fixing a cutoff value S that would determine the slice size $T_n - T_{n-1} = \beta_n s_n$. Specifically one computes the Initial value shooting problem:

$$\frac{dz_1}{ds} = \omega_n(1+z_2),\ 0 < s \leq s_n,\ \frac{dz_2}{ds} = b|\omega_n|^{q-1}|1+z_2|^{q-1}(1+z_2) - \frac{1}{\omega_n}|1+z_1|^{p-1}(1+z_1),$$

with $z_1(0) = 0 \geq 0$, $z_2(0) = 0$, $|z_1(s_n)| = S$. The sequences $\{y_{1,n}\}$, $\{y_{2,n}\}$, and $\{T_n\}$ verify:

$$y_{1,n} = y_{1,n-1}[1 + z_1(s_n)],\ y_{2,n} = y_{2,n-1}[1 + z_2(s_n)],\ T_n = T_{n-1} + \beta^{(n)} s_n.$$

The choice of the cutoff value S is based on the facts that if the blow-up is non-oscillatory, then for $n \geq n_0$, $|y_{1,n}| = |y_{1,n-1}|(1+S)$. Otherwise, if the blow-up is oscillatory, then for $\forall n$, $|y_{1,n-1}|(S-1) \leq |y_{1,n}| \leq |y_{1,n-1}|(1+S)$. As a result, if the blow-up is oscillatory, then a sufficient condition for the choice of S is $S > 2$.
Results of Numerical Tests for $y'' = -|y|^{p-1}y + |y'|^{q-1}y'$**,** $q = \frac{2p}{p+1} = 1.2592$**,** $y(0) = 1$**,** $y'(0) = 1$**,** $b = 1.5$**,** $q = \frac{2p}{p+1}$ $p = 1.7$**,** $\epsilon_{Tol} = \frac{1}{2}10^{-006}$**,** $\tau_0 = 7.8125e{-}003$

Cutoff S	Computed T_b	Number of Slices
2	4.741936108013850	49
3	4.741940732163934	37
5	4.741941541394696	27

References

1. Haraux A.: Systèmes dynamiques dissipatifs et applications. RMA, vol 17, (1991) Masson, Paris, Milan, Barcelone, Bonn.
2. Lasalle J.: Some extensions of Liapunov's second method, IRE Trans. Circuit Theory, CT-7 (1960) 520-527.
3. Souplet P.: Critical Exponents, special large-time behavior and oscillatory blow-up in nonlinear ode's Differential and Integral Equations, 11, (1998), 147-167.
4. Souplet P.: Etude des solutions globales de certaines équations différentielles ordinaires du second ordre non-linéaires. C.R. Acad. Sci. Paris, t., 313, Série I (1991), 365-370.
5. Souplet P.: Existence of exceptional growing-up solutions for a class of nonlinear second order ordinary differential equations. Asymptotic Analysis, 11 (1995), 185-207.
6. Souplet P., Jazar M., Balabane M.: Oscillatory blow-up in nonlinear second order ode's : The critical case. Discrete And Continuous dynamical systems Volume 9, Number 3, May 2003.
7. Nassif N., Fayad D., Cortas M.: Sliced-time computations with Re-scaling for Blowing-Up solutions to Initial Value Differential Equations. V.S. Sunderam et al. (Eds.): ICCS 2005, pp. 58-65, Springer-Verlag 2005.

The conditioning of FD matrix sequences coming from semi-elliptic Differential Equations

D. Noutsos[1], S. Serra Capizzano[2], and P. Vassalos[3]

[1] Department of Mathematics, University of Ioannina, Ioannina, Greece.
dnoutsos@cc.uoi.gr
[2] Dipartimento di Fisica e Matematica, Università dell'Insubria - Sede di Como, Italy.
stefano.serrac@uninsubria.it, serra@mail.dm.unipi.it
[3] Department of Informatics, Athens University of Economics and Business, Athens, Greece.
pvassal@aueb.gr, pvassal@uoi.gr

Abstract. We are concerned with the study of spectral properties of the sequence of matrices $\{A_n(a)\}$ coming from the discretization, using centered finite differences of minimal order, of elliptic (or semielliptic) differential operators $L(a,u)$ of the form

$$\begin{cases} -\frac{d}{dx}\left(a(x)\frac{d}{dx}u(x)\right) = f(x) & \text{on } \Omega = (0,1) \\ \text{Dirichlet B.C. on } \partial\Omega \end{cases}$$

where the nonnegative, bounded coefficient function $a(x)$ of the differential operator may have some isolated zeros in $\bar{\Omega} = \Omega \cup \partial\Omega$. More precisely, we state and prove the explicit form of the inverse of $\{A_n(a)\}$ and some asymptotic behavior of the minimal eigenvalue (condition number) of the related matrices. As a conclusion, and in connection with our theoretical findings, first we extend the analysis to higher order (semi-elliptic) differential operators, and then we present various numerical experiments, showing that similar results must hold true in 2D as well.

Reachability and Holdability of Nonnegative States

Dimitrios Noutsos[1] and Michael Tsatsomeros[2]

[1] Department of Mathematics, University of Ioannina, GR-45110 Ioannina, Greece,
dnoutsos@uoi.gr
[2] Department of Mathematics, Washington State University, Pullman, WA 99164, USA,
tsat@math.wsu.edu

Abstract. Linear differential systems $\dot{x}(t) = Ax(t)$, $A \in \mathbb{R}^{n\times n}$, whose solutions become and remain nonnegative are studied. The corresponding matrices possess the Perron-Frobenius property and are associated to eventually nonnegative matrices, namely, matrices whose powers become and remain nonnegative. Initial conditions that result in nonnegative states are shown to form a convex cone that is related to the matrix exponential e^{tA} and its eventual nonnegativity.

Weighted Quadrature Rules for Finite Element Methods

Saulo P. Oliveira[1], Alexandre L. Madureira[2] and Frederic Valentin[2]

[1] LAGEP-CPGG, Salvador-BA, Brazil. saulopo@cpgg.ufba.br.
[2] LNCC, Petrópolis-RJ, Brazil. alm@lncc.br, valentin@lncc.br.

Abstract. We discuss the numerical integration of polynomials times exponential weighting functions arising from multiscale finite element computations. The new rules are more accurate than standard quadratures and are better suited to existing codes than formulas computed by symbolic integration. We test our approach in a multiscale finite element method for the 2D reaction-diffusion equation.

Standard finite elements usually fail to accurately solve equations with multiscale behavior. This can happen if coefficients are oscillatory or if a small parameter multiplies some of the terms in the equation. A strategy to overcome these difficulties is to use special spaces instead of the space of piecewise polynomial functions [1, 2]. However, for polynomial basis functions, standard quadratures are *exact* and this is not the case for more complicated spaces. We investigate quadratures to integrate elementwise products of polynomials and exponential basis functions. Such integrals appear when developing enriched methods for reaction-advection-diffusion equations [2], but also in other contexts [3]. Quadrature formulas are simpler to implement into existing finite element codes than results of symbolic integrations.

We define an N-point *weighted quadrature* in $[a, b]$ with *weighting function* w by a set of *integration weights* A_l and *integration points* $x_l \in [a, b]$ such that

$$\int_a^b q(x)w(x)\,dx \approx \sum_{j=1}^{N} A_l q(x_l) \tag{1}$$

for a given function q. The *Newton-Cotes rule*, one of the simplest quadratures of degree of precision n, is defined by choosing $x_l = a + (l-1)(b-a)/n$ and

$$A_l = \int_a^b \prod_{\substack{i=1 \\ i\neq l}}^{n+1} \frac{(x - x_i)}{(x_l - x_i)} w(x)\,dx, \quad l = 1, \ldots, n+1\ . \tag{2}$$

The *Gaussian* quadrature uses integration weights defined as in (2), but the integration points are the roots of the n-th degree polynomial p satisfying

$$\int_a^b p(x)q(x)w(x)\,dx = 0 \quad \forall\ q \text{ of degree } \leq n\ . \tag{3}$$

The Gaussian quadrature has the *optimal* degree of precision $2n-1$. However, Gaussian quadratures may not be the best choice when performing weighted integrals in finite element codes, since the quadrature points may change from element to element. Newton-Cotes rules are sub-optimal, but allow one to fix the quadrature points and re-calculate only the quadrature weights.

We employ one-dimensional quadratures to approximate weighted integrals over quadrilateral regions. Using isoparametric maps, such integrals can be transformed into integrals in the reference square $[-1,1]^2$. Assuming that $w(x,y) = w_x(x)w_y(y)$ and that $f(x,y)$ is a polynomial function of degree at most $2N-1$ in both x and y, we have that

$$\int_{-1}^{1}\int_{-1}^{1} f(x,y)w(x,y)\,dxdy = \sum_{j=1}^{N}\sum_{k=1}^{N} A_j^x A_k^y f(x_j,y_k) \ , \tag{4}$$

where x_l and A_l^x are the integration points and weights for the 1D Gaussian quadrature with respect to w_x (similarly for y). The above rule is referred to as a *product rule*. For instance, let $\hat{\phi}_1(x) = (1-x)/2$ and $w_x(x) = \exp[-a_x(1-\hat{\phi}_1(x))]$, $a_x > 0$. The integration weights A_j^x for the nine-point Newton-Cotes rule with

$$x_1 = y_1 = -1/3, \qquad x_2 = y_2 = 0, \qquad x_3 = y_3 = 1/3 \ , \tag{5}$$

are found by replacing (5) and $w(x) = \exp[-a_x(1-\hat{\phi}_1(x))]$ into (2). In particular,

$$A_3^x = 6\frac{12 - a_x(5-a_x) - (12 + a_x(7+2a_x))e^{-a_x}}{a_x^3} \ .$$

Replacing a_x by a_y in the equations above yields the definition of A_j^y. The Gaussian rule with similar degree of precision has two points in either direction (a total of four points); the orthogonal polynomial that generates the integration points associated to the weight function $w_x(x)$ is

$$p_2(t) = 8 + a^2 - 8at + a^2t^2 - 2a\frac{a(8+a^2-2at) - 2(4-at)\sinh(a)}{2 + a^2 - 2\cosh(a)} \ .$$

For the sake of illustration, we plot the point locations as we vary a_y, keeping $a_x = 10$. We choose $w(x,y) = w_x(x)w_y(y)$. We plot in Fig. 1 the Gaussian points for $a_y = 1, 10, 100$. We also plot the points of the Newton–Cotes quadrature, which do not depend neither on a_x, nor on a_y.

Quadratures in Triangular Regions

Optimal quadratures for triangles rely on two-dimensional orthogonal polynomials or on the solution of non-linear systems [4]. Similarly to quadrilaterals, integrals in arbitrary triangles can be transformed into integrals in the triangle with vertices $(0,0)$, $(0,1)$ and $(1,0)$. However, the limits of integration in

$$I_w(f) := \int_0^1 \int_0^{1-x} f(x,y)w(x,y)\,dydx, \quad w(x,y) = w_x(x)w_y(y) \ . \tag{6}$$

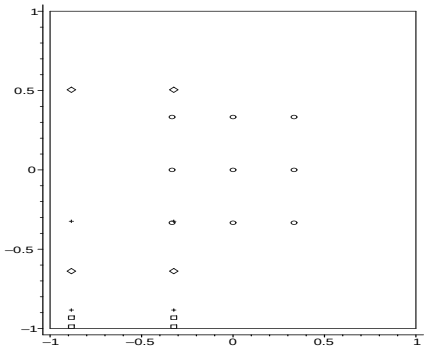


Fig. 1. Gaussian points with $a_x = 10$. Diamonds correspond to $a_y = 1$, crosses to $a_y = 10$, and squares to $a_y = 100$. The fixed circles correspond to Newton–Cotes.

prevent the direct use of product rules. We consider next integrals of the form (6) with $w(x, y) = e^{-ax-by}$, where a and b are postive numbers.

Let us start with a three-point Newton–Cotes Rule. Each integration weight A_k can be found by integrating the Lagrange interpolation polynomial associated to the point $\boldsymbol{p}_k = (x_k, y_k)$ as in (2). Given the points $\boldsymbol{p}_1 = (1/2, 1/2)$, $\boldsymbol{p}_2 = (0, 1/2)$, and $\boldsymbol{p}_3 = (1/2, 0)$, we have $A_k = b_k/(a^2(a-b)b^2)$, where

$$
\begin{aligned}
b_1 &= e^{-a}(2+a)b^2 - a^2(2+b)e^{-b} - (a(b-2)-2b)(a-b),\\
b_2 &= (a-2)(a-b)^2 + (2+b-a)a^2e^{-b} - (a^2 - a(b-4) - 2b)be^{-a},\\
b_3 &= (b-2)(b-a)^2 + (2+a-b)b^2e^{-a} - (b^2 - b(a-4) - 2a)ae^{-b}\ .
\end{aligned}
$$

The Gaussian quadrature of degree of precision one easily follows from the equation $I_w(f) = A_1 f(x_1, y_1)$. Making $f = 1$ yields $A_1 = I_w(1)$; x_1 and y_1 follow from choosing $f = x$ and $f = y$, i.e., $x_1 = I_w(x)/A_1$, and $y_1 = I_w(y)/A_1$. If $w(x, y) = e^{-ax-by}$, then $A_1 = [b(1-e^{-a}) - a(1-e^{-b})]/[a(b-a)b]$, while

$$
\begin{aligned}
x_1 &= [(a-b)^2 + b((a-b)(1+a)+a)e^{-a} - a^2e^{-b}]/[a^2(b-a)^2bA_1],\\
y_1 &= [(a-b)^2 - a((a-b)(1+b)-b)e^{-b} - b^2e^{-a}]/[a(b-a)^2b^2A_1]\ .
\end{aligned}
$$

Application: a Multiscale Finite Element

Let us consider the linear reaction-diffusion problem

$$
-\varepsilon\,\Delta u + \sigma\, u = f \quad \text{in } \Omega \subset \mathbb{R}^2, \quad u = 0 \quad \text{on } \partial\Omega\ , \tag{7}
$$

where $\sigma, \varepsilon > 0$. To approximate (7), we discretize Ω by a conforming and regular partition using triangular elements K and select the finite dimensional subspace $V_h(\Omega) \subset H_0^1(\Omega)$ of piecewise linear polynomials. We seek $u_h \in V_h(\Omega)$ such that

$$
\varepsilon \int_\Omega \nabla u_h \cdot \nabla v_h \, d\boldsymbol{x} + \sigma \int_\Omega u_h\, v_h \, d\boldsymbol{x} = \int_\Omega f\, v_h \, d\boldsymbol{x} \quad \forall v_h \in V_h(\Omega)\ . \tag{8}
$$

The classical Galerkin method just described is inadequate if $\varepsilon \ll \sigma h_K^2$, where h_K denotes the characteristic length of element K. The method lacks stability, and non-physical oscillations appear in the numerical solution. Such issue is treated in [2] by enriching the trial space $V_h(\Omega)$ with *multi-scale functions* $\lambda(\boldsymbol{x}) = \sinh(\alpha_K\,\psi(\boldsymbol{x}))/\sinh(\alpha_K)$, where $\alpha_K \sim h_K(\sigma/\varepsilon)^{1/2}$ is the Peclet number, and $\psi(\boldsymbol{x})$ are piecewise linear shape functions. Thus we need to accurately compute

$$\int_K \lambda(\boldsymbol{x})\,\psi(\boldsymbol{x})\,d\boldsymbol{x}, \qquad \int_K \nabla\lambda(\boldsymbol{x})\,\nabla\psi(\boldsymbol{x})\,d\boldsymbol{x}\ .$$

The above integrals can be written in the form presented in the previous section.

Let the domain Ω be the unit square, which we discretize by a non-uniform mesh of 400 elements. We impose the boundary conditions $u(x,0) = u(0,y) = 0$ and $u(x,1) = u(1,y) = 1$. We set $\sigma = 1$, $f = 0$, and $\varepsilon = 10^{-6}$. The three-point Newton-Cotes rule allows us to conserve all desirable properties of the multi-scale method unlike the classical one-point Gauss, which leads to a loss of accuracy similar to the one observed through the Galerkin method (Fig. 2).

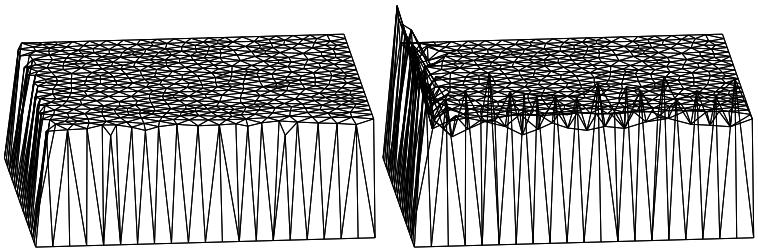


Fig. 2. Solutions by the new exponential-adaptative integration formula (left) and standard (non-weighted) one-point Gauss integration (right).

References

1. I. Babuška, U. Banerjee, and J. Osborn. Generalized finite element methods — main ideas, results and perspectives. *Int. J. Comput. Methods*, 1:67–103, 2004.
2. L. Franca, A. Madureira, and F. Valentin. Towards multiscale functions: Enriching finite element spaces with local but not bubble-like functions. *Comput. Methods Appl. Mech. Engrg.*, 60:3006–3021, 2005.
3. R. Sacco and M. Stynes. Finite element methods for convection-diffusion problems using exponential splines on triangles. *Computers Math. Applic.*, 35:35–45, 1998.
4. A. H. Stroud. *Approximate Calculation of Multiple Integrals*. Prentice-Hall, Englewood Cliffs, 1971.

Asymmetric periodic orbits in the photogravitational restricted three-body problem

Kostantinos Papadakis[1], Omiros Ragos[2], and Christodoulos Litzerinos[1]

[1] Department of Engineering Sciences, University of Patras, GR-26504 Patras, Greece
k.papadakis@des.upatras.gr
[2] Department of Mathematics, University of Patras, GR-26504 Patras, Greece
ragos@math.upatras.gr

Abstract. In this paper we study the $Ox-$asymmetric solutions of the planar photogravitational restricted three-body problem in the case of primaries with equal masses ($1-\mu=\mu=0.5$) and equal values of the radiation pressure parameters ($q_1=q_2=q$). In particular, we concern with the families of asymmetric orbits which bifurcate from the well known families a, b, and c. Their evolution is examined via the numerical construction of series of the critical bifurcation points of a, b, and c with respect to the variation of the radiation parameter q, in the range between the classical gravitation case $q=1$ and the critical case $q^*=1/8$ when the triangular equilibrium points coincide with the inner collinear point [4].

Key Words: Asymmetric orbit, critical orbit, Levi-Civita regularization, numerical integration, periodic orbit, radiation pressure, three-body problem.

Introduction

We have adopted the usual dimensionless rectangular rotating coordinate system in which the equations expressing the motion of the test particle are given by :

$$\begin{aligned}\ddot{x}-2\dot{y}&=x-\frac{q_1(1-\mu)}{r_1^3}(x+\mu)-\frac{q_2\mu}{r_2^3}(x+\mu-1),\\ \ddot{y}+2\dot{x}&=y\left(1-\frac{q_1(1-\mu)}{r_1^3}-\frac{q_2\mu}{r_2^3}\right),\end{aligned}\tag{1}$$

where,

$$r_1=\sqrt{(x+\mu)^2+y^2},\qquad r_2=\sqrt{(x+\mu-1)^2+y^2}.$$

For details see [6]. The Jacobian integral of this motion is given by the expression

$$C=x^2+y^2+2\left(\frac{q_1(1-\mu)}{r_1}+\frac{q_2\mu}{r_2}\right)-(\dot{x}^2+\dot{y}^2).\tag{2}$$

For the determination of the series of critical orbits we have used the method that has been developed by Hénon [1, 2]. Whenever we had to face orbits which went through the close vicinity of the singularities of the problem, we regularized the equations of motion by applying the transformations of Levi-Civita [3].

Numerical Results and Comments

In our case study, the evolution of the family a is similar to that of b due to the symmetry with respect to Oy, so, in what follows, we will just refer to family b. In the case $q = 1$, each of the families a, and b contains exactly one critical orbit for which $a_h = 1$ and $b_h = 0$. In Table 1 and the left part of Figure 1 we present the evolution of the bifurcation point of b. Here we note by x_0 and x_1

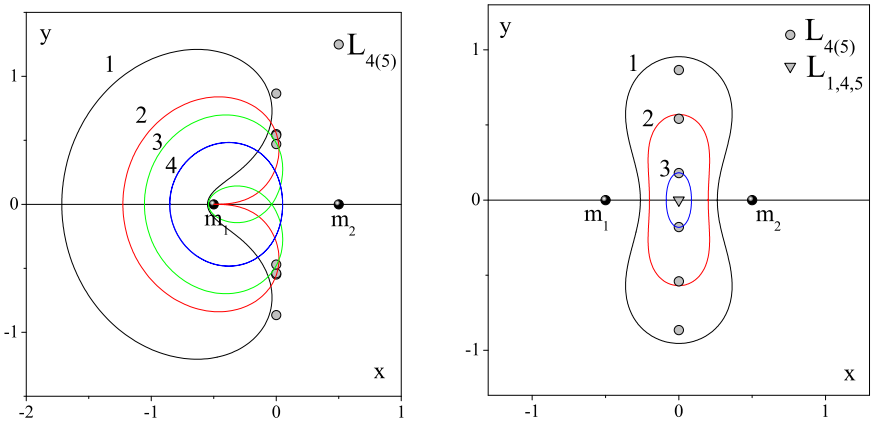


Fig. 1. Left: The evolution of the critical periodic orbit of the family b as the radiation factor varies : (1) $q = 1$, (2) $q = 0.409$, (3) $q = 0.32355$, and (4) $q = 0.397547$. Right: The corresponding evolution of the original critical periodic orbit of the family c : (1) $q = 1$, (2) $q = 0.4$, and (3) $q = 0.15$. Small circles indicate the positions of L_4 and L_5. The small triangle (right frame) indicates the common position of L_1, L_4 and L_5 when $q = q^*$.

the $x-$coordinates of the intersections of the O$x-$axis with the members of the series of critical orbits at $t = 0$ and $t = T/2$. In Figure 1, the orbit named by (1) represents the period-1 critical orbit of b when $q = 1$. The orbit (2) is the bifurcation point of b for $q = 0.409$ and it is still a solution of period 1. The solution indicated by (3) is a period-2 critical solution when $q = 0.32355$. For $q = 0.397547$ we have the solution (4) which is the termination periodic orbit of the series and coincides to a member of the family h. For any value of q in the range from 1 to 0.397547, the families a and b continue to contain exactly one critical solution ($a_h = 1$ and $b_h = 0$).

In the case $q = 1$, the family c also contains exactly one critical orbit for which $a_h = 1$ and $b_h = 0$. This original critical orbit continues to exist until $q = q^*$. Its

Table 1. Series, with respect to the radiation parameter q, of critical periodic orbits $(a_h = 1, b_h = 0)$ of the family b.

q	x_0	x_1	$T/2$	C
1.0	-1.71555788	-0.54973450	2.89508541	2.02078683
0.9	-1.65423664	-0.54116423	2.87621884	1.84732407
0.8	-1.58742265	-0.53220019	2.85434619	1.66660773
0.7	-1.51363274	-0.52289059	2.82862262	1.47708066
0.6	-1.43055580	-0.51344303	2.79786439	1.27633591
0.5	-1.33410663	-0.50462341	2.76045357	1.06002398
0.4	-1.21467874	-0.50008147	2.71481830	0.81699447
0.323493	-1.05121053	-0.54532783	2.68181192	0.53617102
0.397547	-0.85286119	-0.85286119	2.70576395	0.48489815

evolution with respect to q is presented in the right part of Figure 1 and the left part of Table 2. The corresponding series consists of period-1 symmetric orbits whose size gradually decreases until, finally, this series terminates on L_1. The

Table 2. Two series, with respect to the radiation parameter q, of critical periodic orbits $(a_h = 1, b_h = 0)$ of the family c.

q	$x_0(=-x_1)$	$T/2$	C	q	$x_0(=-x_1)$	$T/2$	C
1.0	-0.26244069	2.63087648	2.37166911	0.955	-0.33179341	6.16472801	2.72677099
0.9	-0.25540454	2.63030912	2.21393119	0.9	-0.31577097	6.12020973	2.60447449
0.8	-0.24761727	2.63209761	2.04817200	0.8	-0.29198680	6.04754239	2.38018634
0.7	-0.23882563	2.63729051	1.87301444	0.7	-0.27110771	5.98210494	2.14973073
0.6	-0.22861934	2.64761066	1.68660412	0.6	-0.25108033	5.92423804	1.91023469
0.5	-0.21627271	2.66609874	1.48633231	0.5	-0.23045895	5.87799828	1.65873290
0.4	-0.20033150	2.69868437	1.26830458	0.4	-0.20741814	5.85443733	1.39127733
0.3	-0.17725424	2.75881792	1.02618644	0.3	-0.17827074	5.88550058	1.10173368
0.2	-0.13456609	2.88646856	0.74814705	0.2	-0.13115529	6.10260555	0.77889489
1/8	0.0	3.14159265	0.5	1/8	-0.03701696	7.03027714	0.50039431

afore mentioned series of critical solutions of family c is not unique. For values of the parameter q less than 0.955 this family contains more critical orbits. For these values of q, the family c starts from the collinear equilibrium point L_1 and terminates on a homoclinic asymptotic solution which intersects the x-axis perpendicularly and spirals to L_4 for $t \to +\infty$ and to L_5 for $t \to -\infty$ [5]. The characteristic curve (C, x_0) of the family c also spirals asymptotically to $(C_{L_{4,5}}, x_{L_{4,5}})$. Along each loop of the spiral there is a critical solution of c. This characteristic curve retains this behavior for $0.955 \geq q \geq 0.14$. So, theoretically, for each of these values of q, the family c contains an infinite number of bifurcation points with families of asymmetric periodic orbits. If q is smaller than 0.14 but greater than q^*, there are exactly two critical solutions in the family c. For $q < q^*$, one critical orbit exists on the "Short"-family that starts

from $L_{1,4,5}$. The behavior of the critical solution of c that persists throughout the whole range from 0.955 to q^* is represented in the right part of Table 2. The shapes of the members of the corresponding series are almost alike to those given in the right part of Figure 1.

In the left part of Figure 2, we present the intersection points x_0 and x_1 of the members of the series of critical orbits of families a and b. In the right part of the same figure we display the corresponding intersections points that concern the two series of critical solutions which are given in Table 2. In Figure 3, the

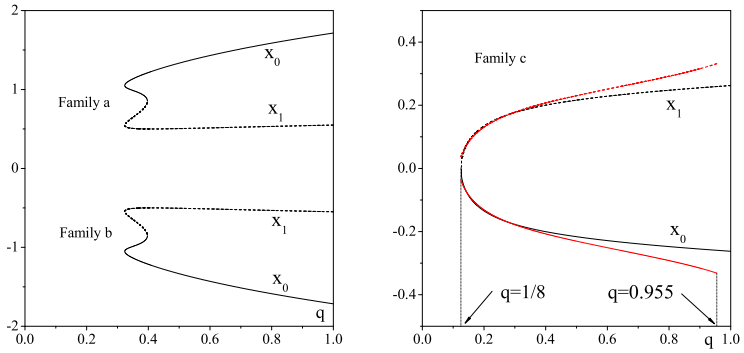


Fig. 2. The intersection points x_0 and x_1 of the critical periodic orbits ($a_h = 1$, $b_h = 0$) of the families a, b (left) and c (right) for varying q.

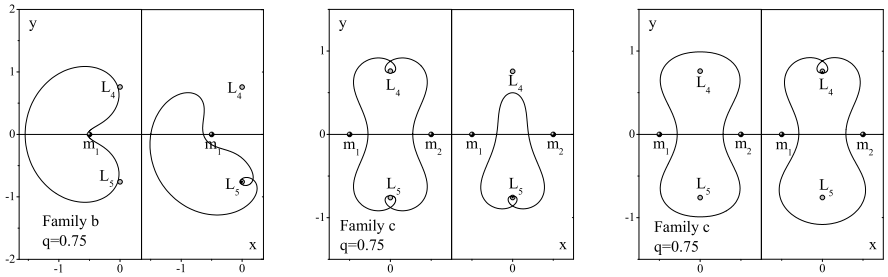


Fig. 3. $q = 0.75$. First frame : The critical member of b and the asymptotic orbit-termination point of the bifurcating family of asymmetric periodic orbits. Second and third frame : The two critical orbits of c and the terminations orbits of the bifurcating families of asymmetric solutions.

bifurcation and the termination point of three families of asymmetric periodic solutions are illustrated for $q = 0.75$. The first frame corresponds to the asymmetric family which bifurcates from b while the other frames represent the two asymmetric families bifurcating from c.

References

1. Hénon, M.: Exploration numérique du problème restreint. II Masses égales, stabilité des orbites périodiques, Ann. Astrophys. **28** (1965) 992–1007.
2. Hénon M.: Families of asymmetric periodic orbits in Hill's problem of three bodies, Cel. Mech. & Dyn. Astr. **93** (2005) 87–100.
3. Levi-Civita, T.: Sur la résolution qualitative du probléme des trois corps, Acta Math. **30** (1906) 305–327.
4. Markellos, V.V., Perdios, E.A., Papadakis, K.E.: The stability of inner collinear equilibrium points in the photogravitational elliptic restricted probem, Astr. and Spac. Sc. **199** (1993) 139–146.
5. Papadakis, K.E.: Families of periodic orbits in the photogravitational three-body probem, Astr. and Spac. Sc. **245** (1996) 1–13.
6. Simmons, J.F.L., McDonald, A.J.C., Brown, J.C.: The restricted 3-body problem with radiation pressure, Celest. Mech. **35** (1985) 145–187.

The distance from a matrix polynomial to matrix polynomials with a prescribed multiple eigenvalue[1]

Nikolaos Papathanasiou and Panayiotis Psarrakos[2]

Department of Mathematics
National Technical University of Athens
Zografou Campus, 15780
Athens, Greece

Extended Abstract. Consider an $n \times n$ *matrix polynomial* $P(\lambda) = \sum_{j=0}^{m} A_j \lambda^j$, where λ is a complex variable and $A_j \in \mathbb{C}^{n\times n}$ with $\det A_m \neq 0$. The *spectrum* of $P(\lambda)$, i.e., the set of all *eigenvalues* of $P(\lambda)$, is defined and denoted by

$$\sigma(P) = \{\lambda \in \mathbb{C} : \det P(\lambda) = 0\}.$$

An eigenvalue $\lambda_0 \in \sigma(P)$ is called multiple if its multiplicity as a zero of $\det P(\lambda)$, that is, its *algebraic multiplicity*, is greater than one. Moreover, the *geometric multiplicity of* $\lambda_0 \in \sigma(P)$ is the dimension of the null space of matrix $P(\lambda_0)$. The study of matrix polynomials has a long history, especially with regard to their spectral analysis, which leads to the solutions of higher order linear systems of differential equations [1].

We are interested in perturbations of $P(\lambda)$ of the form $Q(\lambda) = \sum_{j=0}^{m}(A_j + \Delta_j)\lambda^j$, where the matrices $\Delta_j \in \mathbb{C}^{n\times n}$ are arbitrary. For a given $\varepsilon > 0$ and a set of nonnegative weights $\mathrm{w} = \{w_0, w_1, \ldots, w_m\}$ with $w_0 > 0$, we define the class of admissible perturbed matrix polynomials

$$\mathcal{B}(P, \varepsilon, \mathrm{w}) \;=\; \left\{ Q(\lambda) = \sum_{j=0}^{m} (A_j + \Delta_j)\lambda^j : \|\Delta_j\|_2 \leq \varepsilon\, w_j,\ j = 0, 1, \ldots, m \right\}.$$

The weights w_j allow freedom in how perturbations are measured.

For a $\mu \in \mathbb{C}$, we define the *distance from* $P(\lambda)$ *to* μ *as a multiple eigenvalue* by

$$\mathcal{E}_a(\mu) \;=\; \min\{\varepsilon \geq 0 : \exists\, Q(\lambda) \in \mathcal{B}(P, \varepsilon, \mathrm{w}) \text{ with } \mu \text{ as a multiple eigenvalue}\},$$

and the *distance from* $P(\lambda)$ *to* μ *as an eigenvalue with geometric multiplicity* κ by

$$\mathcal{E}_{g,\kappa}(\mu) = \min\{\varepsilon \geq 0 : \exists\, Q(\lambda) \in \mathcal{B}(P, \varepsilon, \mathrm{w}) \text{ with } \mu \text{ as an eigenvalue of geometric multiplicity at least } \kappa\}.$$

[1] Research supported by a grant of the EPEAEK project PYTHAGORAS II. The project is co-funded by the European Social Fund (75%) and Greek National Resources (25%).

[2] Corresponding author. E-mail: ppsarr@math.ntua.gr.

Denote by

$$s_1(A) \geq s_2(A) \geq \cdots \geq s_n(A) \geq 0$$

the singular values of a matrix A, and let

$$w(\lambda) = w_m\lambda^m + \cdots + w_1\lambda + w_0 \quad \text{and} \quad \phi = \frac{w'(|\mu|)}{w(|\mu|)}\,\frac{\overline{\mu}}{|\mu|}.$$

Define also the $2n \times 2n$ matrix polynomial

$$F[P(\lambda);\gamma] = \begin{bmatrix} P(\lambda) & 0 \\ \gamma P'(\lambda) & P(\lambda) \end{bmatrix} ; \quad \gamma \geq 0,$$

and for $\begin{bmatrix} u_1(\gamma) \\ u_2(\gamma) \end{bmatrix}, \begin{bmatrix} v_1(\gamma) \\ v_2(\gamma) \end{bmatrix} \in \mathbb{C}^{2n}$ ($u_k(\gamma)$, $v_k(\gamma) \in \mathbb{C}^n$) a pair of left and right singular vectors of $s_{2n-1}(F[P(\mu);\gamma])$, respectively, the $n \times 2$ matrices

$$U(\gamma) = [u_1(\gamma)\ u_2(\gamma)] \quad \text{and} \quad V(\gamma) = [v_1(\gamma)\ v_2(\gamma)].$$

Then we have the following results [2]:

(a) It holds that

$$\mathcal{E}_{g,\kappa}(\mu) = \frac{s_{n-\kappa+1}(P(\mu))}{w(|\mu|)}.$$

(b) For every $\gamma > 0$,

$$\mathcal{E}_a(\mu) \geq \frac{s_{2n-1}(F[P(\mu);\gamma])}{w(|\mu|)} \left\| \begin{bmatrix} 1 & 0 \\ \gamma\frac{w'(|\mu|)}{w(|\mu|)} & 1 \end{bmatrix} \right\|_2^{-1},$$

and if $rank(V(\gamma)) = 2$, then

$$\mathcal{E}_a(\mu) \leq \frac{s_{2n-1}(F[P(\mu);\gamma])}{w(|\mu|)} \left\| U(\gamma) \begin{bmatrix} 1 & -\gamma\,\phi \\ 0 & 1 \end{bmatrix} V(\gamma)^\dagger \right\|_2,$$

where $V(\gamma)^\dagger$ denotes the Moore-Penrose pseudoinverse of $V(\gamma)$.

(c) If $\mu \in \mathbb{C}\backslash\sigma(P')$, $s_{2n-1}(F[P(\mu);\gamma])$ attains a maximum value at $\gamma_* > 0$ and $s_* = s_{2n-1}(F[P(\mu);\gamma_*]) > 0$, then there exists a right singular vector of s_*, $\begin{bmatrix} v_1(\gamma_*) \\ v_2(\gamma_*) \end{bmatrix} \in \mathbb{C}^{2n}$, such that

$$\mathcal{E}_a(\mu) \leq \frac{s_*}{w(|\mu|)} \left\| V(\gamma_*) \begin{bmatrix} 1 & -\gamma_*\,\phi \\ 0 & 1 \end{bmatrix} V(\gamma_*)^\dagger \right\|_2.$$

(d) If $\mu \in \mathbb{C}\backslash\sigma(P)$, $s_{2n-1}(F[P(\mu);\gamma])$ attains a maximum value at $\gamma_* = 0$ and $s_* = s_{2n-1}(F[P(\mu);0]) = s_n(P(\mu))$ (> 0), then

$$\mathcal{E}_a(\mu) \leq \frac{s_*}{w_0}.$$

In all cases, perturbations that correspond to the upper bounds are directly constructed. Finally, numerical examples are presented to illustrate and evaluate our results.

References

1. Gohberg I., Lancaster P., Rodman L.: *Matrix Polynomials*, Academic Press, New York, 1982.
2. Papathanasiou N., Psarrakos P.: The distance from a matrix polynomial to matrix polynomials with a prescribed multiple eigenvalue. *Submitted*, 2007.

Simultaneous Solution of Large Scale Linear Systems and Eigenvalue Problems

G. Pashos, M.E. Kavousanakis, A.N. Spyropoulos, J.A. Palyvos, and A.G. Boudouvis *

School of Chemical Engineering, National Technical University of Athens, 15780, Greece

Abstract. A method for simultaneous solution of large and sparse linearized equation sets and the corresponding eigenvalue problems is presented. Such problems arise from the discretization and the solution of nonlinear problems with the finite element method and Newton iteration. The method is based on a parallel version of the preconditioned GMRES(m) by deflation. The parallel code exploits the architecture of the computational clusters using the MPI (Message Passing Interface). The proposed method has high parallel speedup and small memory requirements.

Introduction

The main computational cost of the finite element codes comes from the solution of large linear algebraic equation systems. Krylov-type iterative solvers are commonly used for the solution of these systems due to their small memory requirements and high parallel efficiency compared with the direct solvers.

Of crucial importance in engineering applications and related computations is the structure of the solution space of nonlinear problems, which depicts the dependence of the solution on the parameters. Most interesting among the aspects of the solution space are singularities, i.e., critical values of parameters at which solution multiplicity and stability change, such as bifurcation and turning points.

In computational practice, a turning point can be easily detected from the failure of Newton iteration to converge in an ordinary parameter continuation. It is circumnavigated by special parameter continuation techniques, such as of arc-length type [4], and its detection is completed by an eigenvalue computation on each side of the singularity. On the other hand, bifurcation points can be passed without any noticeable effect on the convergence rate of Newton iteration, provided that the continuation step is large enough to straddle the singularity - which is usually the case. Therefore, to secure detection of singular points, several eigenvalue problems with the Jacobian must be solved during the continuation for monitoring the so-called dangerous eigenvalues and the corresponding eigenvectors; this results to large computational cost.

* corresponding author; boudouvi@chemeng.ntua.gr

Newton-like methods [3] are commonly used for the numerical solution of nonlinear equations systems

$$F(u, \lambda) = 0 \tag{1}$$

$F : \mathbb{R}^N \times \mathbb{R} \to \mathbb{R}^N$ is a vector of nonlinear functions, u is a vector of size N and $\lambda \in \mathbb{R}$ is a continuation parameter. Newton's method solves (1) iteratively starting from an initial guess u^0 and at each step approximates the solution of (1) with the vector

$$u^{k+1} = u^k + x^k, \quad k = 0, 1...\text{until convergence} \tag{2}$$

In (2), x^k is the solution of the linear system

$$J(u^k, \lambda)x^k = -F(u^k, \lambda) \tag{3}$$

where $J(u^k, \lambda) \equiv F_u(u^k, \lambda) \in \mathbb{R}^{N \times N}$ is the Jacobian matrix.

When the Jacobian is large, storage limitations as well as exploitation of parallel computing demand the iterative solution of the linear system (3). Krylov subspace iterative methods are commonly used to extract an approximate solution of (3). A Krylov-type method requires only the product of the Jacobian matrix with several vectors and not the explicit computation of the elements of the Jacobian [1]. This is indispensable in cases where no analytical expression for these elements is available (i.e. matrix-free computations).

The restarted variant of the GMRES method [5], commonly known as GMRES(m) is the Krylov-type iterative solver of preference in case where the linear system (3) is non-symmetric. The combination of the GMRES(m) with a good preconditioning technique is necessary for its convergence. At regular points as well as near singular points, the GMRES(m) is preconditioned by a deflation technique [2].

Preconditioned GMRES(m) by deflation

Preconditioned GMRES is the method of choice for the iterative solution of large algebraic equation sets with non-symmetric matrices, on the basis of its parallel efficiency [6]. Starting from an initial guess, x_0, of the solution of (3), GMRES uses Arnoldi's method, combined with an orthogonalization technique - the Modified Gram-Schmidt method is used here - to construct an orthonormal basis $V_m \in \mathbb{R}^{N \times m}$ of the m-dimensional Krylov subspace, $K_m(J, v) = \text{span}\left\{v, Jv, J^2v, ..., J^{m-1}v\right\}$, where $v \equiv r_o/||r_o||_2$, $r_o \equiv -F - Jx_o$, $F \equiv F\left(u^k, \lambda\right)$, $J \equiv J\left(u^k, \lambda\right)$.The new approximation of the solution is

$$x_m = x_o + V_m y_m \tag{4}$$

where y_m is a vector of size m and it is computed from the solution of the least squares problem

$$y_m = \underset{y}{\text{argmin}} \|\beta e_1 - \bar{H}_m y\|_2, \quad y \in \mathbb{R}^m \tag{5}$$

In (5), $\beta \equiv \|r_o\|_2$, $e_1 = [1, 0..., 0]^T$ and $\bar{H}_m \in \mathbb{R}^{(m+1)\times m}$ is an upper Hessenberg matrix, such as

$$JV_m = V_{m+1}\bar{H}_m \Rightarrow V_m^T JV_m = H_m \tag{6}$$

$H_m \in \mathbb{R}^{m\times m}$ is an upper Hessenberg matrix obtained from the $\bar{H}_m$ by deleting its last row.

The storage requirements and the computational cost of Arnoldi's method increase rapidly with m and, thus, a restarting variant of the GMRES -the GMRES(m)- is used in practice. When m reaches a certain preset value, the algorithm restarts, using the last approximation x_m from (4) as a new initial guess.

A preconditioner is essential to enhance the convergence rate of the GMRES(m), especially near singular points [7]. Thus, the original linear system (3) must be transformed to an equivalent one that has better convergence properties. The linear system (3) is preconditioned from the right

$$JM^{-1}z = -F, \quad x = M^{-1}z \tag{7}$$

In (7) z is a vector of size N and $M^{-1} \in \mathbb{R}^{N\times N}$ is the preconditioner matrix which is constructed from a deflation technique [2] and it is given from

$$M^{-1} = I_N + U(|\mu|T^{-1} - I_r)U^T \tag{8}$$

where $\mu \in \mathbb{R}$ is the largest eigenvalue of the Jacobian matrix, $I_N \in \mathbb{R}^{N\times N}$, $I_r \in \mathbb{R}^{r\times r}$ are identity matrices, $U \in \mathbb{R}^{N\times r}$ is an orthonormal basis of the r-dimensional invariant subspace, P_r, corresponding to the r smallest (in absolute value) eigenvalues of the Jacobian and $T \in \mathbb{R}^{r\times r}$ such as

$$T = U^T JU \tag{9}$$

Practical implementation - Results

The largest eigenvalue, needed in (8), and the eigenvectors of the Jacobian, needed in (9), are approximated by those of the Hessenberg matrix H_m.

At each restart of the GMRES(m), l eigenvectors of the Jacobian are approximated from l eigenvectors, $z_i \in \mathbb{R}^m$, of the H_m that correspond to the smallest eigenvalues. In practice ($l = 1$ or 2). The new vectors:

$$u_i = V_m z_i, \quad i = 1, ..., l \tag{10}$$

are orthonormalized against those of U and added to U. So the dimension r of P increases by l.

In practice, the dimension m of H_m is small (here $m = 1,000$ for $N = 1,0000,000$). So the computational cost of the eigenvalue problem for H_m is negligible.

The preconditioner needs two arrays of size $2Nr$ for the storage of U and JU needed in (8) and (9). In order to save memory requirements and computational

cost, an upper limit r_{max} on r is set ($r_{max} = 20$ on the problems where the algorithm is implemented). When $r > r_{max}$ the update of the preconditioner is continued (i.e. l new vectors are added to P) but at the same time, l vectors of P that correspond to the l largest eigenvalues of the matrix T are deflated. The orthonormal basis U of P is constructed from the eigenvectors z_i of H_m times the orthonormal basis V_m of the Krylov subspace (8). Thus, the vectors u_i approximate the eigenvectors of the Jacobian matrix that correspond to its smallest eigenvalues. These eigenvalues are computed from the $r_{max} \times r_{max}$ matrix T (9). The $N \times N$ elements of the preconditioner matrix M^{-1} are not computed explicitly and so are not stored. Instead, the expression of the preconditioner in (8) is used to perform the matrix-vector product needed in (7).

The main operations of the preconditioned GMRES(m) by deflation algorithm are vector updates, inner products and matrix-vector products. These operations can be efficiently performed on several processors in parallel [6]. The MPI-based parallel code is applied to the solution of linearized equation sets resulting from the discretization, by the Galerkin / finite element method, of a three-dimensional, nonlinear and free boundary problem of interfacial magnetohydrostatics [7]. The computations were done on a 32 processor computational cluster of the Computer Center of the School of Chemical Engineering of NTUA. The code exhibits parallel speedup up to 31 on 32 processors for a problem size $N = 1,0000,0000$.

Acknowledgements. This work has been funded by the project PENED 2003. The project is cofinanced 75% of public expenditure through EC - European Social Fund, 25% of public expenditure through Ministry of Development - General Secretariat of Research and Technology and through private sector, under measure 8.3 of OPERATIONAL PROGRAMME "COMPETITIVENESS" in the 3rd Community Support Programme.

References

1. P.N. Brown, Y. Saad, *Hybrid Krylov methods for nonlinear systems of equations*, SIAM J. Sci. Stat. Comp. **11** (1990), 450-481.
2. K. Burrage, J. Erhel, *On the performance of various adaptive preconditioned GMRES strategies*, Numer. Linear Algebra Appl. **5** (1998), 101-121.
3. J.D. Dennis, R.B. Schnabel, *Numerical Methods for Unconstrained Optimization and Nonlinear Equations*, SIAM: Philadelphia, 1996.
4. H.B. Keller, *Numerical solution of bifurcation and nonlinear eigenvalue problems*, Applications of Bifurcation Theory, Academic Press, New York, 1977, 359-384.
5. Y. Saad, M.H. Schultz, *A generalized minimal residual algorithm for solving nonsymmetric linear systems*, SIAM J. Sci. Stat. Comp. **7** (1986), 856-869.
6. A.N. Spyropoulos, J.A. Palyvos, A.G. Boudouvis, *Finite element computations on cluster of PC's and workstations*, in: Proc. EURO-PDP '00, IEEE Computer Society, Los Alamitos, CA, USA, 1999, 56-61.
7. A.N. Spyropoulos, J.A. Palyvos, A.G. Boudouvis, *Bifurcation detection with the (un)preconditioned GMRES(m)*, Comput. Methods Appl. Mech. Engrg. **193** (2004), 4707-4716.

Effective Modification of the BFGS Method for Training Recurrent Neural Networks

C-C. Peng and G.D. Magoulas

School of Computer Science, Birkbeck College, University of London
Malet Street, London WC1E 7HX

Abstract. In this paper we propose a new variant of BFGS that incorporates adaptive mechanisms, such as nonmonotone strategy and self-scaling parameters and explore its performance in training recurrent networks. Preliminary simulation results show the proposed algorithm outperforms the BFGS, providing an effective modification that is capable of training recurrent networks of various architectures.
Keywords: recurrent neural networks, quasi-Newton methods, BFGS, nonmonotone line-search, second-order training algorithms

Introduction

A recurrent neural network (RNN) is an artificial neural network in which self-loop and backward connections between nodes are allowed. RNNs are well-known for their power to memorise time dependencies, model nonlinear systems, map input sequences to output sequences and in principle they can implement any kind of sequential behaviour. Despite some attempts to use second-order methods for training RNNs, first-order methods still remain the most popular choice. This is mainly attributed to the nature of the particular problem and the time dependency of the training data which generate error landscapes that cause instabilities in the calculation of the Hessian matrix, vanishing gradients and convergence to local minima that are far away from any desired minimisers.

In this paper, we propose a modified version of BFGS, which combines the usage of adaptive nonmonotone learning and self-scaling utility; the former takes the benefits of Lipschitz constant and provides more information of the morphology of a given function, while the latter has been proved to have the ability of solving unconstrained nonconvex optimisation problems.

Nonmonotone BFGS with Adaptive Self-Scaling

Traditional optimisation strategies for RNNs use monotone strategies which may be trapped in a local minimum point and never jump out to the global one under ill conditions, such as poorly initialised weights in the case of RNNs. In an attempt to alleviate this situation this work explores the use of a nonmonotone approach, which was first proposed in [3]

$$f\left(x_{k+1}\right) \leq \max_{0 \leq j \leq M_k} \left[f\left(x_{k-j}\right)\right] + \gamma \alpha_k g_k^T d_k, \tag{1}$$

where f is the objective function and g is the gradient, d the search direction, M_k is named nonmonotone learning horizon [6] and the constant $\gamma \in (0,1)$. Furthermore, [6] proposed a dynamic and efficient way to automatically adapt the size of M through a local estimation of the Lipschitz constant which could provide helpful information on the morphology of a function.

We combine this approach with a quasi-Newton method. At each iteration, the new approximated Hessian matrix B_{k+1} is required to satisfy the quasi-Newton condition $B_{k+1}s_k = y_k$, where s_k and y_k are the changes in function variable and in gradient, respectively. In our work, B_{k+1} is updated using the Broyden-Fletcher-Goldfarb-Shanno formula (BFGS), which is the most commonly used update technique for training feedforward neural networks. In practice, it is better to tune the Hessian approximations at each iteration when eigenvalues become large. To this end, there are several scaling approaches in the optimisation literature, such as [1, 4], where the scaling factor ρ_k, [8], can be defined as

$$\rho_k = \frac{y_k^T s_k}{s_k^T B_k s_k} \tag{2}$$

Numerical evidences show that methods that apply a scaling factor for B_{k+1} are superior to the original quasi-Newton methods. In other words, when$\rho_k$is sufficiently large, the eigenvalues of B_{k+1} are relative small, with strong self-correcting property [8]. Despite this property looks particularly appealing for training RNNs, to the best of our knowledge it has not been explored at all in this area to improve the effectiveness of second-order training algorithms. A high-level description of the proposed algorithm follows.

Algorithm: **Adaptive Self-scaling Non-monotone BFGS**

STEP 0. Initialise a point x_0, k, a symmetric positive definite matrix B_0, value of nonmonotone learning horizon M;
STEP 1. If $g(x_k) = 0$, stop;
STEP 2. Determine the search direction by $d_k = -B_k^{-1} g_k$;
STEP 3. Adapt M^k by following conditions:
$M^k = \begin{cases} M^{k-1} + 1, \text{ if } \Lambda^k < \Lambda^{k-1} < \Lambda^{k-2} \\ M^{k-1} - 1, \text{ if } \Lambda^k > \Lambda^{k-1} > \Lambda^{k-2} \\ M^{k-1}, \text{ otherwise,} \end{cases}$,where $\Lambda^k = \frac{\|g(x_k) - g(x_{k-1})\|}{x_k - x_{k-1}}$;
STEP 4. Find a step length α_k using the following line search.
For $0 < \lambda_1 < \lambda_2$ and $\sigma, \delta \in (0,1)$, at each iteration, one chooses a parameter l_k such that the step length $\alpha_k = \bar{\alpha}_k \cdot \sigma^{l_k}$, where $\bar{\alpha}_k \in (\lambda_1, \lambda_2)$, satisfies
$f(x_k + \alpha_k d_k) \leq \max\limits_{0 \leq j \leq M^k} f(x_{k-j}) + \delta \cdot \alpha_k \cdot g(x_k)^T \cdot d_k$
STEP 5. Generate a new iteration point by $x_k = x_k + \alpha_k d_k$;
STEP 6. Update the Hessian approximation B_k by the following self-scaling BFGS

$$B_{k+1} = \rho_k \left[B_k - \frac{B_k s_k s_k B_k^T}{s_k^T B_k s_k} \right] + \frac{y_k y_k^T}{y_k^T s_k},$$

where $s_k = x_{k+1} - x_k$, $y_k = g_{k+1} - g_k$ and ρ_k is the self-scaling factor in the form of eq.(2);
STEP 7. Set $k := k + 1$, go to STEP 1.

There are three heuristic parameters in the above algorithm, i.e. M^k in Step 3 and σ, δ in Step 4. Step 3 requires past M and Λ values to operate properly. Thus, for a few iterations the monotone Charalambous line-search method [2] is used to accumulate information in order to update M. We also set an upper bound for M^k to help the algorithm concentrate on the recent past. In Step 4, σ regulates the step size, i.e., larger σ smaller trail step-size, while δ controls the amount of change. More details of tuning these heuristic parameters could be found in the section of our experiments.

Experimental Results

Parity-N Problem

The first test case is the well-known n-bit parity problem which is nonlinear separable and has been widely used to verify the performance of novel training algorithms [5]. We consider here two instances: the parity-5 problem and the parity-10. The stopping criterion is set to a Mean-Squared-Error=0.01 within 2000 epochs. In both case, the heuristic parameters are set as $3 \leq M^k \leq 15, \quad \sigma = 0.9$ and $\delta = 0.01$. As shown in Tables 1, the performance of our method for the parity-5 problem on 3 different neural architectures, i.e. Feedforward Time Delay (FFTD), Elman's Recurrent Network (LRN) and Nonlinear Autoregressive Network with Exogenous Inputs (NARX), using 5 and 7 hidden nodes is always better than the original BFGS. The notation used in these tables is as follows: *#hidden* indicates the number of hidden nodes used; *Conv* indicates the percentage of runs that met the MSE condition within 2000 epochs; *Ave* give the average MSE (in %) achieved by each method at the end of training; *Min* and *Max* give the minimum and the maximum number of epochs to converge to the MSE condition.

In particular, Tables 1 shows that the new method is able to locate minimisers with smaller function values than the original method which is important in certain real-world problems to provide good generalisation. For example, in Table 1, 16.7% of the BFGS-trained FFTD networks reached an MSE=0.01 in a maximum of 189 epochs, while the average MSE achieved by BFGS in that case was 0.063093. That was caused by the fact that the majority of the BFGS-trained networks didn't reach the MSE goal within 2000 epochs; some of them stuck to minima with higher function values while others failed to converge because of instabilities in the Hessian.

For the parity-10 problem we used FFTD and LRN with 10 hidden nodes which is the standard configuration for this problem and we also experimented with 2, 5 and 10 hidden nodes using a NARX network; in all cases conditions were the same as in parity-5. The results are shown in Tables 2. When BFGS

fails to reach the error goal we only provide the average error obtained. Also in Table 2, a 0% convergence indicates that not a single run of the BFGS method converged within 2000 epochs. We observed that ASCNMBFGS provided a consistent behaviour and a better ability to escape from swallow local minima which in our opinion can be attributed to its nonmonotone behaviour.

Table 1. Simulation results for the parity-5 using three RNN architectures

Algorithm	#hidden	Net	Conv	Ave.	Min	Max
BFGS	5	FFTD	16.7	6.309	134	189
		LRN	13.3	6.127	110	157
		NARX	60.0	1.991	47	149
	7	FFTD	23.3	3.725	64	1324
		LRN	36.7	2.470	35	1376
		NARX	66.7	1.866	49	628
ASCNMBFGS	5	FFTD	100.0	0.896	53	189
		LRN	100.0	0.899	55	178
		NARX	100.0	0.841	41	69
	7	FFTD	100.0	0.912	62	87
		LRN	100.0	0.887	64	85
		NARX	100.0	0.858	47	78

Table 2. Simulation results for the parity-10 using three RNN architectures

Algorithm	#hidden	Net	Conv	Ave.	Min	Max
BFGS	10	FFTD	0.0	6.049	–	–
		LRN	0.0	5.873	–	–
		NARX	66.7	0.069	158	573
	5	NARX	66.7	3.392	92	957
	2		0.0	7.083	–	–
ASCNMBFGS	10	FFTD	100.0	0.996	672	1129
		LRN	100.0	0.998	889	1970
		NARX	100.0	0.009	112	149
	5	NARX	100.0	0.091	64	144
	2		100.0	0.912	75	152

Reading Aloud

This is a temporal sequence prediction problem [7], which is used for verifying the generalisation of our algorithm. The task is to learn the mapping a set of orthographic representation to their phonological forms [7]. Both subsets of orthography and phonology have 3 different parts, i.e., onset, vowel and coda, with 30, 27 and 48 possible characters for the input, and 23, 14 and 24 possible characters for the output, respectively.

Table 3. Simulation results for the reading aloud problem

Algorithm	ASCNMBFGS	BFGS
Ave. Train Error (%)	5.5979	6.0887
Ave. Test Error (%)	0.6830	1.7852

In the original paper [7], the authors use a specially designed architecture with 100 hidden nodes for this 105-input and 61-output problem. Here we use a NARX network with 5 hidden nodes only, and instead of 1900 training epochs, we apply only 300 epochs within each run. The heuristic parameters used here are $3 \leq M^k \leq 15, \quad \sigma = 0.5$ and $\delta = 0.9$. The results are shown in Table 3 for the BFGS and the new algorithm. Both methods are able to train networks to reach small training errors but when it comes to average performance in both training and testing (generalisation) the new modified BFGS showed better ability to produce solutions with lower errors on the average (cf. Table 3).

Conclusion and Future Work

Traditional algorithms for training RNNs are usually gradient descent-based and apply monotone decrease of the learning error. In this paper, we proposed an adaptive self-scaling nonmonotone BFGS method that aims to overcome the drawbacks of the original BFGS. Further testing and comparisons are of course needed but in our tests so far the modified method demonstrated some promising results, outperforming the original method considerably. It provided more stable behaviour and thus a higher possibility of convergence.

References

[1] M. Al-Baali, Numerical experience with a class of self-scaling quasi-Newton algorithms, J. *Optim. Theory Appl.*, 96 (1998), 533-553.

[2] C. Charalambous, A conjugate gradient algorithm for the efficient training of artificial neural networks. IEE Proceedings Part G. 139 (1992) 301 -310.

[3] L. Grippo, F. Lampariello and S. Lucidi, A nonmonotone line search technique for Newton's method, *SIAM J. Numerical Analysis*, 23 (1986), 707-716.

[4] J. Nocedal and Y. Yuan, Analysis of a self-scaling quasi-Newton method, *Math. Program*, 61 (1993), 19-37.

[5] P.K.H. Phua and D. Ming, Parallel nonlinear optimization techniques for training neural networks, *IEEE Trans. Neural Networks*, 14(6), pp. 1460-1468, 2003.

[6] V.P. Plagianakos, G.D. Magoulas, M.N. Vrahatis, Deterministic nonmonotone strategies for effective training of multi-layer perceptrons. *IEEE Trans. Neural Networks*, 13(6) (2002) 1268-1284.

[7] D. Plaut, J. McClelland, M. Seidenberg and K. Patterson, Understanding normal and impaired reading: computational principles in quasi-regular domains, Psychological Review, 103 (1996), 56-115.

[8] H.X. Yin and D.L. Du, The global convergence of self-scaling BFGS algorithm with nonmonotone line search for unconstrained nonconvex optimization problems. *Acta Math. Sinica.* (2006).

Analysis of a class of hybrid dynamical systems with hysteresis phenomenon

Céline Quémard[1]

LISA UPRES EA 4014 - Université d'Angers,
62, avenue Notre-Dame du Lac, F-49000 Angers, France

Abstract. This paper aims to complete and to generalize the analysis of a particular class of hybrid dynamical systems (h.d.s.) presented in [1]. We present a method to obtain equations of cycles of any period k, $k \geq 1$ and we extend our application fields thanks to a more general mathematical model. Realistic applications to a thermal device and to a power converter come to illustrate all theoretical results.

Presentation of the studied mathematical model

General presentation

In $\mathbb{R}^N$, we consider a basis which in practice, will be either the canonical basis or an eigenvectors basis. In relation to this basis, we consider the following h.d.s.:

$$\dot{X}(t) = A(q(\xi(t)))X(t) + V(q(\xi(t))), \quad \xi(t) = c_t - WX(t), \tag{1}$$

where A is a square matrix of order N, V, X are columns matrices of order N, W is a row matrix of order N, all those matrices having real entries. Term c_t is a constant of $\mathbb{R}$. Moreover, we suppose that matrix A is stable and that X and so ξ are continuous. In this model, the discrete variable is q, taking two possible values according to ξ and follows a hysteresis phenomenon.

Thermostat with an anticipative resistance model

The first application we consider is the one of a thermostat with an anticipative resistance which controls a convector located in the same room (see Figure 1 on the left). A Newton law and a power assessment give the following system:

$$\begin{cases} m_t C_t \dot{x} = -\frac{x-y}{R_t} + q(x)P_t \\ m_p C_p \dot{y} = -\frac{y-z}{R_c} - \frac{y-\theta_e}{R_m} \\ m_c C_c \dot{z} = -\frac{z-y}{R_c} + q(x)P_c \end{cases}$$

which can be put in form (1):

$$\begin{cases} \dot{X} = AX(t) + q(\xi(t))B + C, \\ \xi(t) = LX(t). \end{cases}$$

Coefficients R_t, R_c, R_m are thermal resistances, C_t, C_p and C_c are mass heats according to indexes t, p, c which correspond respectively to the thermostat, the room and the convector.
Discrete variable q here follows the hysteresis phenomenon described on the right in Figure 1, where θ_1 and θ_2 represent respectively lower and upper thresholds.

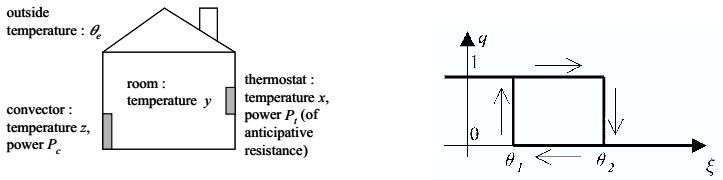


Fig. 1. Thermal process (on the left) and hysteresis phenomenon (on the right)

DC/DC converter with relay feedback control model

The second application is a DC/DC converter with relay feedback control presented in [5]. The equivalent circuit of this system is given in Figure 2. Electronics

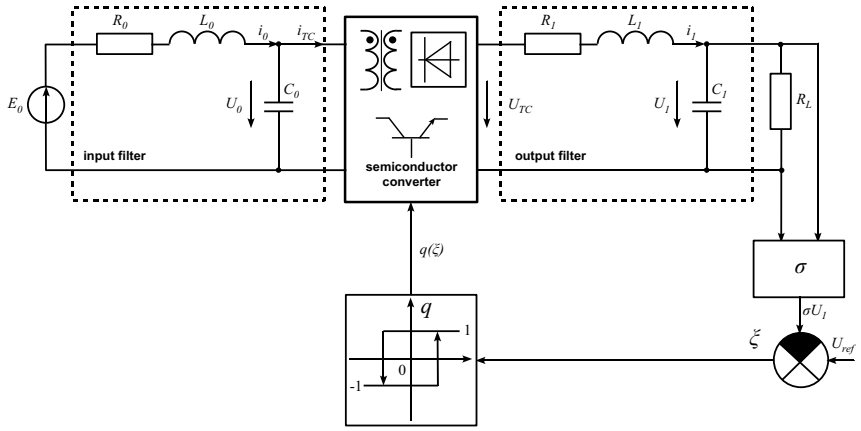


Fig. 2. Equivalent circuit of the DC/DC converter with relay feedback control

laws give the following differential system:

$$\begin{cases} \dot{i_0} = -\frac{R_0}{L_0} i_0 - \frac{1}{L_0} U_0 + \frac{E_0}{L_0} \\ \dot{U_0} = \frac{1}{C_0} i_0 - \frac{1+q(\xi(t))}{2C_0} i_1 \\ \dot{i_1} = \frac{1+q(\xi(t))}{2L_1} U_0 - \frac{R_1}{L_1} i_1 - \frac{1}{L_1} U_1 \\ \dot{U_1} = \frac{1}{C_1} i_1 - \frac{1}{C_1 R_L} U_1, \end{cases}$$

which can be put in form (1) like this:

$$\begin{cases} \dot{X}(t) = A(q(\xi(t)))X(t) + B, \\ \xi(t) = U_{ref} - UX(t). \end{cases}$$

Coefficients R_0, R_1, R_L represent resistances, L_0, L_1 are currents in inductances and C_0, C_1 are voltages on the capacitors. This circuit compares the output voltage σU_1, $0 < \sigma < 1$, to the reference signal U_{ref}. The difference $U_{ref} - \sigma U_1$ (deviation signal) is applied to the hysteresis ($q = -1$ or $q = 1$). Values $-\chi_0$ and χ_0 represent respectively the lower and the upper thresholds.

Determination of period-k equations, $k \geq 1$

We generalize in this section the work begun in [1], [2], giving a method to obtain equations for cycles of any period k, $k \geq 1$.
Let t_0 be a given initial instant and let $t_1 < t_2 < \ldots < t_n < t_{n+1} < \ldots$ be the suite of successive distinct switching times in $[t_0, \infty[$. Let us set $q_n \triangleq q(\xi(t_n))$, $\Delta q_n \triangleq q_n - q_{n-1}$, $\xi_n \triangleq \xi(t_n)$, $A_n \triangleq A(q_n)$, $V_n \triangleq V(q_n)$. Classical integration of differential system (1) gives in $[t_n, t_{n+1}[$:

$$X(t) = \mathrm{e}^{(t-t_n)A_n}\Gamma_n - A_n^{-1}V_n, \tag{2}$$

where $\Gamma_n \in \mathbb{R}^N$ correspond to integration constants, functions of n.
Thus, introducing notation $\forall n \geq 1$, $\sigma_n = t_n - t_{n-1} > 0$ and considering the assumption of continuity at t_n, we obtain:

$$\forall n \geq 1, \quad \Gamma_n = \mathrm{e}^{\sigma_n A_{n-1}}\Gamma_{n-1} + A_n^{-1}V_n - A_{n-1}^{-1}V_{n-1}. \tag{3}$$

Moreover, constant Γ_0 is given by (2) for the case where $n = 0$ and $t = t_0$. Then, considering ξ_n as a function f of upper and lower thresholds S_1, S_2 and of q_{n-1}, q_n and also using the definition of ξ_n, we obtain $\forall n \geq 1$:

$$c_t - W(\Gamma_n - A_n^{-1}V_n) - f(S_1, S_2, q_{n-1}, q_n) = 0. \tag{4}$$

Solutions of (1) with unknowns $X(t)$, $(t_n)_{n\geq 1}$ is equivalent to the ones of (3), (4) with unknowns $(\Gamma_n)_{n\geq 1}$, $(\sigma_n)_{n\geq 1}$. Later on, we restrict us to a numerical solution. From this, we can propose a general expression for equations of period-k cycles, $k \geq 1$, different than the one proposed in [5].
Let $(U_n)_{n\in\mathbb{N}}$ be a suite. We introduce this notation : $U_n^i = U_{2k+i}$, $n \geq 0$, for $i = 1, .., 2k$ with $k \in \mathbb{N}^*$ corresponding to the period of the cycle. The suite of successive switching times is noted $(\sigma_n^1, ..., \sigma_n^{2k})_{n\in\mathbb{N}}$ and the suite of integration constants is noted $(\Gamma_n^1, ..., \Gamma_n^{2k})_{n\in\mathbb{N}}$ so we can set: $R_n \triangleq (\sigma_n^1, \Gamma_n^1, \ldots, \sigma_n^{2k}, \Gamma_n^{2k})$.
System of equations (3), (4), $\forall n \geq 1$ is equivalent to system $H(R_n, R_{n+1}) = 0$, $\forall n \geq 1$, where $H = (H_1, \ldots, H_{4k})^T$ is a function that we define for $i = 1, \ldots, 2k$ by:

$$\begin{cases} H_i(R_n, R_{n+1}) = \Gamma_{n+1}^i - \mathrm{e}^{\sigma_{n+1}^i A_{i-1}}\Gamma_r^{i-1} - A_i^{-1}V_i + A_{i-1}^{-1}V_{i-1} = 0, \\ H_{2k+i}(R_n, R_{n+1}) = c_t - W(\Gamma_{n+1}^i - A_i^{-1}V_i) - f(S_1, S_2, q_{n-1}, q_n) = 0, \end{cases} \tag{5}$$

with $r = n$ if $i = 1$, $r = n + 1$ otherwise, $\Gamma_{n+1}^0 = \Gamma_{n+1}^{2k}$, $i = 0$ if i is even and $i = 1$ if i is odd. Replacing R_n by its limit $R = (\sigma^1, \Gamma^1, \ldots, \sigma^{2k}, \Gamma^{2k})^T$ in system (5), from each of $2k$ first equations and using the remaining $(2k - 1)$, we can determine Γ^i, $i = 1, .., 2k$, only as functions of σ^i, $i = 1, .., 2k$. Then, replacing those equations for Γ^i, $i = 1, .., 2k$ in the $2k$ last equations H_{2k+i}, $i = 1, .., 2k$ of system (5) with $R_n = R$, we deduce for $i = 1, .., 2k$ the following system:

$$\begin{aligned} &F_i = c_t - W((I_N - \textstyle\prod_{m=1}^{2k} D_{(i-m+1)\mathrm{mod}(2k)})^{-1}(I_N + \textstyle\sum_{j=1}^{2k-1}(-1)^j \\ &(\textstyle\prod_{l=1}^{2k-j} D_{(i-l+1)\mathrm{mod}(2k)}))(A_i^{-1}V_i - A_{i-1}^{-1}V_{i-1}) - A_i^{-1}V_i) - f(S_1, S_2, q_{n-1}, q_n)) = 0. \end{aligned} \tag{6}$$

Remark 1. To reduce notations, we used here $[j] = j\mathrm{mod}(2k)$ and $D_j = \mathrm{e}^{\sigma^j A_{j-1}}$, $D_0 = D_{2k}$. Moreover, if $[j] = 0$, we admit that $\Gamma^{[j]} = \Gamma^{2k}$ and that $\sigma^{[j]} = \sigma^{2k}$.

Period doubling bifurcation

A one parameter variation enables to underline the crossing from a period-1 to a period-2 cycle. It is a phenomenon called period doubling bifurcation [5]. At the bifurcation point, the cycle of period 1 loses its stability (one of the eigenvalues of the Jacobian of the Poincaré application is equal to -1) when a stable cycle of period 2 appears.
Generally, bifurcations of this type are only observed on bifurcation diagrams. However, authors of [4] propose a theorem to prove theoretically obtained graphic results. Nevertheless, this version of the theorem is limited since it only concerns systems of dimension 1. That's why, in [3], we generalize and prove this result for systems of any dimension N, $N \geq 1$. This theorem is applied for values which give diagram bifurcations in figure 3 on the left for the example of the thermostat and on the right for the one of the DC/DC converter. Therefore, since all conditions are satisfied, it confirms the existence of period doubling bifurcations for our relative simple model.

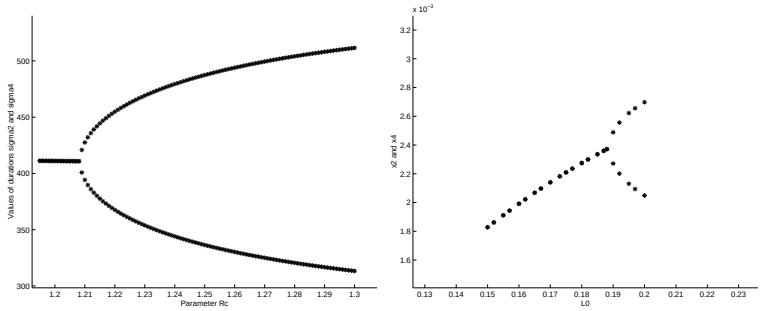


Fig. 3. Diagram bifurcations for particular values for the application of the thermostat (on the left) and of the DC/DC converter (on the right)

References

1. Quémard, C., Jolly, J.-C., Ferrier, J.-L., *Search for Cycles in Piecewise Linear Hybrid Dynamical Systems with Autonomous Switchings. Application to a Thermal Process* , Int. Conf. IMACS, 2005.
2. Quémard, C., Jolly, J.-C., Ferrier, J.-L., *Search for Period-2 Cycles in a Class of H.D.S. with Autonomous Switchings*, Int. Conf. ADHS, 2006.
3. Quémard, C., Jolly, J.-C., *Bifurcation doublement de période pour une classe particulière de systèmes dynamiques hybrides commutations autonomes*, Congrès SMAI, 2007.
4. Robinson, C., *Dynamical Systems*, CRC Press, 1999.
5. Zhusubaliyev, Z.T., Mosekilde, E., *Bifurcations and Chaos in Piecewise-Smooth Dynamical Systems*, World Scientific, 2003.

Accelerated Finite Difference Method for a Simplified Phase Field Model.*

Chrisovalantis A. Sfyrakis **

Mathematics Department, University of Athens,
Panepistimiopolis, 15784 Zographou, Greece
and
Institute of Applied and Computational Mathematics,Fo.R.T.H.,
P.D. Box 1527, 71110 Heraklion, Greece
hammer@math.uoa.gr

Abstract. We consider the phase field model consisting of the system of p.d.e' s

$$q(\theta)\phi_t = \nabla \cdot (A(\theta)\nabla\phi) + f(\phi, u),$$
$$u_t = \Delta u + [p(\phi)]_t,$$

where $\phi = \phi(x, y, t)$ is the phase indicator function, $\theta = \arctan(\phi_y/\phi_x)$, $u = u(x, y, t)$ is the temperature, $q,\ p$, and f are given scalar functions, and A is a 2×2 matrix of given functions of θ. This system describes the evolution of phase and temperature in a two phase medium, and is posed for $t \geq 0$ on a rectangle in the x, y plane with appropriate boundary and initial conditions. We solve the system using two numerical methods. First, we solve the system by a finite difference method, based on the explicit Euler scheme for the first equation and the Crank-Nicolson-ADI method for the second. We also solve the system by another finite difference method,that uses for both equations the Crank-Nicolson-ADI method. We show results of relevant numerical experiments, compare the errors of the two methods, and compare their speed-up when we implement them using parallel possessors.

* Expanded version of a talk presented at the Conference in Numerical Analysis 2007 (NumAn 2007) Recent Approaches to Numerical Analysis: Theory, Methods and Applications September 3-7, 2007 – Kalamata, Greece.

** This work was supported by a "Pythagoras" EPEAEK II grant to the Department of Mathematics, University of Athens, which was co-funded by the E.U. European Social Fund and the Greek Ministry of Education.

Performance Comparison of the Element Free Galerkin Method and the Finite Pointset Method

I.V. Shevchenko

Computational Center of Southern Federal University
High Performance Computing Department
200/1 Stachki Avenue, Building 2, Rostov-on-Don, 344090, Russian Federation

Abstract. Neglecting turbulent motion of the atmosphere we consider the Navier-Stokes equations to calculate the wind velocities. We use a Lagrangian discrete vortex method to compute main characteristics of the flow. The core of the vortex method is to find the solution to the Poisson equation. For solving the Poisson equation with Dirichlet and Neumann boundary conditions the Element Free Galerkin (EFG) and the Finite Pointset (FP) methods, as well as a modification of the latter, are examined. It is shown that the EFG-method increases the computational speed in comparison with the FP-method. It is determined that the grave disadvantage of the FP-method is a low-rate convergence while the computational complexity of each iteration is reasonable. The use of the modified FP-method shows that the elapsed time is comparable with that of the EFG-method although as the problem size increases the advantage of the FP-method is not so evident.

Mathematical formulation of the problem

The viscous gas dynamics equations can be written in the following form [5]:

$$\begin{cases} D_t\rho + \rho\left(\nabla\cdot\mathbf{V}\right) = 0, \\ \rho D_t\mathbf{V} = \rho(\mathbf{g}+\mathbf{K}) - \nabla\left(p + (2/3)\mu\left(\nabla\cdot\mathbf{V}\right)\right) + 2\left(\nabla\cdot\mu\dot{S}\right), \\ \rho c_v D_t T = \nabla\cdot(\lambda\nabla T) + 2\mu\dot{S}^2 - p\left(\nabla\cdot\mathbf{V}\right) - (2/3)\mu\left(\nabla\cdot\mathbf{V}\right)^2, \\ p = \rho R T. \end{cases} \tag{1}$$

Here t is the time, ρ is the gas density, $\mathbf{V} = (u, v, w)$ is the velocity vector with the axis projections Ox_1, Ox_2 and Ox_3 respectively, $\mathbf{g}$ is the acceleration of gravity, $\mathbf{K}$ is the Coriolis force, p is the pressure, μ is the dynamic viscosity, $\dot{S}$ is the strain rate tensor, T is the temperature, c_v and c_p are the heat capacities at constant volume and at constant pressure respectively, λ is the heat conduction coefficient, R is the absolute gas constant, the symbol D_t at a function determines the corresponding total derivative, and the bold type stands for vector quantities. To solve system (1) for a particular case, problem oriented initial and boundary conditions must be specified.

The primary goal of our research is to solve system (1) with a meshfree vortex method the heart of which is to find the solution to the Poisson equation. Thus, as a subproblem one should compare different meshfree methods in the light of convergence and speed performance. For doing that, we simplify the problem (1):

$$\begin{cases} \nabla \cdot \mathbf{V} = 0, \\ D_t \mathbf{V} = \mathbf{g} - (1/\rho_0)\nabla p + \nu \Delta \mathbf{V}, \end{cases} \tag{2}$$

where ν is the kinematic viscosity coefficient, and ρ_o is the gas density which is constant.

We use the Lagrangian discrete vortex method to find the solution to system (2). We define the vorticity as $\overline{\omega} \equiv \nabla \times \mathbf{V}$. In the subsequent discussion, the space dimension plays no crucial role. Thus, basing on the fact that $\overline{\omega}$ is normal to the plane of the flow and taking into account the potentiality of $\mathbf{g}$ the evolution of vorticity is given by

$$D_t \omega = \nu \Delta \omega. \tag{3}$$

To obtain $\mathbf{V} = (u, v)$ we are guided by the solenoidality of the velocity field:

$$\mathbf{V} = \nabla \times \psi, \text{ where } \psi \text{ is a vector potential.} \tag{4}$$

Substitution of (4) in $\overline{\omega} \equiv \nabla \times \mathbf{V}$ and bidimensionality of the equations lead to the Poisson equation

$$\Delta \psi = -\omega. \tag{5}$$

The solution of (5) and subsequent substitution of it in (4) give the required velocity vector.

Approximation of the equations and methods of numerical investigation

There is a great number of both theoretical and numerical work dedicated to various methods of solving the Poisson equation. Some methods base on using a computational grid while the others rely on the meshfree approach. Usually, the classical meshfree PIC-method [2] is used to solve the Poisson equation. It is known that the method has several drawbacks such as numerical stability, numerical dissipation etc. In this case the "truly" meshfree method is a natural choise. Two of these meshfree methods are considered in the paper, namely the Element Free Galerkin method [1] and the Finite Pointset method [4, 6, 7, 3, 8].

We consider the Poisson equation, in a simply connected domain $\Omega = [0, 1] \times [0, 1]$, which is given by

$$\Delta \psi = -\omega, \ \psi|_{\Gamma_g} = g, \ \left.\frac{\partial \psi}{\partial \mathbf{n}}\right|_{\Gamma_s} = s, \ \partial\Omega = \Gamma_g \cup \Gamma_s. \tag{6}$$

The application of the EFG-method implies using the weak form of (6):

$$\int\limits_{\Omega} \nabla\psi \cdot \nabla\chi d\Omega = \int\limits_{\Omega} \chi\omega \; d\Omega + \int\limits_{\Gamma_s} \frac{\partial\psi}{\partial\mathbf{n}}\chi d\Gamma_s + \int\limits_{\Gamma_g} (\psi - g)\,\lambda\chi d\Gamma_g + \int\limits_{\Gamma_g} \lambda\chi d\Gamma_g. \quad (7)$$

Here χ is a test function, and λ is a Lagrangian multiplier which is used to take the essential boundary conditions into consideration. In order to approximate the distribution of function ψ in the neighborhood of a point $\mathbf{x}$ over a number of randomly located nodes $\mathbf{x_i}, i = 1, 2, \ldots, n$, we define the moving least squares approximant $\psi^h(\mathbf{x})$ of ψ in the form

$$\psi^h(\mathbf{x}) = \mathbf{p}(\mathbf{x})\mathbf{a}(\mathbf{x}), \quad (8)$$

where $\mathbf{p}(\mathbf{x}) = [p_1(\mathbf{x}), p_2(\mathbf{x}), \ldots, p_m(\mathbf{x})]$ is a complete monomial basis of order m, and $\mathbf{a}(\mathbf{x}) = [a_1(\mathbf{x}), a_2(\mathbf{x}), \ldots, a_m(\mathbf{x})]$ is a vector of coefficients which are functions (to be determined) of the coordinates $\mathbf{x} = (x_1, x_2)$. By using the linear basis $\mathbf{p}(\mathbf{x}) = [1, x_1, x_2]$ and applying the standard technique for solving equation (8) we obtain

$$\psi^h(\mathbf{x}) = \sum_{i=1}^{n} \Phi_i(\mathbf{x})\hat{\psi}_i, \; \Phi_i(\mathbf{x}) = \mathbf{p}(\mathbf{x})\mathbf{A^{-1}}(\mathbf{x})\mathbf{B}(\mathbf{x})_i, \; \psi^h(\mathbf{x}_i) \equiv \psi_i \neq \hat{\psi}_i. \quad (9)$$

Here $\mathbf{A}(\mathbf{x}) = \mathbf{B}(\mathbf{x})\mathbf{P}$, $\mathbf{B}(\mathbf{x}) = \mathbf{P^T W}$, $\mathbf{P} = [\mathbf{p}(\mathbf{x_1}), \mathbf{p}(\mathbf{x_2}), ..., \mathbf{p}(\mathbf{x_n})]$, and the matrix of the weight functions $\mathbf{W} = \mathrm{diag}(w_1(\mathbf{x}), w_2(\mathbf{x}), \ldots, w_n(\mathbf{x}))$ is defined as

$$w(\mathbf{x} - \mathbf{x}_i) \equiv w(r) = \begin{cases} 2/3 - 4r^2 + 4r^3 & , r \leq 1/2, \\ 4/3 - 4r + 4r^2 - 4/3r^3 & , 1/2 < r \leq 1, \\ 0 & , r > 1. \end{cases} \quad (10)$$

The normalized distance from the point $\mathbf{x}_i$ to the point $\mathbf{x}$ is $r = \|\mathbf{x} - \mathbf{x}_i\|/d_{mi}$, and d_{mi} is the size of the domain of influence at a node. In the present implementation the test function $\chi(\mathbf{x})$ is selected to be the same as the weight function $w(r)$. Thus, substitution of (9) in (7) leads to a system of linear algebraic equations in terms of $\hat{\psi}_i$ which are assumed to be substituted in (9) to get the required solution.

Describe the FP-method in short. As well as in the previous case, the basis for the FP-method is the weighted least squares approximation. The essence of the FP-method is to approximate a function $\psi(\mathbf{x}_i)$ and its derivatives around $\mathbf{x}$ using Taylor's series expansion:

$$\psi(\mathbf{x}_i) = \sum_{j=0}^{\infty} \frac{1}{j!} (\mathbf{x}_i - \mathbf{x}, \nabla)^j \, \psi(\mathbf{x}). \quad (11)$$

Basing on equation (11) and restricting our considerations to a second order accuracy and the number of points (m) in the sub-domain we lead to a system of linear algebraic equations. The system is overdetermined and underdetermined

for $m+k>6$ and $m+k<6$ respectively. The value k is supposed to be 1 or 2 for the Dirichlet and the Neumann boundary value problems. The solution to the system is found with the weighted least squares method using the weight function (10). To find the solution with the FP-method the following iteration process is used

$$\psi^{(\tau)}(\mathbf{x}_i)=\sum_{j=0}^{2}\frac{(\mathbf{x}_i-\mathbf{x},\nabla)^j}{j!}\psi^{(\tau+1)}(\mathbf{x}), i=1,2,\ldots,m, \tau\geq 0, \psi^{(0)}(\mathbf{x}_i)=0. \quad (12)$$

The system (12) must satisfy the Poisson equation at each point. For this purpose, it is necessary to add the following equation

$$\Delta\psi^{(\tau+1)}=-\omega \quad (13)$$

to the system (12) for the Dirichlet boundary value problem, as well as

$$\left.\frac{\partial\psi^{(\tau+1)}}{\partial\mathbf{n}}\right|_{\Gamma_s}=s \quad (14)$$

in case of the Neumann boundary value problem. Thus, the final system of linear algebraic equations at $\mathbf{x}$ is given by

$$\psi^{(\tau+1)}(\mathbf{x})=\mathbf{D}^{-1}(\mathbf{M}^{\mathrm{T}}\mathbf{W})\left[\psi^{(\tau)}(\mathbf{x}_i),-\omega,s\right]^{\mathrm{T}}, i=1,2,\ldots,m\ , \quad (15)$$

where the matrix $\mathbf{M}$ is determined from (12)-(14), and $\mathbf{D}=\mathbf{M}^{\mathrm{T}}\mathbf{W}\mathbf{M}$.

The bottom line is the FP-method is local, i.e. it depends on the sub-domain only. The distribution of points in the sub-domain can be irregular that provides the method with all features of the meshfree methods. Furthermore, the local iteration process gives considerable advantages in parallelization of the algorithm but, as it is shown below, it degradates the performance of computations. Generally speaking, to iterate it is necessary to know not the whole solution vector of system (15) but its first component only. This circumstance was the cause of modification of the FP-method [6]. The heart of the modification is to apply an iterationless technique to find the solution. Running a few steps forward we would like to note that this iterationless method reduces the computation time dramatically.

As we have to find the first component of the solution vector, the authors of [6] propose to get an explicit equation for the component at the point $\mathbf{x}$ basing on (15). Thus, for all the points of the domain we can obtain a system of linear algebraic equations the further inversion of which gives us the desired solution.

Results of the investigation

In the section we briefly present the results of comparison between the EFG and the FP methods for the two-dimensional Poisson equation with homogeneous

boundary conditions of Dirichlet and Neumann types. The comparison is done in the following way. For each fixed grid size N we determine the relative error between the exact and numerical solutions for the EFG and the modified FP methods. The iteration number $|\tau|$ of the original FP-method is chosen so that for the corresponding N the value $\min\limits_{|\tau|}(\varepsilon_{FPM}) \approx \varepsilon_{EFGM}$. For the sake of simplicity, we use a uniform distribution of points in the domain that does not narrow the scope of the research.

First we consider the Dirichlet boundary value problem

$$\Delta\psi = \nabla \cdot \nabla\phi, \;\; \psi|_{\partial\Omega} = 0, \;\; \phi(x_1, x_2) \equiv \psi_{\text{exact}}(x_1, x_2). \tag{16}$$

Summarize the results of the comparison in table 1.

Table 1. Comparison between the relative error ε and the elapsed time T for the EFG-method, the FP-method and its modification

N	ε_{EFGM}	ε_{MFPM}	ε_{FPM}	$\lvert\tau\rvert$	T_{FPM}/T_{EFGM}	T_{EFGM}/T_{MFPM}
16×16	0.476026	0.173516	0.478233	3	0.37	6.50
32×32	0.101127	0.040869	0.101431	58	7.36	3.16
64×64	0.021170	0.009743	0.021162	780	107.11	0.80

As it follows from table 1, the EFG-method surpasses the FP-method in computation speed. However, the EFG-method is inferior to the modified FP-method in both computation speed and accuracy. Although, as the problem size increases the difference in the computation speed between the EFG and the modified FP methods is not so evident. The further comparison of the methods has been done for homogeneous boundary conditions of Neumann type, i.e.

$$\Delta\psi = \nabla \cdot \nabla\phi, \;\; \left.\frac{\partial\psi}{\partial\mathbf{n}}\right|_{\partial\Omega} = 0, \;\; \phi(x_1, x_2) \equiv \psi_{\text{exact}}(x, y). \tag{17}$$

The comparison between both methods is represented in table 2.

¿From table 2, we can draw a conclusion that even for $N = 64 \times 64$ both methods have satisfactory accuracy. As well as in the previous case, the computational speed of the FP-method in comparison with that of the EFG-method remains low. The analysis of the modified FP-method shows that it operates faster then the EFG-method although, in some case, its computation time is higher.

Conclusion

The investigation has shown that the FP-method is rather slow for all the examples considered in the paper. That is bound up with its low-rate convergence.

Table 2. Comparison between the relative error and the elapsed time T for the element-free Galerkin method, the FP-method and its modification

N	ε_{EFGM}	ε_{MFPM}	ε_{FPM}	$\lvert\tau\rvert$	T_{FPM}/T_{EFGM}	T_{EFGM}/T_{MFPM}
16×16	2.277533	1.664913	0.741099	2	1.50	1.00
32×32	0.379331	0.472410	0.062771	3	1.88	1.25
64×64	0.084967	0.206072	0.077314	14	4.59	1.12

The modified FP-method shows the highest computation speed with regards to almost all the test examples save for the largest grid size for the Dirichlet boundary value problem. In spite of that, any theoretical issues on its stability analysis, convergence rate and accuracy at present moment are not known to the author. Thus, basing upon the presented and known results concerning the modified FP-method it is possible to draw a conclusion that the method can be applied to solving large scientific problems. Despite this, a deeper theoretical analysis is needed, which will be the primary intent of our future research.

The author is grateful to Dr. S. Tiwari for helpful discussions and valuable suggestions.

References

1. Belytschko, T., Lu, Y., Gu, L.: Element free Galerkin methods. International Journal for Numerical Methods in Engineering. **37** (1994) 229–256.
2. Harlow, F.H.: Particle-in-cell computing method for fluid dynamics. Meth. Comput. Phys. **3** (1964) 319–343.
3. Hietel, D., Junk, M., Kuhnert, J., Tiwari, S.: Meshless methods for Conservation Laws. Analysis and Numerical Methods for Conservation Laws. G. Warnecke (Eds). Springer. (2005).
4. Kuhnert, J., Tiwari, S.: Grid Free method for Poisson Equation. Wavelet Analysis and Applications. New Age International (P). Ltd. Publishers. (2004).
5. Landau, L.D., Lifshitz, E.M.: Gidrodynamica. Nauka. Moskva. (1988).
6. Tiwari, S., Kuhnert, J.: A numerical scheme for solving incompressible and low mach number flows by Finite Pointset method, First Indo-German Conference on PDE, Scietific Computing and Optimization in Applications. Trier. (2004).
7. Tiwari, S., Kuhnert, J.: A numerical scheme for solving incompressible and low Mach number flows by Finite Pointset Method. Lecture Notes in Computational Science and Engineering. **43**. (2005) (Meshfree Methods for Partial Differential Equations II), Griebel M., Schweitzer M. A. (Eds), Springer.
8. Tiwari, S., Antonov, S., Hietel, D., Kuhnert, J., Wegener, R.: A Meshfree Method for Simulations of Interactions between Fluids and Flexible Structures, Lecture Notes in Computational Science and Engineering. **57** (2006) (Meshfree Methods for Partial Differential Equations III), Griebel, M., Schweitzer, M. A. (Eds), Springer.

On the Iterative Analysis of the Generalized Dirichlet-Neumann Map for Elliptic PDEs*

A. G. Sifalakis[1], S. R. Fulton[2], E. P. Papadopoulou[1], and Y. G. Saridakis[1]

[1] Dept of Sciences, Technical University of Crete, 73100 Chania, Greece
[2] Dept of Math and CS, Clarkson University, Potsdam NY 13699-5815, USA

Abstract. Taking advantage of the structural properties of the Collocation coefficient matrix associated with the Dirichlet-Neumann map for linear elliptic PDEs, we present a complete spectral analysis for the Laplace's equation on a square domain with the same type of boundary conditions on all sides. Through this analysis we are able to recover optimal classical SOR and Krylov iterative methods.

Introduction

Recently, Fokas[1, 4] introduced a new unified approach for analyzing linear and integrable nonlinear PDEs. Central issue to this approach is a generalized Dirichlet to Neumann map, characterized through the solution of the so-called *global relation*, namely, an equation, valid for all values of a complex parameter k, coupling specified known and unknown values of the solution and its derivatives on the boundary. In particular, for the case of Laplace's equation, $q_{z\bar{z}} = 0$, in a convex bounded polygon D with vertices $z_1, z_2, \ldots, z_n$ (modulo n) indexed counter-clockwise, the associated *Global Relation* takes the form (see also [2, 3])

$$\sum_{j=1}^{n} \int_{S_j} e^{-ikz} q_z dz = 0, \quad k \in \mathbb{C}\,, \tag{1}$$

where S_j denotes the side from z_j to z_{j+1} (not including the end points). If, for $z \in S_j$, $1 \leq j \leq n$, we now let $g^{(j)}$ denote the derivative of the solution in the direction making an angle $\beta_j, 0 \leq \beta_j \leq \pi$ with the side S_j, namely : $\cos(\beta_j)\, q_s^{(j)} + \sin(\beta_j)\, q_n^{(j)} = g^{(j)}$, and $f^{(j)}$ denote the derivative of the solution in the direction normal to the above direction, namely : $-\sin(\beta_j)\, q_s^{(j)} + \cos(\beta_j)\, q_n^{(j)} = f^{(j)}$, where $q_s^{(j)}$ and $q_n^{(j)}$ denote the tangential and (outward) normal components of q_z along the side S_j, then the *Generalized Dirichlet-Neumann map*, that is the relation between the sets $\left\{f^{(j)}(s)\right\}$ and $\left\{g^{(j)}(s)\right\}_{j=1}^{n}$, is characterized by the single equation

$$\sum_{j=1}^{n} |h_j|\, e^{i(\beta_j - k m_j)} \int_{-\pi}^{\pi} e^{-ikh_j s} \left(f^{(j)} - i g^{(j)}\right) ds = 0, \qquad k \in \mathbb{C} \tag{2}$$

* This work was supported by the Greek Ministry of Education "Herakleitos" EPEAEK Grant which is partially funded by the EU

where, for $j = 1, 2, \ldots, n$, and $z_{n+1} = z_1$,

$$h_j := \frac{1}{2\pi}\left(z_{j+1} - z_j\right) \quad , \quad m_j := \frac{1}{2}\left(z_{j+1} + z_j\right) \quad , \quad s := \frac{z - m_j}{h_j} \, . \tag{3}$$

For the numerical solution of the Generalized Dirichlet-Neumann map in (3), a Collocation-type method has been developed (see [2] and [3]) : Suppose that the set $\left\{g^{(j)}\left(s\right)\right\}_{j=1}^{n}$ is given through the boundary conditions, and that $\left\{f^{(j)}\left(s\right)\right\}_{j=1}^{n}$ is approximated by $\left\{f_N^{(j)}\left(s\right)\right\}_{j=1}^{n}$ where

$$f_N^{(j)}\left(s\right) = f_*^{(j)}\left(s\right) + \sum_{r=1}^{N} U_r^j \varphi_r(s) \, , \tag{4}$$

with N being an even integer, $2\pi f_*^{(j)}\left(s\right) := \left(s+\pi\right) f^{(j)}\left(\pi\right) - \left(s-\pi\right) f^{(j)}\left(-\pi\right)$ (the values of $f^{(j)}(\pi)$ and $f^{(j)}(-\pi)$ can be computed by the continuity requirements at the vertices of the polygon), and the set of functions $\left\{\varphi_r\left(s\right)\right\}_{r=1}^{N}$ being the *basis* functions. If we evaluate equation (3) on the following n-rays of the complex k-plane : $k_p = -\frac{l}{h_p}, \quad l \in \mathbb{R}^+, \quad p = 1, \ldots, n$, then the real coefficients U_r^j satisfy the system of linear algebraic equations

$$\sum_{j=1}^{n} \frac{|h_j|}{|h_p|} e^{i(\beta_j - \beta_p)} e^{-i\frac{l}{h_p}(m_p - m_j)} \sum_{r=1}^{N} U_r^j \int_{-\pi}^{\pi} e^{il\frac{h_j}{h_p}s} \varphi_r(s) ds = G_p\left(l\right) \tag{5}$$

where $G_p(l)$ denotes the known function

$$G_p\left(l\right) = i \sum_{j=1}^{n} \frac{|h_j|}{|h_p|} e^{i(\beta_j - \beta_p)} e^{-i\frac{l}{h_p}(m_p - m_j)} \int_{-\pi}^{\pi} e^{il\frac{h_j}{h_p}s} \left(g^{(j)}\left(s\right) + i f_*^{(j)}\left(s\right)\right) ds \, , \tag{6}$$

and l is chosen as follows: $l = \frac{1}{2}, \frac{3}{2}, \ldots, \frac{N-1}{2}$ and $l = 1, 2, \ldots, \frac{N}{2}$ for the real and imaginary part of equations (7), respectively, defining a set of *Collocation points*.

The Case of Square Domains

In this section we summarize some of the results included in [5], pertaining to the case of regular polygon domains (namely $|h_j| = h$ for all j), and, in particular, square domains ($n = 4$) having the same type of boundary conditions on all sides (namely $\beta_j = \beta$ for all j). For this case, one may easily verify that the Generalized Dirichlet-Neumann map in (5) reduces to

$$\sum_{j=1}^{4} e^{-i\frac{l}{h_p}(m_p - m_j)} \sum_{r=1}^{N} U_r^j \int_{-\pi}^{\pi} e^{il\frac{h_j}{h_p}s} \varphi_r(s) ds = G_p\left(l\right) \, , \tag{7}$$

where $G_p(l)$ is as in (6) simplified analogously. The resulting linear system is

$$A\boldsymbol{U} = \boldsymbol{G} \quad , \quad A \in \mathbb{R}^{4N,4N} \quad , \quad \boldsymbol{U}, \boldsymbol{G} \in \mathbb{R}^{4N} \quad , \tag{8}$$

where the Collocation coefficient matrix is in the block partitioned form (cf. [5])

$$A = \begin{pmatrix} A_0 & O & A_1 & O \\ O & A_0 & O & A_1 \\ A_1 & O & A_0 & O \\ O & A_1 & O & A_0 \end{pmatrix} \quad , \quad A_0, A_1 \in \mathbb{R}^{N,N} \ , \tag{9}$$

with

$$A_1 = DA_0 \quad , \quad D = diag(d_1, \ldots, d_N) \ , d_r = (-1)^{r-1} e^{-r\pi} \quad , \quad r = 1, \ldots, N \ . \tag{10}$$

Furthermore, the elements of the block diagonal submatrix A_0 are defined through the Finite Cosine/Sine Fourier Transform of the basis functions $\phi_r(s)$:

$$A_0 = (a_{q,r}) \quad , \quad a_{q,r} = \begin{cases} \int\limits_{-\pi}^{\pi} \cos(\frac{q}{2}s)\phi_r(s)ds \quad , \quad q = \text{odd} \\ \\ \int\limits_{-\pi}^{\pi} \sin(\frac{q}{2}s)\phi_r(s)ds \quad , \quad q = \text{even} \end{cases} \ , \tag{11}$$

and, as $\phi_r(s)$ are assumed appropriately chosen real linearly independent functions, A_0 is assumed nonsingular. We remark that, for the case of sine basis functions (cf. [3]), that is $\phi_r(s) = \sin(r\frac{s+\pi}{2})$, A_0 is a nonsingular diagonal matrix.

As it is shown in [5], the associated with A block Jacobi iteration matrix T_0 is obviously weakly cyclic of index 2 and *similar* to the matrix

$$S = -\begin{pmatrix} O & O & D & O \\ O & O & O & D \\ D & O & O & O \\ O & D & O & O \end{pmatrix} \tag{12}$$

where D is as defined in (10). Therefore the spectrum $\sigma(\cdot)$ of T_0 satisfies

$$\sigma(T_0) = \{\pm e^{-r\pi} \ , \ \pm e^{-r\pi}\}_{r=1}^{N} \ . \tag{13}$$

Evidently,

$$\varrho(T_0) = e^{-\pi} \approxeq 0.0432 \quad , \tag{14}$$

where $\varrho(\cdot)$ denotes the spectral radius. Taking advantage of the weakly cyclic structure of T_0 and well known results from the literature, the Gauss-Seidel and the optimal SOR iterative methods, with iteration matrices T_1 and T_{ω_b} respectively, satisfy (cf. [5])

$$\begin{aligned} \varrho(T_1) &= e^{-2\pi} \approxeq 0.0019 \\ \varrho(T_{\omega_b}) &= \frac{2}{1 + \sqrt{1 - e^{-2\pi}}} - 1 \approxeq 0.0005 \end{aligned} \quad . \tag{15}$$

The above spectral analysis has been also used in [5] to determine the spectra of various preconditioned matrices in order to determine effective convergence properties of Krylov subspace methods. There, it is shown that the spectra of the Jacobi, the Gauss-Seidel, and the Symmetric Gauss-Siedel preconditioned matrix A are all real and clustered around unity. Hence, following [7], the Bi-CGSTAB[6], combined with the above preconditioning schemes, is the method of preference. For a complete analysis see [5].
At this point we remark that the above spectral analysis is independent of the choice of basis functions as well as independent of the type of boundary conditions.
To numerically demonstrate the above results we include Table 1 referring to the performance of all mentioned iterative methods when they apply to the model problem considered in [3] for the case of Chebyshev basis functions (see [3]).

Table 1 Performance of Iterative Methods

Method	Precondi-tioner	$N = 8$			$N = 16$		
		Error	Iter.	Time	Error	Iter.	Time
Jacobi	—	2.09e-05	15	1.24e-03	5.78e-13	14	2.02e-03
Gauss-Seidel	—	2.09e-05	9	1.10e-03	5.78e-13	9	1.52e-03
SOR	—	2.09e-05	7	4.50e-04	5.78e-13	7	8.43e-04
Bi-CGSTAB	—	2.09e-05	10	1.16e-03	5.78e-13	27	2.67e-03
	Jacobi	2.09e-05	3	1.21e-03	5.78e-13	3	1.49e-03
	Gauss-Seidel	2.09e-05	3	1.18e-03	5.78e-13	3	1.88e-03
	Sym. Gauss-Seidel	2.09e-05	2	1.42e-03	5.78e-13	2	1.90e-03
GMRES(10)	—	2.09e-05	13	1.29e-03	5.78e-13	125	9.73e-03
	Jacobi	2.09e-05	11	1.52e-03	5.78e-13	11	2.43e-03
	Gauss-Seidel	2.09e-05	11	1.61e-03	5.78e-13	7	1.82e-03
	Sym. Gauss-Seidel	2.09e-05	7	1.55e-03	5.78e-13	7	2.47e-03

References

1. A.S.Fokas, A unified transform method for solving linear and certain nonlinear PDEs, *Proc. R. Soc. London A***53** (1997), 1411-1443.
2. S. Fulton, A.S. Fokas and C. Xenophontos, An Analytical Method for Linear Elliptic PDEs and its Numerical Implementation, *J. of CAM* **167** (2004), 465-483.
3. A. Sifalakis, A.S. Fokas, S. Fulton and Y.G. Saridakis, The Generalized Dirichlet-Neumann Map for Linear Elliptic PDEs and its Numerical Implementation, *J. of Comput. and Appl. Maths.* (in press)
4. A.S.Fokas, Two-dimensional linear PDEs in a convex polygon, *Proc. R. Soc. London A* **457** (2001), 371-393.
5. A. Sifalakis, S. Fulton, E.P. Papadopoulou and Y.G. Saridakis, On the Iterative Analysis of the Generalized Dirichlet-Neumann Map for Elliptic PDEs, (in preparation)
6. H.A. Van Der Vorst, Bi-CGSTAB: A fast and smoothly converging variant of Bi-CG for the solution of nonsymmetric linear systems, *SIAM J. Sci. Statist. Comput.*, 13,1992, pp. 631-644.
7. J. Dongarra, I. Duff, D. Sorensen, H. van der Vorst, *Numerical Linear Algebra for High-Performance Computers* , SIAM, 1998.

A greedy approach to transversal selection for nonlinear systems of equations

D.G. Sotiropoulos[1] and I.E. Livieris[2]

[1] Ionian University, Department of Informatics, Corfu, Greece
dgs@ionio.gr
[2] University of Patras, Department of Mathematics, Patras, Greece
livieris@math.upatras.gr

Abstract. Interval methods have been established for rigorously bounding all solutions of a nonlinear system of equations within a given region. In this paper, we introduce a new method for determining a good pivoting sequence for Gauss-Seidel method, based on a greedy algorithm, called 4M, solving assignment problems with worst case complexity $O(n^2)$.

Key words: nonlinear systems, interval arithmetic, interval Newton methods, perfect matching, transversal.

Introduction and Motivation

We consider the problem of finding with certainty *all* zeros of a nonlinear continuously differentiable function $f : \mathbb{R}^n \to \mathbb{R}^n$ in a given interval vector $[x] \in \mathbb{IR}^n$ (an n-dimensional box). This problem is difficult due to its inherent computational complexity (NP-hard) and due to numerical issues involved to guarantee correctness. Interval Newton methods have been established for finding all real roots within the specified domain of a nonlinear system with both mathematical and computational certainty. In such methods, the basic idea is to apply the interval Gauss-Seidel method to the preconditioned linearized system [12]. Nowadays, the *interval Gauss-Seidel method* [5, 13] serves as a basis not only for interval Newton algorithms (see [4, Chapter 13]) but also for interval constraint algorithms [6]. However, this method is "blind" since it works in a straightforward manner, without taking account the coupling between the variables and functions. A natural question that arises is: if it is possible to dynamically accelerate the convergence rate of nonlinear Gauss-Seidel method, that still remains an open problem.

Herbort & Ratz [7] introduced the problem in their attempt to develop a new componentwise Newton operator, using a univariate Newton iteration on a unary projection of f_i onto one of the variables $x_1, \ldots, x_n$. Actually, finding such an assignment set is known as finding a *transversal* in the incidence matrix associated with the problem. Thus, a transversal may not be unique, but finding a "good" one is not a trivial problem. Sotiropoulos et al. [14] suggested to compute a transversal as a preprocessing step to interval Gauss-Seidel method for solving

polynomial systems based on the structure of the equations. In recent works, Goualard [2] and Goulard & Jermann [3] investigated the impact of the selection of a transversal on the speed of convergence of interval methods based on the nonlinear Gauss-Seidel procedure. In [3], the authors concluded that it is not possible to select statically a good transversal at the beginning of the solving process, and therefore, the selection must be reconsidered dynamically at each iteration of the solving process.

In this work, we propose a greedy algorithm which determines a transversal dynamically by exploring information not only from the *incidence matrix* (static) but also from the current subregion (dynamic). The algorithm has the advantage that it does not use any first order information, in contrast to the previous proposed algorithms [7, 2, 3, 14]. From our point of view, the selection of an transversal can be seen as a *matching problem* on the bipartite graph $G = (\mathcal{F}, \mathcal{X}, E)$ associated with the the incidence matrix of the nonlinear system.

The greedy algorithm `4M`

Given a system of nonlinear equations of the form

$$f_i(x_1, x_2, \ldots, x_n), \quad 1 \leq i \leq n \tag{1}$$

where the variables $x_j,\ j = 1, \ldots, n$ are bounded by real intervals i.e. $x_j \in [x_j]$. The associated *incident matrix* $A = [a_{ij}]$ of the nonlinear system (1) is a zero-one matrix where a_{ij} is set to 1 if variable x_j occurs in function f_i. We represent the incidence matrix with a bipartite graph $G(\mathcal{F}, \mathcal{X}, E)$ where each vertex $f_i \in \mathcal{F}$ and $x_j \in \mathcal{X}$ corresponds to the function f_i and the variable x_j of the nonlinear system (1), respectively. In order to seriate the vertices of each vertex set we apply an ordering relation between them, associating a tuple, consists of information from the nonlinear system. Specifically, for each vertex $f_i \in \mathcal{F}$ we associate the tuple $\{d_G(f_i), w([f]_i)\}$ and for each vertex $x_j \in \mathcal{X}$ the tuple $\{w([x]_j), d_G(x_j)\}$.

The algorithm is iterative and requires as input only the bipartite graph $G\,(\mathcal{F}, \mathcal{X}, E)$. At each iteration, Algorithm-`4M` selects the vertex $f_i \in \mathcal{F}$ with the minimum degree. If this vertex it is not unique, we select the least of them according to the tuple of each vertex. Afterwards, it is matched with its best neighbor that is, the vertex with the biggest tuple, in $O(n^2)$. The two matched vertices are eliminated from further processing and consequently the degree of each of their neighbors is decreased by one. The edge incident to the selected vertices is inserted in the matching set $\mathcal{M}$. Further more at any iteration of this process, if any vertex in set $\mathcal{X}$ has degree equal to one then it is matched with its unique neighbor and the vertices are removed from the graph, as well. The iterations are repeated until that bipartite graph is empty, in $O(n^2)$. Finally, every edge (f_i, x_j) in $\mathcal{M}$ is sorted according to the tuple of each vertex x_j.

Every edge of set $\mathcal{M}$ determines the transversal of the nonlinear system at the current iteration. In particular, the edge $(f_i, x_j) \in \mathcal{M}$, represents the element (i, j) of the incidence matrix, which implies, in terms of nonlinear system, that variable x_j shall be projected onto the function f_i.

We establish that the feasibility of the matching problem on the bipartite graph G ensures the existence of the perfect matching, not its uniqueness. Our proposed algorithm looks for the appropriate perfect matching among all others, nevertheless, all other existing perfect matchings don't satisfy the primitives of the greedy selection we have defined.

Lemma 1. *Suppose the bipartite graph $G(\mathcal{F}, \mathcal{X}, E)$, arising by the representation of the incidence matrix of a nonlinear system (1), where $\mathcal{F}$ is the set that corresponds to the functions, $\mathcal{X}$ is the set that corresponds to the variables and $E = \mathcal{F} \times \mathcal{X}$. Algorithm-4M achieves a perfect matching of graph G, in $O(n^2)$.*

Note that once a vertex is matched and removed from the bipartite graph, it is never revisited by the algorithm and all the other unmatched edges incident on it are removed from the graph, thus proving the correctness of the algorithm. Further, the algorithm returns a perfect match, that is because once two vertices are matched, they remain matched until the end of the process.

Numerical results

In this section, we introduce experimental results in order to demonstrate the acceletating of the efficiency of interval Gauss-Seidel method using our proposed algorithm as a preprocessing step (4M+Gauss-Seidel) and comparing it with the traditional method (Gauss Seidel) in a variate of benchmarks. The test problems have been taken from numerical [1, 8, 11] and interval analysis [7, 10] papers.

The implementation has been carried out in C++ using the C-XSC 2.0 library [9]. We present illustrative examples, highlighting the performance and the superiority of our proposed method.

Solving process		Gauss-Seidel			4M+Gauss-Seidel		
No.	Problem	FcEv	JcEv	Ps	FcEv	JcEv	Ps
1.	Floudas	17167	13306	3845	1351	989	345
2.	Ex.Powell	259284	168471	90805	138977	94499	44470
3.	Cyclohexane	n/a	n/a	n/a	44747	36017	8713
4.	Kinematics	17155	13164	3975	2340	1757	567
5.	Powell	6591088	5459059	1132028	2642	2123	517
6.	Caprasse	1520810	1329956	190815	170918	149249	21633
7.	Brown5	125092	93113	31976	20481	14959	5519
8.	Economics5	567619	474293	93321	47728	39351	8373
9.	Economics6	n/a	n/a	n/a	389061	336795	52262
10.	6Body	n/a	n/a	n/a	210983	176067	34911

Table 1 compares the results of the two solving processes applied to ten test problems. The first row denotes the method that has been used as solving process for isolating all zeros of a nonlinear system. For each test problem we list the

number of function (`FcEv`) and jacobian (`JcEv`) evaluations and the number of pruning steps (`Ps`). A "`n/a`" in a column means that the solver was unable to find all the solutions of the problem within two hours. It is shown that, our proposed technique contributing in decreasing of both function and jacobian evaluations and of pruning steps.

Conclusion

In this paper, we have proposed a new direction of research, merging combinatorial matching theory and a greedy based technique for obtaining a maximum transversal. Our future framework will be to access our technique to interval constraint solvers and devise more advanced structure-based heuristics.

References

1. D. Bini and B. Mourrain. Handbook of polynomial systems. November 1996.
2. F. Goualard. On considering an interval constraint solving algorithm as a free-steering nonlinear gauss-seidel procedure. In *SAC '05: Proceedings of the 2005 ACM symposium on Applied computing*.
3. F. Goualard and C. Jermann. On the selection of a transversal to solve nonlinear systems with interval arithmetic. In *Procs. of the International Conference on Computational Science 2006*, volume 3991 of *Lecture Notes in Computer Science*, pages 332–339. Springer-Verlag, 2006.
4. R. Hammer, M. Hocks, U. Kulisch, and D. Ratz. *C++ Toolbox for Verified Computing I, Basic Numerical Problems: Theory, Algorithms, and Programs*. Springer-Verlag, 1995.
5. E. R. Hansen and S. Sengupta. Bounding solutions of systems of equations using interval analysis. *BIT*, 21:203–211, 1981.
6. P. Van Hentenryck, D. McAllester, and D. Kapur. Solving polynomial systems using a branch and prune approach. *SIAM J. Numer. Anal.*, 34(2):797–827, 1997.
7. S. Herbort and D. Ratz. Improving the efficiency of a nonlinear–system–solver using a componentwise newton method. Technical Report Bericht 2/1997, Institut für Angewandte Mathematik, Universität Karlsruhe (TH), 1997.
8. R. B. Kearfott. Some tests of generalized bicection. *ACM Transactions on Mathematical Software*, 13:197–220, 1987.
9. R. Klatte, U. Kulisch, C. Lawo, M. Rauch, and A. Wiethoff. *C-XSC – A C++ Class Library for Extended Scientific Computing*. Springer-Verlag, Heidelberg, 1993.
10. Van Hentenryck P. Michel L. and Deville Y. *Numerica: A Modeling Language for Global Optimization*. MIT Press, 1997.
11. K. Meintjes and A. P. Morgan. Chemical equilibrium systems as numerical test problems. *ACM TOMS*, 16(2):143–151, 1990.
12. A. Neumaier. *Interval Methods for systems of equations*. Cambridge University Press, 1990.
13. D. Ratz. *Automatische Ergebnisverifikation bei globalen Optimierungsproblemen*. PhD thesis, Universität Karlsruhe, 1992.
14. D.G. Sotiropoulos, J.A. Nikas, and T.N. Grapsa. Improving the efficiency of a polynomial system solver via a reordering technique. In *Proceedings of 4th GRACM Congress on Computational Mechanics (GRACM 2002)*, pages 970–976, 2002.

The use of orthogonal Bergman polynomials for the reconstruction of planar domains

Nikos Stylianopoulos

University of Cyprus

Abstract. We describe and analyze a stable algorithm for the reconstruction of bounded simply-connected planar domains from moments. This algorithm is based on (a) the computation of the associated Bergman orthogonal polynomials from the moments, by using an Arnoldi- type version of the Gram-Schmidt process and (b) the approximation of the boundary of the domain, by using the asymptotic properties of the Bergman polynomials. The performance of the algorithm is demonstrated by a number of numerical examples.

Combining Evolutionary and Stochastic Gradient Techniques for System Identification

Konstantinos Theofilatos[1], Grigorios Beligiannis[2], and Spiridon Likothanassis[1]

[1] University of Patras, Department of Computer Enginnering and Informatics, 26500 Rio, Greece,
theofilk@ceid.upatras.gr, likothan@ceid.upatras.gr
[2] University of Ioannina, Department of Business Administration in Food and Argicultural Enterprises, G. Seferi 2, 30100 Agrinio, Greece
gbeligia@cc.uoi.gr

Abstract. The method presented in the present contribution is a method of ARMA system identification using a hybrid algorithm which combines GAs and the LMS algorithm. LMS is used in the step of the evaluation of the fitness function in order to enhance the chromosomes produced by the GA. Furthermore, when the GA is terminated, the LMS is applied again in order to optimize the final result. In this way we utilize the advantages of both algorithms. Simulation results demonstrate the effectiveness and efficiency of the proposed method by comparing it with other methods presented in the literature.

Introduction

Using the term system identification (modeling), we mean locating a model with known structure and parameters, which verges on the operation of a real system with unknown characteristics. Modelling is one of the main methodological tools in science and in some sectors of scientific research. Furthermore, scientific theories and knowledge are based on theoretical models. On the other hand, modeling activities constitute a substantial learning procedure. From previous remarks, one can understand the great role of modeling in the evolution of science.

GAs have been used extensively in the identification of unknown systems (modeling). Specifically, they have been combined with many other classic optimization methods and as a result many very effective hybrid algorithms have been created. One such hybrid algorithm, which is made by combining GAs with RLS is described in [1]. The method is called GARLS and provides quite satisfactory results. Furthermore, in [2] an effective modeling method is created combining GAs with Simulated Annealing. Finally, in [3], the RGO method is presented which is a variation of GAs in order to make semi-local search without being caged in local extremas.

The method presented in the present contribution is a method of ARMA system identification using a hybrid algorithm. This hybrid algorithm combines GAs

and the LMS algorithm. LMS is used in the step of the evaluation of the fitness function in order to enhance the chromosomes produced by the GA. Furthermore, when the GA is terminated, we apply LMS again in order to optimize the final result. In this way we utilize the advantages of both algorithms. Specifically, GAs are applied to optimization problems with huge search space and many local extremas, while LMS is used for local search in which it is very effective.
Except from the present contribution, in past, there have been some other efforts to combine GAs with the LMS algorithm in order to solve the problem of modeling unknown systems. For example, in [4], a method that uses LMS to create the population of the first generation of the GA is proposed.

The Proposed Hybrid Algorithm

Basic Ideas

The proposed method is inspired by [4]. According to this method gradient methods are used to initialize a GA which is used to find the final solution. So, the GA is used for local search. However, GAs, as known, have better performance when the surface in which they search is big with a lot of local optima, while gradient methods are used for local search. So, we thought, that if we use a GA to initialize a gradient method (in our case the LMS algorithm) we might have better results. From the method presented in [4], our method uses the gradient method. This is a simple LMS, which gives an MA estimation of the System which we try to identify. The reasons for choosing LMS as our gradient method are the following: a) it is simple to implement, b) it is on-line, c) it converges fast, d) it has small computational complexity.
Next, using the algorithm of Hartmut Brandestein and Rolf Unbehauen, the MA model is transformed to an ARMA model whose order is defined by the GA. Considering that an IIR system is the limit of an FIR system we conclude that, under certain assumptions, our algorithm is expected to give satisfactory results. We suppose that the unknown models that we are going to identify have, without loss of generality, the same number of coefficients both on the numerator and on the denominator.This assumption is not expected to reduce the performance and the flexibility of the system, because if a coefficient does not exist in the system that we try to identify, we expect the algorithm to assign a near zero value to it.
Finding the system's order is a very difficult task with a lot of local optima. For this reason, we use the GA to solve it. For simplicity and in order to reduce the complexity of the algorithm, the order of the system is supposed to be within the interval [1,10], which indeed includes a great number of systems. Another problem that we have to deal with is that we do not know the specific order of the MA system which we should use for every model in order to take satisfactory results. We solved this problem by using linked lists as chromosomes. In this way the chromosomes are of varying length and we can use the GA to solve the problem of finding the best order for the MA model.

Explicit Description of the Proposed Algorithm

The proposed method constitutes of four phases which are described below:
Phase 1: In this phase, the algorithm creates the initial population which is going to be used by the GA. Every chromosome is a type List object of the C++ Genetic Algorithms' library GAlib [5]. This lets us use chromosomes with varying length. The first node of the list contains an integer number from 1 to 10 which corresponds to the order of the ARMA system. The other nodes contain elements of type double which correspond to the MA filters' parameters which is going to be used as an initial filter for the LMS algorithm.
Phase 2: In this phase, the GA is applied to the initial population created during the previous phase. The GA used is the simple Genetic Algorithm provided by GALIB (GASimpleGA) [5]. The algorithm does not use overlapping populations and implements the parents' selection using the roulette wheel selection method. Finally, this specific algorithm uses the elitism technique, which assures that the best chromosome of each generation is included in the next generation's population.
Phase 3: In this phase the proposed algorithm follows the following steps:

(1.) The LMS algorithm is applied for data of length dlms with initial parameters the ones located in the best chromosome computed by the GA. So, an MA filter is created.
(2.) Using the algorithm of Hartmut Brandestein and Rolf Unbehauen the MA filter is transformed in an ARMA filter with order equal to the number that is the first element of the chromosome.
(3.) The mean square error (MSE) of the ARMA system for the specific data window is computed.

The ARMA filter taken from the previous procedure is the final filter that the proposed algorithm proposes as the best solution.
Phase 4: In this phase the proposed method checks the stability of the final solution.

Experimental Results

In order to demonstrate the effectiveness and efficiency of the proposed algorithm we conducted many experiments on different models and compared it with many different approaches presented in the literature. In the current contribution we present the application of the proposed algorithm on the ARMA system identification problem which is referred in [6], where two algorithms are presented for system identification, which are named OPS and FOS respectively. Both algorithms need to know 'a priori' the order of the system which is to be identified. The system described is used to test these algorithms in cases where the order of the system is chosen to be 5 (instead of 4 which is the real order) for the denominator and 6 for the numerator (instead of 4 which is the real order). Many experiments were conducted using additive noise of 10db and without additive

noise. The estimations given by the algorithms OPS, FOS are not close enough to the real system. We tried to identify this system using the proposed algorithm. The set of parameters used in this example is the following: (Ipop=50, Pcross=0.6, Pm=0.01). The steps for the LMS algorithms which are used in the proposed algorithm were m1lms=0.05, m2lms=0.005, respectively. Initially, we did not use additive noise. The average MSE that resulted after 100 runs was 0.0048. 60 times out of 100, the algorithm found the correct order of the system, while 40 times out of 100 the order found by the proposed algorithm was bigger than the real one. We repeated the previous experiment with presence of additive noise with SNR=10db. The average MSE that resulted after 100 runs was 0.58. This value is quite big because the proposed algorithm found the correct order only half times, while 30 times out of 100 found an order smaller than the real one. However, the performance of the proposed algorithm is much better than the performance of the algorithms OPS, FOS because the coefficients found are much closer to the real coefficients than the ones computed by the algorithms OPS, FOS.

Conclusions and Future Work

In this contribution a new hybrid intelligent algorithm foe system identification is presented. Experimental results showed that the proposed method is very effective in identifying unknown systems, even in cases with high additive noise. Furthermore, we have observed that in the most cases, the proposed method has found the correct order of the unknown systems without using a lot of a priori information. So, the proposed hybrid algorithm has created models that not only had small MSE but also were similar to the real systems. Except for that, all the models derived from the proposed method are stable.

References

1. Warwick, K., Kang, Y.-H., Mitchell, R. J.: Genetic least squares for system identification. Soft Computing 3. Springer Verlag. (1999) 200–205
2. Tan, K. C., Li, Y., Murray-Smith, D. J., Sharman, K. C.: System Identification and Linearization Using Genetic Algorithms with Simulated Annealing. Centre for Systems and Control, and Department of Electronics and Electrical Engineering University of Glasgow, Glasgow G12 8LT, U.K.
3. Garrido, S., Moreno, L.: Learning Adaptive Parameters with Restricted Genetic Optimization Method. J. Mira and A. Prieto (Eds.): IWANN 2001, LNCS 2084 (2001) 612–620
4. Malavazos, K.: Genetic algorithms and the proble of Arma systems' identification. Degree project for Master studies in the Department of Computer Mechanics and Informatics, University of Patras, Greece. (2002)
5. GAlib - A C++ Library of Genetic Algorithm Components, Matthew Wall, Massachusetts Institute of Technology (MIT).
6. Lu, S., Ju, K. H., Chon, K. H.: A new Algorithm for Linear and Nonlinear ARMA Parameter Estimation Using Affine Geometry. IEEE Trancactions on Biomedical Engineering. **48** (2001) 1116–1124

Numerical-Symbolical Methods Computing the Rank of Block Bidiagonal Toeplitz Matrices

Dimitrios Triantafyllou[1] and Marilena Mitrouli[1]

Department of Mathematics, University of Athens, Panepistemiopolis 15784, Athens, Greece
dtriant@math.uoa.gr, mmitroul@math.uoa.gr *

Abstract. The computation of the Rank of a matrix is an interesting problem with applications in many computational fields of science such as control theory, numerical linear algebra etc. In the present paper we study the computation of the Rank of a block bidiagonal sequence of matrices called PAPS sequence. We propose matrix-based, numerical and symbolical, updating and direct methods computing the Rank of block bidiagonal Toeplitz matrices and compare them with classical procedures. Methods such as QR factorization and SVD are stable but inefficient because of the big size of the initial matrix. Updating methods exploit the special structure of the PAPS matrix. We present new algorithms and modifications of the classical ones which deploy the special form of the PAPS sequence reducing significant the required flops and lead to fast and efficient algorithms. The numerical implementation of the algorithms leads to serious problems such as the computation of the numerical Rank in contrast with the symbolical implementation which guarantees the computation of the exact Rank of the matrix. The combination of numerical and symbolical operations suggests a new approach in software mathematical computations denoted as hybrid computations. For some of the above methods their hybrid nature is presented.

Introduction

The computation of the rank of a matrix [1, 3, 7] is a problem that has concerned many computational fields of science such as numerical analysis, numerical linear algebra, control theory etc. In this paper we present some methods computed the rank of a special structured matrix called PAPS. This matrix appears in the computation of the Weierstrass Canonical form of regular Matrix Pencils [5]. Taking advantage of its special block bidiagonal Toeplitz form we introduce reliable and efficient algorithms for the computation of its rank.

The paper is organized as follows. In section 2, we present an updating method for computing the rank of the PAPS matrix. Next we briefly describe the most significant direct methods for computing the the rank of a matrix and we suggest a new direct method which computes in an effective way the rank of the PAPS sequence. In section 3 are presented useful conclusions.

* This research was financially supported by the Special Account for Research Grand of Athens University.

The PAPS Sequense

Let A, B be two $n \times n$ matrices and let $\Gamma_i, ,\ i = 1, 2, \ldots, k, \ldots$, be the following sequence of matrices:

$$\Gamma_0 = A,\ \Gamma_1 = \begin{bmatrix} A & 0 \\ B & A \end{bmatrix},\ \Gamma_2 = \begin{bmatrix} A & 0 & 0 \\ B & A & 0 \\ 0 & B & A \end{bmatrix},\ \ldots,\ \Gamma_k = \begin{bmatrix} A & & \\ B & A & \\ & \ddots & \ddots \\ & & B \quad A \end{bmatrix}.$$

Our aim is to compute the rank of each term of the previous sequence.

Updating Method

Let $r_0 = rank\Gamma_0$, $r_1 = rank\Gamma_1 - rank\Gamma_0$, ..., $r_k = rank\Gamma_k - rank\Gamma_{k-1}$. If $rankA = k$ there will be k linear independent rows e_1, e_2, ..., e_k which produce the row space of A. Similarly, if $rankB = l$ there will be l linear independent rows $e_1^{'}$, $e_2^{'}$, ..., $e_l^{'}$, which produce the row space of B. Because $rankA = k$ we can zero with row operations the n-k linear dependent rows of A. Let A^0 be the resultant matrix. For the second term Γ_1 of the sequence, with row operations in matrix A arises the matrix:

$$\Gamma_1 = \begin{bmatrix} A & 0 \\ B & A \end{bmatrix} \longrightarrow \begin{bmatrix} A^{(0)} & 0 \\ B & A \end{bmatrix}$$

After row operations we zero the n-l linear dependent rows in B. There is also the possibility to be zeroed some other rows of $e_1^{'}$, $e_2^{'}$, ..., $e_l^{'}$ which produce the matrix B if they are coincided with some rows of e_1, e_2, ..., e_k which produce the matrix A. The row operations in B affect the entries of A which are right of B. Let $B^{(1)}$ and $A^{(1)}$ be the resultant matrices. After the row operations arises the following matrix:

$$\begin{bmatrix} A^{(0)} & 0 \\ B^{(1)} & A^{(1)} \end{bmatrix}$$

where $A^{(1)}$ is not necessary equal to $A^{(0)}$. So $rank\Gamma_1 = rankA^{(0)} + rank[B^{(1)}, A^{(1)}]$. Also $r_1 = rank\Gamma_1 - rank\Gamma_0 = rank[B^{(1)}, A^{(1)}]$.

We continue similarly in the third term:

$$\Gamma_2 = \begin{bmatrix} A & 0 & 0 \\ B & A & 0 \\ 0 & B & A \end{bmatrix} \ \textit{and so}\ rank\left(\begin{bmatrix} A & 0 & 0 \\ B & A & 0 \\ 0 & B & A \end{bmatrix}\right) = rank\left(\begin{bmatrix} A^{(0)} & 0 & 0 \\ B^{(1)} & A^{(1)} & 0 \\ 0 & B & A \end{bmatrix}\right)$$

and so we continue with the last matrix. With row operations in the last block-row $[0, B, A]$ and particularly in B we zero the n-l linear dependent rows of B. From the l remaining rows of B we could delete some rows if these rows are linear

combination of the rows of $A^{(1)}$ and these rows of $A^{(1)}$ have in the corresponding position of $B^{(1)}$ zero rows. After these operations we have the following matrix:

$$\begin{bmatrix} A^{(0)} & 0 & 0 \\ B^{(1)} & A^{(1)} & 0 \\ 0 & B^{(2)} & A^{(2)} \end{bmatrix} \Rightarrow rank\Gamma_2 = rank \begin{bmatrix} A^{(0)} & 0 & 0 \\ B^{(1)} & A^{(1)} & 0 \\ 0 & B^{(2)} & A^{(2)} \end{bmatrix}, r_2 = rank \left[B^{(2)} A^{(2)} \right].$$

Generalizing we have that $r_\theta = rank \left[B^{(\theta)} A^{(\theta)} \right]$.

Direct Methods

There are many classical direct methods for the computation of the rank of the PAPS sequence. These methods handle the whole matrix in contrast with the analyzed in the previous subsection updating method. The Gauss-Jordan (GJ) factorization, the QR factorization with column pivoting (QRCP) [4], the Rank Revealing QR (RRQR) [1], the Singular Value Decomposition (SVD) [2] and the Partial SVD (PSVD) [6] are the most known technics for the computation of the rank of a matrix. But as the number of blocks [B A] in PAPS sequence is increasing, the implementation of the previous classical methods becomes inefficient since the required flops of each method is of order of $O(n^3)$ for a $n \times n$ initial matrix. If the number of blocks [B A] is of order of n then all previous methods demand $O(n^4)$ flops and thus their complexity makes them inefficient for implementation.

We move the first block [A] to the end of the matrix:

$$\Gamma_{n+1} = \begin{bmatrix} A & & & \\ B & A & & \\ & B & A & \\ & & \ddots & \ddots \\ & & B & A \end{bmatrix} \rightarrow \widetilde{\Gamma}_{n+1} = \begin{bmatrix} B & A & & \\ & B & A & \\ & & \ddots & \ddots \\ & & B & A \\ A & & & \end{bmatrix}$$

We can take advantage of the special form of the initial modified matrix: As we can see in the previous form of $\widetilde{\Gamma}_{n+1}$ there are n same [B A] blocks (right shifted each time). We triangularize the first two blocks using the LU with partial pivoting or the QR factorization:

$$LU(\begin{bmatrix} B & A & 0 \\ 0 & B & A \end{bmatrix}) \rightarrow U \; or \; QR(\begin{bmatrix} B & A & 0 \\ 0 & B & A \end{bmatrix}) \rightarrow R$$

where U, R are upper triangular matrices. We update the other entries (the $[\frac{n-1}{2}]$ pairs of [B A]) without making any other calculations. We repeat the previous procedure until the number of the same blocks be less or equal to 3. Then we apply one more time the LU or QR factorization to zero specific entries of the whole (which is almost triangular) matrix. The number of the repeated steps is less or equal to $log_2(n)$. Each step requires $O(\frac{n^3}{3})$ or $O(\frac{2n^3}{3})$ flops for every LU or QR factorization and thus the total complexity remains of order $O(n^3)$ which

makes the method efficient.

The modified LU with partial pivoting (MPLU) or modified QR (MQR) factorization

While number of same blocks of $\widetilde{\Gamma}_{n+1} > 3$ do
  if number of same blocks of $\widetilde{\Gamma}_{n+1}$ = odd
    move the last block to the bottom of the matrix
  endif
  Compute the upper triangular matrix R or U
  R=QR(Γ) or U=LU(Γ), where Γ contains
  the two first same blocks of $\widetilde{\Gamma}_{n+1}$
Compute the upper triangular matrix R or U :
R =QR($\widetilde{\Gamma}_{n+1}^{(n)}$) or U =LU($\widetilde{\Gamma}_{n+1}^{(n)}$)

This algorithm can be implemented in the first phase of SVD or PSVD.

Conclusions

The most efficient way to compute the rank of the PAPS sequence is the PSVD using the MQR factorization in its first phase. The algorithm can be implemented numerically since the Housholder and the Givens transformations are orthogonal and thus stable. Alternatively we can use the MPLU in the first phase of PSVD reducing significantly the required flops. Because the gaussian elimination with partial pivoting is theoretically not stable we can implement the MPLU symbolically and next the Givens rotations numerically combining in a hybrid way the two arithmetics. The classical methods are inefficient because of the flops that they demand. The updating method requires less flops than the classical methods because it handles only a part of the PAPS sequence and not the entire matrix.

References

1. Chan, T.: Rank revealing QR fact. Lin.Alg. and its App. **88/89** (1987) 67–82
2. Datta B.N.: Numerical Linear Algebra and Applications, Second Edition, Brooks/Cole Publishing Company, United States of America (1995)
3. Foster, L.: Rank and null space calculations using matrix decomposition without column interchanges. Lin. Alg. and its Appl. **74** (1984) 47–71
4. Golub, G.H., Van Loan, C.F.: Matrix Computations, Third Edition, The John Hopkins Univercity Press, Baltimore, London (1989)
5. Kalogeropoulos, G., Mitrouli, M.: On the computation of the Weierstrass Canonical form of a Regular Matrix Pencil. Control and Computers. **22** (1994)
6. Van Huffel: Partial singular value decomposition algorithm. J. of Comp. and Applied Math. **33** (1990) 105–112
7. Yalamov, P.Y., Mitrouli, M.: A fast Algorithm for Index Annihilation Computations. J. of Comp. and Appl. Math. **108** (1999) 99–111

Applying robust multibit watermarks to digital images

Dimitrios Tsolis[1], Spiridon Nikolopoulos[2], Lambros Drossos[3], Spiridon Sioutas[4], and Theodore Papatheodorou[1]

1. Informatics and Telematics Institute, Centre of For Research and Technology, Greece
2. Department of Computer Engineering and Informatics, University of Patras, Greece
3. Department of Applied Informatics in Administration and Economics, Technological Institute of Messolongi
4. Department of Informatics, Ionian University

Abstract. The current work is focusing on the implementation of a robust multibit watermarking algorithm for digital images, which is based on an innovative spread spectrum technique analysis. The paper presents the watermark embedding and detection algorithms, which use both wavelets and the Discrete Cosine Transform and analyzes the arising issues.

Key words: Multibit watermarking, spread spectrum analysis, wavelet domain, subband-DCT, copyright protection, digital images.

Introduction

Watermarking is probably the most promising technological approach against Intellectual Property Rights protection [2]. The majority of watermarking systems achieving high robustness are only capable of embedding one bit of information placing specific limitations on the potentials of the encrypted information. Most of the real word applications raise the requirement of a multibit robust watermarking scheme where the detectors output can be interpreted into meaningful and valuable information.

Multibit Watermark Technique

Spread Spectrum Watermarking in the Wavelet Domain

Generally, a watermark is a narrow band signal, which is embedded to the wide band signal of a digital image [3]. Spread spectrum techniques allow the encoded information to be spread across a wide range of frequencies. Thus, if the signal is distorted by some process that damages only a fraction of the frequencies, such as a band-pass filter or addition of band limited noise, the encrypted information will still be identifiable. Furthermore, high frequencies are appropriate for rendering the watermarked message invisible but are inefficient in terms of robustness, whereas low frequencies are appropriate with regards to robustness but are useless because of the unacceptable visual impact.

General Description of the Additive Algorithm

In additive watermarking algorithms, the signature data is a sequence of numbers w_i of length N that is embedded in a suitably selected subset of the host signal data coefficients, f. The basic and commonly used embedding formula is

$$f'(m,n) = f(m,n)(1 + awi) \; . \tag{1}$$

where a is a weighting factor and F' is the resulting modified host data coefficients carrying the watermark information. Alternative embedding formulas have been proposed by Cox [39,Master thesis], such as

$$f'(m,n) = f(m,n) + awi \; . \tag{2}$$

or using the logarithm of the original coefficients,

$$f'(m,n) = f(m,n)e^{awi} \; . \tag{3}$$

An important property of the above formula is that an inverse embedding function,

$$w_i' = \frac{f''(m,n) - f(m,n)}{a \times f(m,n)} \; . \tag{4}$$

can be easily derived to compute w' from f" given the original host coefficients as reference. By f' we denote the received, possibly altered, image that might contain the watermark w. At the next step, the extracted watermark sequence w' is compared to the original embedded watermark w using the normalized correlation of the sequences as a similarity measure

$$\delta = \frac{w' \times w}{\|w'\| \times \|w\|} \; . \tag{5}$$

The similarity varies in the interval [-1,1], a value well above 0 close to 1 indicates the extracted sequence w' matching the embedded sequence w and therefore concluding that the image has been watermarked with w. A detection threshold can be established to make the detection decision, ¿. The detection threshold can be derived either experimentally by observing the correlation of random sequences or analytically. For example, a threshold

$$\tau = \frac{\alpha}{S \times N} \sum^{N} |f'| \; . \tag{6}$$

can be used, where S, the standard deviation, is 2 or 3.

Of course, the choice of the threshold influences the false-positive and false-negative probability. Hence, a lot of effort has been focused on devising reliable methods to compute predictable correlation thresholds and efficient watermark detection systems.

The weighting factor a does not necessarily have to be constant over the entire watermark sequence, but can be chosen adaptively to capture and exploit local properties of the host signal.

Before the watermark embedding, the host image F is usually subjected to a two dimensional transform T such as the DCT, DFT or DWT to derive a frequency representation f of the data, f=TxF. Following the watermarking modifications in the frequency domain, the spatial image representation is regained by applying the inverse transform

$$T^{-1}, F = T^{-1} \times f \ . \tag{7}$$

Subband-DCT

In our implementation, we adopted a method in which both wavelets and the well known Discrete Cosine Transform (DCT) are involved [1]. Highpass and lowpass filters are used to subsample and filter the original image. The combination of the two filters for each direction (horizontal and vertical) of filtering produces four subbands for each level of decomposition. The band that corresponds to lowpass filtering in both directions (LL band) can be further subsampled and filtered thus providing another level of decomposition. Finally, each of the bands is transformed applying the DCT transform. In the proposed scheme a one level decomposition with four bands was selected, utilizing the most trivial wavelets, originally introduced by Haar. The next stage is transforming the produced bands using the DCT. The watermark casting is performed according to the following additive rule:

$$t'_i = t_i + \alpha t_i x_i \ . \tag{8}$$

where t_i are the transformed coefficients, are the watermarked coefficients and x_i is a random sequence of Gaussian distribution, used as a watermark. The a-parameter specifies the casting strength.

Spread spectrum multibit watermarking technique

The embedding of a robust multibit watermark is accomplished through casting several zero-bit watermarks onto specified coefficients. The image watermark, a random sequence of Gaussian distribution in our case, is casted multiple times onto the selected coefficients preserving the same sequence length but shifting the start point of casting by one place. Actually the final watermark that will be embedded into the image is not a single sequence but many different sequences generated with different seeds. These sequences will be casted, one after the other, on the mid coefficients of the image, using the additive rule mentioned above and begging from successive starting points. If all sequences where to be casted, beginning from the same starting point, then, besides the severe robustness reduction resulting from the weak correlation, the possibility of false positive detector response would dramatically increase, since every number that has participated as a seed during the sequence generation procedure, will be estimated by the detector as a valid watermark key. Shifting the starting point by one degree for every sequence casting ensures that the false positive rate will remain in very small level due to the artificial desynchronisation introduced. Every single

random sequence of Gaussian distribution is generated using a different number as the seed for the Gaussian sequence generator. It is important to differentiate the sequences in order not to mislead the detection mechanism, since it is based on the correlation between the extracted sequence and the sequence produced with the watermark key.

The watermark key is responsible both for the generation of the first sequence and the construction of a vector, containing the rest of the numbers that will serve as the corresponding seeds. The placement of several Gaussian sequences into the image content can model, under specific conventions, a multibit watermark. The detection of a zero-bit watermark is interpreted as if the bit value of the specified bit is set to one. On the contrary, failure of the detector to detect the zero-bit watermark leads to the conclusion of a zero bit value. Thus, in order for a message to be casted into the image content, it is initially encoded using the binary system and applied afterwards in the sense of zero-bit watermarks using the embedding mechanism and according to the derived bit sequence.

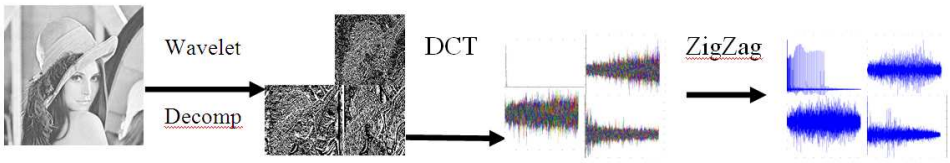


Fig. 1. Zig-Zag coefficients of the LL Band

Concluding Remarks

Most of the effort addressed in this work was dedicated on formulating a simple and easy to implement technique for robustly embedding multibit watermarks into digital images. The result was a technique applicable to every spread spectrum frequency domain watermarking method capable of hiding 214 different messages while maintaining a sufficient level of robustness.

References

1. Fotopoulos V., Skodras A. N.: A Subband DCT Approach to Image Watermarking (5-8 September 2000) Tampere, Finland: X European Signal Processing Conference (EUSIPCO-2000)
2. Katzenbeisser F., Petitcolas F.A.P: Information Hiding - Techniques of Steganography and Digital Watermarking. Artech House, Computer Series (2000) 95-172
3. Wayner P.: Disappearing Cryptography - Information Hiding: Steganography and Watermarking. Morgan Kaufmann (2002) 291-318.

Some Results on Sign Symmetric Matrices

Michael G. Tzoumas

mtzoumas@sch.gr

Abstract. In the last four decades many researchers have studied and analyzed the study of sign symmetry and positivity of principal minors of matrices, since these issues are related to stability. In this work we extend the theory about sign symmetric basic p–circulant permutation and sifted p–circulant matrices. We present and prove sufficient and necessary conditions for P–matrices and necessary conditions for P^2–matrices. Finally we present a class of matrices, where the P^2–matrices are stable

A Numerical Technique for Computing Real Eigenvalues of Real Tridiagonal Matrices

F. N. Valvi[1] and V. S. Geroyannis[2]

[1] Department of Mathematics, University of Patras,
GR-26500 PATRAS, GREECE, fvalvi@upatras.gr
[2] Department of Physics, University of Patras,
GR-26500 PATRAS, GREECE, vgeroyan@upatras.gr

Abstract. In this paper, we present a numerical technique for computing the real eigenvalues of real tridiagonal matrices, which have, generally, both real and complex conjugate eigenvalues. This technique consists in the highly accurate computation of the integral of the ratio $p'(\lambda)/p(\lambda)$, $p(\lambda)$ being the characteristic polynomial of the tridiagonal matrix under consideration, along a closed contour $\mathcal{C}$ defining a complex region $\mathfrak{R} \subset \mathbb{C}$, within which eigenvalues of interest are located. The value of this integral counts the eigenvalues enclosed in $\mathfrak{R}$. Choosing $\mathcal{C}$ to be a rectangle with basis Δx and height running from $-i\Delta y$ to $+i\Delta y$, we can both count and localize the real eigenvalues lying on Δx by decreasing Δy and Δx, respectively. That is, a sufficiently small Δy excludes the complex conjugate eigenvalues and a bisection–type method applied on Δx gives the real eigenvalue(s) of interest.

Introduction

A real tridiagonal matrix

$$A = \begin{bmatrix} a_1 & b_2 & 0 & \dots & 0 & 0 \\ c_2 & a_2 & b_3 & \dots & 0 & 0 \\ 0 & c_3 & a_3 & \dots & 0 & 0 \\ \dots & & & & & \\ 0 & 0 & 0 & \dots & a_{n-1} & b_n \\ 0 & 0 & 0 & \dots & c_n & a_n \end{bmatrix}, \tag{1}$$

where $a_i, i = 1, 2, \dots, n$ and $b_i, c_i, i = 2, 3, \dots, n \in \mathbb{R}$, and $b_i c_i > 0$ has all its eigenvalues real. When $b_i c_i \leqslant 0$ for even one value of i, its eigenvalues are generally complex. Computing the real eigenvalues of the latter case is a very interesting problem since it has many engineering and scientific applications.

To this purpose, we try to localize the zeros of the characteristic polynomial $p(\lambda) \equiv p_n(\lambda)$ of A, which is given by the recursive formula

$$\begin{cases} p_k(\lambda) = (\lambda - a_k)p_{k-1}(\lambda) - b_k c_k p_{k-2}(\lambda), & k = 2, 3, \dots, n, \\ p_1(\lambda) = \lambda - a_1, \\ p_0(\lambda) = 1. \end{cases} \tag{2}$$

In particular, we consider a complex region $\mathfrak{R} \subset \mathbb{C}$ defined as the interior of the closed contour $\mathcal{C}$, within which eigenvalues of interest for the characteristic polynomial $p(\lambda)$ of the tridiagonal matrix A shall be localized and estimated. Then the Residue Theorem implies that the total number, N, of the eigenvalues of $p(\lambda)$ which lie in $\mathfrak{R}$ is given by

$$N = \frac{1}{2\pi i} \int_{\mathcal{C}} \frac{p'(\lambda)}{p(\lambda)} \, d\lambda. \tag{3}$$

Thus, we can calculate N via numerical integration of the contour integral (3). The details of such a treatment are given in the next section.

Numerical Integration of the Closed Contour Integral

The numerical integration (numerical quadrature) of an arbitrary function $g(z)$ is basically no different than the numerical integration of the corresponding ordinary differential equation (ODE) $df/dz = g(z)$. In particular, the numerical evaluation of the contour integral (3) is equivalent to the numerical solution of an initial value problem (IVP) in an ODE; that is, for our problem,

$$\begin{cases} \dfrac{df}{d\lambda} = \dfrac{p'(\lambda)}{p(\lambda)}, \\ \\ f_s \equiv f(\lambda_s) = f_s^{\mathrm{re}} + i f_s^{\mathrm{im}}, \end{cases} \tag{4}$$

where λ_s is the starting integration point and f_s is the initial value for the unknown function $f(\lambda)$. Since $\mathcal{C}$ is a closed contour, the final integration point, λ_f, coincides with the starting integration point, $\lambda_f = \lambda_s$. If we define the closed countour $\mathcal{C}$ as the rectangle

$$\mathcal{C} = \left\{ \begin{array}{ll} x_s - i\Delta y & \longrightarrow x_s + i\Delta y \\ x_s + i\Delta y & \longrightarrow (x_s - \Delta x) + i\Delta y \\ (x_s - \Delta x) + i\Delta y & \longrightarrow (x_s - \Delta x) - i\Delta y \\ (x_s - \Delta x) - i\Delta y & \longrightarrow x_s - i\Delta y \end{array} \right\}, \tag{5}$$

where the quantities x_s, Δx, Δy are given as initial data, then the starting integration point is $\lambda_s = x_s - i\Delta y$ and, without loss of generality, the initial value f_s is such that $f_s^{\mathrm{re}} = 0, f_s^{\mathrm{im}} = 0$. Then, we need a numerical tool to solve this IVP, to calculate the value $f(\lambda_f)$, and thus to find N,

$$N = \frac{f(\lambda_f)}{2\pi i}. \tag{6}$$

Efficiency of ATOMFT for the Particular IPV

After having converted our problem to the IVP (4), we apply the ATOMFT System (Ref. [1]) and we verify its efficiency and accuracy with respect to the problem under consideration.

ATOMFT is a powerful, flexible, Fortran 77-based software system for fast and accurate solution of systems of ODEs. The user prepares a Fortran-like statement of the system of ODEs to be solved, the ATOMFT translator reads this statement and writes a Fortran 77 program, which then runs and solves the system of ODEs by using long Taylor series (Refs. [2–4]). ATOMFT is distributed at no cost (download from www.eng.mu.edu/corlissg/ATOMFT3_11/Atom3_11). The user has certain obligations described in Ref. [5].

The ATOMFT system is simple enough to be used by students, practical enough to be used by engineers, and versatile enough to be used by research mathematicians (Ref. [1], Sect. 1.4). The very high order and precise error control used by ATOMFT enable it to solve problems, for which other methods have difficulties. ATOMFT supports solution of ODEs defined in the complex plane (details for solving such problems are given in Ref. [1], Sects. 3.2.4, 3.2.5, 3.7; details concerning numerical quadrature problems and their conversion to numerical integration of ODEs are given in Sect. 6.3).

To the purpose of testing ATOMFT in the IVP (4), we have defined the tridiagonal matrix

$$A = \begin{bmatrix} 4 & -3 & 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 5 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 3 & 2 & 10^{-6} & 0 & 0 & 0 & 0 \\ 0 & 0 & -10^{-6} & 2 & 8 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 7 & -2 & 0 & 0 \\ 0 & 0 & 0 & 0 & 3 & 9 & -1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 6 & 3 \\ 0 & 0 & 0 & 0 & 0 & 0 & -1 & 2 \end{bmatrix}, \tag{7}$$

with eigenvalues

$$\begin{cases} \lambda_1 = 8 + i\,2.23607, \\ \lambda_2 = 8 - i\,2.23607, \\ \lambda_3 = 5, \\ \lambda_4 = 4.5 + i\,1.65831, \\ \lambda_5 = 4.5 - i\,1.65831, \\ \lambda_6 = 3, \\ \lambda_7 = 2 + i\,10^{-6}, \\ \lambda_8 = 2 - i\,10^{-6}. \end{cases} \tag{8}$$

We have run ATOMFT defining the closed contour $\mathcal{C}$ by choosing several values for its definition data $x_s, \Delta x,$ and Δy. We have verified that, in any case, ATOMFT counts correctly the eigenvalues lying in the interior of $\mathcal{C}$. Indicatively,

setting $x_s = 10$ and $\Delta x = 10$, we find that

$$\begin{cases} N = 8 \ \text{ for } \ \Delta y = 3, \\ N = 6 \ \text{ for } \ \Delta y = 2, \\ N = 4 \ \text{ for } \ \Delta y = 1, \\ N = 4 \ \text{ for } \ \Delta y = 10^{-5}, \\ N = 2 \ \text{ for } \ \Delta y \leqslant 10^{-7}. \end{cases} \tag{9}$$

Thus, we are able to count correctly the real eigenvalues of A, i.e, the eigenvalues λ_3 and λ_6 of Eq. (8), and then to calculate them (e.g., by a bisection–type method).

Acknowledgments

The authors acknowledge the use of the ATOMFT System.

References

1. Chang, Y. F.: ATOMFT User Manual, Version 3.11. www.eng.mu.edu/corlissg/Pubs/ATOMFT3_11/ Atom3_11/Manual (1994)
2. Chang, Y. F.: Automatic solution of differential equations. In Constructive and Computational Methods for Differential Equations, edited by D. L. Colton and R. P. Gilbert. Springer Lecture Notes in Math. **430** (1974) 61–94
3. Chang Y. F., and Corliss, G.: Ratio-like and recurrence relation tests for convergence of series. J. Inst. Math. Appl. **25** (1980) 349–359
4. Chang, Y. F., and Corliss, G.: Solving ordinary differential equations using Taylor series. ACM Trans. Math. Soft. **8** (1982) 114–144
5. Corliss, G.: READ.ME for ATOMFT v 3.11 (ATOMFT Compiler, version 3.11, Copyright (C) 1979-94, Y. F. Chang. Version 3.11 completed (6/21/93)), revised 20-JAN-1994 by George Corliss. www.eng.mu.edu/corlissg/Pubs/ATOMFT3_11/Atom3_11/read.me

Abstracts of Posters

Local Application of One-Level Trees

D. Anyfantis, M. Karagiannopoulos, S.B. Kotsiantis, and P.E. Pintelas

Educational Software Development Laboratory
Department of Mathematics, University of Patras, Hellas
{dany, mariosk, sotos, pintelas}@math.upatras.gr

Abstract. We propose a technique of local application of one-level decision and regression trees. We recognize local regions having similar characteristics and then build local expert on each of these regions describing the relationship between the data characteristics and the target value. We performed a comparison with other well known lazy methods on standard benchmark datasets and the proposed technique produced the most accurate results.

Introduction

Instance-based (lazy) learners classify an instance by comparing it to a database of pre-classified examples. Local learning [1] can be understood as a general principle that allows extending learning techniques designed for simple models, to the case of complex data for which the model's assumptions would not necessarily hold globally, but can be thought as valid locally. In this paper, we propose a technique of local application of one-level decision and regression trees (decision stumps) [7]. We performed a comparison with other well known lazy methods on standard benchmark datasets and the proposed technique produced the most accurate results. In the next section, we describe the proposed method and we evaluate the proposed method on several UCI datasets by comparing it with other lazy methods. Finally, section 4 concludes the paper and suggests further directions.

Proposed Algorithm and Experiments

Local methods have significant advantages when the probability measure defined on the space of symbolic objects is very complex, but can still be described by a collection of less complex local approximations. Some theoretical results and experimental results [4], [9] indicate that a local learning algorithm provides a feasible solution to this problem. The proposed algorithm builds a model for each point to be estimated, taking into account only a subset of the training points. This subset is chosen on the basis of the preferable distance metric between the testing point and the training point in the input space. For each testing point, a decision stump learner is thus learned using only the training points lying close to the current testing point. Generally, the proposed method consists of the four steps (see Fig. 1).

1. Determine a suitable distance metric.
2. Find the k nearest neighbors using the selected distance metric.
3. Apply the DS algorithm using as training instances the k instances.
4. The 5answer of the model is the prediction for the testing instance.

Fig. 1. Local Decision Stump

In our experiments, we used the most well known -Euclidean similarity function- as distance metric. The proposed algorithm also requires choosing the value of K. In the current implementation we decided to use a fixed value for K (=50): a) in order to keep the training time low and b) about this size of instances is appropriate for a simple algorithm, to build a precise model according to [6], [8]. We have experimented with a number of classification datasets from the UCI repository [2]. In order to calculate the classifiers' accuracy, cross validation was run 10 times for each algorithm and the average value was calculated. It must be mentioned that we used the free available source code for most of the algorithms by [10] for our experiments. We compare the proposed methodology with K-nearest neighbors using k=3 (most common used number of neighbors), as well as k=50 because the proposed algorithm uses 50 neighbors. In addition, we tested Kstar: another instance-based learner which uses entropy as distance measure [5]. In following Tables, we represent as "v" that the specific algorithm performed statistically better than the proposed method according to t-test with $p < 0.05$. On the other hand, "*" indicates that the proposed method performed statistically better than the specific algorithm according to t-test with $p < 0.05$. In all the other cases, there is no significant statistical difference between the results (Draws). As one can see, the performance of the presented method is more accurate than the other techniques. Subsequently, we experimented with a number of datasets from the UCI repository [2]. We compared the proposed methodology with Simple DS algorithm, K-nearest neighbors using k=50 because the proposed algorithm uses 50 neighbors. In addition, we tested Kstar: another instance-based learner which uses entropy as distance measure [5]. Similarly, in order to calculate the models' correlation coefficient for our experiments, cross validation was run 10 times for each algorithm and the average value was calculated. The performance of the proposed method is better than the other tested techniques.

Conclusion

Our experiment in real datasets shows that the proposed method outperforms other lazy classification and regression methods. In a following work we will focus on the problem of reducing the size of the stored set of instances [3] while trying to maintain or even improve generalization accuracy by avoiding noise and overfitting.

Table 1. Comparing local decision stumps with other learners

Datasets	Local DS	Kstar	3NN	DS	50NN
audiology	72.68	80.32 v	67.97 *	46.46 *	35.95 *
autos	74.82	72.01*	67.23 *	44.9 *	48.18 *
colic	80.87	75.71*	80.95	81.52	84.04 v
credit-rating	83.61	79.1 *	84.96	85.51	86.16 v
Glass	70.58	75.31 v	70.02	44.89 *	56.16 *
heart-c	78.29	75.18 *	81.82 v	72.93 *	81.58 v
ionosphere	88.24	84.64 *	86.02 *	82.57 *	71.65 *
Iris	94	94.67	95.2	66.67 *	90.53 *
monk3	93.44	86.22 *	86.72 *	76.01 *	82.46 *
Vehicle	69.58	70.22 *	70.21 *	39.81 *	63.47 *
Vote	95.4	93.22 *	93.08 *	95.63 *	90.41 *

Table 2. Comparing the Algorithms

Dataset	Local DS	Kstar		50NN		DS	
servo	0.89	0.86	*	0.65	*	0.79	*
autoHorse	0.92	0.90		0.85	*	0.72	*
autoMpg	0.89	0.91		0.86	*	0.74	*
bodyfat	0.94	0.87	*	0.91	*	0.82	*
cholesterol	0.12	0.04	*	0.17	v	0.04	*
fishcatch	0.94	0.99	v	0.78	*	0.83	*
housing	0.84	0.90	v	0.77	*	0.60	*
lowbwt	0.78	0.62	*	0.75	*	0.78	
pbc	0.43	0.30	*	0.52	v	0.43	
pwLinear	0.84	0.72	*	0.85		0.68	*
quake	0.09	0.08		0.06	*	0.09	
sensory	0.47	0.39	*	0.36	*	0.29	*
auto93	0.72	0.77	v	0.71		0.59	*

References

1. C.G. Atkeson, A.W. Moore & S. Schaal, Locally weighted learning. Artificial Intelligence Review, 11(1-5), 11-73, 1997.
2. C. Blake & C. Merz, UCI Repository of machine learning databases. Irvine, CA: University of California, Department of Information and Computer Science. http://www.ics.uci.edu/ mlearn/MLRepository.html
3. H. Brighton, C. Mellish, Advances in Instance Selection for Instance-Based Learning Algorithms, Data Mining and KDD, 6, 153-172, 2002.
4. L. Bottou and V. Vapnik, Local learning algorithm, Neural Computation, vol. 4, no. 6, pp. 888-901, 1992.
5. C. John and L. Trigg, K*: An Instance- based Learner Using an Entropic Distance Measure", Proc. of the 12th Inter. Conf. on ML, pp. 108-114, 1995.

6. E. Frank, M. Hall and B. Pfahringer, Locally weighted naive Bayes. Proc. of the 19th Conf. on Uncertainty in AI. Acapulco, Mexico. Morgan Kaufmann, 2003.
7. W. Iba, & P. Langley, Induction of one-level decision trees. Proc. of the Ninth Intern. ML Conf (1992). Aberdeen, Scotland: Morgan Kaufmann.
8. Loader, C., Local Regression and Likelihood. Springer, New York, (1999).
9. V.N. Vapnik, Statistical Learning Theory, Wiley, New York, 1998.
10. Ian H. Witten and Eibe Frank, Data Mining: Practical machine learning tools and techniques, 2nd Edition, Morgan Kaufmann, San Francisco, 2005.

A Note on the Difference Schemes of the Second Order of Accuracy for Hyperbolic Equations

Allaberen Ashyralyev[1] and Mehmet Emir Koksal[23]

[1] Department of Mathematics, Fatih University, 34500, Buyukcekmece, Istanbul, Turkey aashyr@fatih.edu.tr http://math.fatih.edu.tr
[2] Graduate Institute of Sciences and Engineering, Fatih University, 34500, Buyukcekmece, Istanbul, Turkey
[3] Department of Mathematics, Gebze Institute of Technology, 41400, Gebze, Kocaeli, Turkey mekoksal@fatih.edu.tr

Abstract. We are interested in studying the second order of accuracy two-step absolute stable difference schemes for the approximate solutions of the initial value problem

$$\frac{d^2u(t)}{dt^2} + A(t)u(t) = f(t) \quad (0 \le t \le T), u(0) = \varphi, u'(0) = \psi$$

in a Hilbert space H with the self-adjoint positive definite operators $A(t)$. In the present paper two new difference schemes of a second order of accuracy generated by integer power of $A(t)$ for approximately solving this initial-value problem are presented. The stability estimates for the solutions of these difference schemes are established. The theoretical statements for the solution of these difference schemes are supported by the results of the gives numerical examples.

References

1. Ashyralyev, A. and Sobolevskii, P.E., *New Difference schemes for Partial Differential Equations*, Birkhauser Verlag, 2004
2. Ashyralyev, A. and Sobolevskii, P.E., "A note on the difference schemes for hyperbolic equations", *Abstract and Applied Analysis,* 2:63-70, 2001
3. Ashyralyev, A. and Sobolevskii, P.E., "Two new approaches for construction of the high order of accuracy difference schemes for hyperbolic differential equations", *Discrete Dyn. Nat. Soc.,* 2:183-213, 2005
4. Ashyralyev, A. and Koksal, M.E., "On the second order of accuracy difference scheme for hyperbolic equations in a Hilbert space", *Numerical Functional Analysis and Optimization,* 26(7-8):739-772, 2005
5. Ashyralyev, A. and Koksal, M.E., "Stability of a second order of accuracy difference scheme for hyperbolic equations in a Hilbert space", *Discrete Dynamics in Nature and Society,* 2007 (Submitted)
6. Sobolevskii, P.E. and Chebotaryeva, L.M., "Approximate solution by the method of lines of the Cauchy problem for abstract hyperbolic equations", *Izv. Vyssh. Uchebn. Zav. Matematika,* 5:103-116, 1977

Automated Grading of Hip Osteoarthritis Severity Based on Computational Regional Descriptors of Radiographic Hip Joint Space

Ioannis Boniatis[1], Lena Costaridou[1], Dionisis Cavouras[2], Elias Panagiotopoulos[3], and George Panayiotakis[1]

[1] University of Patras, Department of Medical Physics, 265 00 Patras, Greece
iboniat@yahoo.com, {costarid, panayiot}@upatras.gr
[2] Technological Institute of Athens, 122 10 Athens, Greece
cavouras@teiath.gr
[3] University of Patras, Department of Orthopaedics, 265 00 Patras, Greece
ecpanagi@med.upatras.gr

Abstract. A machine learning system was developed for the grading of hip osteoarthritis (OA) severity. Employing custom developed software, 64 Hip Joint Space (HJS) images (18 normal, 46 osteoarthritic) were obtained from the digitized pelvic radiographs of 32 patients of unilateral and / or bilateral hip OA. Computational descriptors, evaluating texture and shape properties of the region of radiographic HJS were generated from the corresponding images and were used in the design of a two-level hierarchical decision tree structure. The latter, discriminated successfully between Normal and osteoarthritic hips at Level1 (96.9% overall accuracy) as well as hips of "Mild-Moderate" OA and of "Severe" OA at Level 2 (89.1% overall accuracy). The suggested approach may contribute to OA-patient management.

Introduction

Osteoarthritis (OA) is a major cause of morbidity worldwide, representing the most common form of joint disorder [1]. Plain film radiography is considered as the imaging modality of reference for the assessment of the osteoarthritic joint [2]. The characteristic radiographic findings of hip OA comprise the narrowing of Hip Joint Space (HJS), the sclerosis of subchondral bone, the formation of osteophytes, the development of subchondral cysts as well as abnormalities of the bone margins [1]. Radiographic assessment of hip OA severity is relied to a great extend on qualitative scales. The latter comprise severity grades, which are subjectively assigned to the studied joint, while the definition of the grades is based on the characteristic radiographic findings of the disease [3]. Among these scales, the one proposed by Kellgren and Lawrence (KL) has been accepted as the reference standard [4].

Previous studies have introduced thresholds of manually measured HJS-width, for characterizing a hip as normal or osteoarthritic [5]. In previous studies

performed by our group, the radiographic texture as well as the shape of HJS have been, separately, utilized for the discrimination among OA-severity categories [6], [7]. In the present study, osteoarthritic alterations of the hip are evaluated by means of the combined use of texture and shape descriptors, not previously employed in the hip OA investigation.

Materials and Methods

Clinical Sample and Radiographic Images

The clinical sample of the study comprised 64 hips (18 normal, 46 osteoarthritic), corresponding to 32 patients of unilateral (18) or bilateral (14) hip OA. For each patient, a pelvic radiograph was available. All radiographs were obtained according to a specific radiographic protocol and were digitized. The radiographic severity of hip OA was graded by each of three orthopaedists, employing the KL scale [4]. Accordingly, three major OA-severity categories were formed, in which the hips were allocated into: "Normal (18 hips)", "Mild /Moderate (16 hips)", and "Severe (30 hips)".

On each radiograph, two HJS-ROIs, corresponding to patient's both hips, were determined, employing custom developed algorithms in Matlab software (The MathWorks Inc., Natick, USA). The whole procedure concerned the following steps: (i) contrast enhancement and emphasis of the articular margins of the hip joint by implementing the Contrast-Limited Adaptive Histogram Equalization (CLAHE) method [8], (ii) formation of an acute angle of 450, providing the medial and lateral limits of the HJS-ROI [9] (see Fig. 1a), and (iii) manual delineation of the inferior and superior articular margins of the joint. This HJS-ROI (Fig. 1b) was subjected to further texture and shape analysis.

Generation of Computational Regional Descriptors

From each HJS-ROI two sets of computational descriptors, evaluating texture and shape properties of the specific anatomical region, were generated employing custom developed algorithms. Texture analysis of HJS-ROIs concerned the following: (i) calculation of the Fourier spectrum of the HJS-ROI image (see Fig. 1c), (ii) representation of the spectrum by means of the function of polar coordinates $S(r, \theta)$and consideration of the latter, for each direction θ, as one dimensional function of the form $S_\theta(r)$, (iii) generation of the one-dimensional spectral-energy signature, according to:

$$S(r) = \sum_{\theta=0}^{\pi} S_\theta(r) \tag{1}$$

and normalization of it to the interval $[0, 1]$ (see Fig. 1d), (v) calculation, as textural features, of the following descriptors of the one-dimensional signal $S(r)$: (a) mean value, (b) variance, (c) skewness, (d) kurtosis of the signature values as well as (e) the absolute difference between the maximum and the mean value

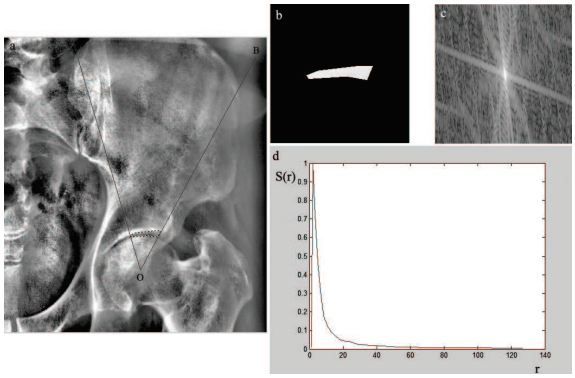


Fig. 1. (a) Determination of Hip Joint Space (HJS) Region of Interest (ROI) by utilizing patient's anatomical landmarks. O: centre of femoral head, V: highest point of homolateral sacral wing. OB: line drawn at 450 to OV. Dotted line represents the delineated articular margins. (b) Segmented HJS-ROI. (c) Spectrum of segmented HJS-ROI. (d) Plot of S(r).

of $S(r)$. On the other hand, Hu's invariant moments were generated and were employed as shape features of radiographic HJS-ROIs [10].

Design of the Computer-Based Grading System

The automatic grading of hip OA severity was performed by a hierarchical decision tree structure, which comprised two levels. The first level of the classification system was implemented by the Bayes classifier [11], used for the characterization of a hip as normal or osteoarthritic. At the second level, the osteoarthritic hips were further discriminated as of "Mild / Moderate" OA or of "Severe" OA by the Probabilistic Neural Network (PNN) classifier [12]. At both levels, the classifiers were designed employing: (i) the computational texture and shape descriptors that were generated from the region of the segmented radiographic HJS-ROI and (ii) the exhaustive search procedure in conjunction with the Leave One Out classification performance evaluation method [11].

Results and Discussion

Statistical analysis revealed the existence of statistically significant differences for the generated computational regional descriptors ($p<0.001$). The overall classification accuracy obtained for the discrimination between normal and osteoarthritic hips was 96.9%, since the Bayes classifier characterized properly 62 out of 64 hips. The specificity was 88.9% (correct classification for 16 out of 18 Normal hips), while all the osteoarthritic hips (46 out of 46) were assigned to the proper category (100% sensitivity). At the second level of the hierarchical tree, the overall accuracy classification accomplished by the PNN was 89.1% (proper

characterization for 41 out of 46 osteoarthritic hips). Only one of 16 hips of "Mild / Moderate" OA was misclassified as of Severe OA (93.8% classification accuracy), while 26 out of 30 hips of "Severe" OA were properly characterized (86.7% classification accuracy).

Conclusion

In conclusion, computational regional descriptors of radiographic Hip Joint Space were found able to quantify osteoarthritic alterations of the hip joint. Taking into consideration the relatively high classification scores obtained by the proposed system, the latter may contribute in OA-patient management.

Acknowledgements.

Ioannis Boniatis was supported by a grant by the State Scholarship Foundation (SSF), Greece.

References

1. Arden, N., Nevitt, M.C.: Osteoarthritis: Epidemiology. Best Pract. Res. Clin. Rheumatol. 20 (2006) 3–25
2. Buckland-Wright, C.: Which Radiographic Techniques Should we Use for Research and Clinical Practice?. Best Pract. Res. Clin. Rheumatol. 20 (2006) 39–55
3. Gnther, K.P., Sun, Y.: Reliability of Radiographic Assessment in Hip and Knee Osteoarthritis. Osteoarthritis Cartilage 7 (1999) 239–246
4. Kellgren, J.H., placeCityLawrence, J.S.: Radiological Assessment of Osteoarthrosis. Ann. Rheum.
   Dis. 16 (1957) 494-501
5. Ingvarsson, T., Hgglund, G., Lindberg, H., Lohmander, L.S: Assessment of Primary Osteoarthritis: Comparison of Radiographic Methods Using placeCityColon Radiographs. Ann. Rheum. Dis. 59 (2000) 650-653
6. Boniatis, placeI., Costaridou, L., Cavouras, D., Kalatzis, placeI., Panagiotopoulos, E., Panayiotakis, G.: Osteoarthritis Severity of the Hip by Computer-Aided Grading of Radiographic Images. Med. Bio. placecountry-regionEng. Comput. 44 (2006) 793–803
7. Boniatis, placeI., Cavouras, D., Costaridou, L., Panagiotopoulos, E., Kalatzis, placeI., Panayiotakis, G.: A Decision Support System for the Automatic Assessment of Hip Osteoarthritis Severity by Hip Joint Space Contour Spectral Analysis. Lecture Notes in Computer Science (LNCS) 4345 (2006) 451-462
8. Pizer, S.M., Amburn, E.O.P., Austin, J.D., et al. Adaptive histogram equalization and its variations. CVGIP (39) 1987 355–368
9. Conrozier, T., Tron, A.M., Balblanc, J.C., et al. : Measurement of the Hip Joint Space Using Computerized Image Analysis. Rev. Rhum. Engl. Ed. 60 (1993) 105-111
10. Gonzalez, R.C., Woods, R.E.: Digital Image Processing. 2nd edn. Prentice Hall, Inc., placeStateNew Jersey (2002)
11. Theodoridis, S., Koutroumbas, K.: Pattern Recognition. 2nd edn. Elsevier Academic Press, placeCityAmsterdam (2003)
12. Specht D.F. Probabilistic neural networks. Neural Netw. 3 (1990) 109-118

A Fast Ensemble of Regressors

M. Karagiannopoulos, D. Anyfantis, S.B. Kotsiantis, and P.E. Pintelas

Educational Software Development Laboratory
Department of Mathematics, University of Patras, Hellas
{mariosk, dany, sotos, pintelas}@math.upatras.gr

Abstract. We have implemented a learning tool that combines the RepTree, the linear regression and the Decision Stump algorithms using the averaging methodology. We performed a large-scale comparison with other state-of-the-art algorithms and fast ensembles on several datasets and we took better accuracy in most cases using less time for training, too.

Introduction

Combining regressors is proposed as a new direction for the improvement of the accuracy of regression models [3]. However, ensembles need increased computation and a research area is to explore learning techniques for scaling up to large datasets. In this work, we try to bridge the gap by using fast weak algorithms for building a rapid ensemble. Section 2 discusses the proposed ensemble method and experiment results of the proposed ensemble with other learning. We conclude in Section 3.

Proposed Ensemble

The training time is often less for generating multiple weak regressors compared to training one strong regressor. This is because strong regressors spend a majority of their training time in fine tuning. Secondly, weak regressors are also less likely to suffer from overfitting problems. As far as the used learning algorithms of the proposed ensemble are concerned, three fast algorithms are used: 1) Linear regression (LR) [6], 2) RepTree [11] and 3) Decision stumps (DS) [8]. The corresponding predictions of the base regression models are then combined with averaging rule to produce the final decision. It must be also mentioned that the proposed ensemble can be easily distributed and parallelized. This parallel and distributed execution of the presented ensemble can achieve linear speedup. For our study, we used a number of well-known datasets by many domains from the UCI repository [1]. In order to calculate the models' correlation coefficient for our experiments, cross validation was run 10 times for each algorithm and the average value was calculated. It must be mentioned that we used the free available source code for most of the algorithms by [11] for our experiment.

During the experiment, the proposed ensemble was compared with a representative algorithm for each of the other sophisticated machine learning techniques: Back Propagation (BP) algorithm [11], SMOreg algorithm [5], Kstar algorithm [9] and decision table algorithm [10]. In Table 1 and Table 2, we represent with "v" that the proposed ensemble looses from the specific algorithm. That is, the specific algorithm performed statistically better than the proposed according to paired t-test with $p < 0.01$. Furthermore, in Tables, "*" indicates that proposed ensemble performed statistically better than the specific regressor according to paired t-test with pj0.01. In all the other cases, there is no significant statistical difference between the results (Draws). We also compare the proposed ensemble with other fast ensembles: Bagging RepTree, Bagging DS, Boosting RepTree and Boosting DS. Bagging is a method for building ensembles that uses different subsets of training data with a single learning method [3]. Additive Regression [7] is a practical implementation of the boosting [4].

Conclusion

The proposed ensemble needed less time for training than all the tested algorithms. The proposed ensemble can also achieve an increase in correlation coefficient from 2% to 17% compared to other learners. In a future work, the proposed ensemble will be made agent-based.

Table 1. Comparing the proposed ensemble with well known regressors

	AverageLRD	Kstar	DT	BP	SMOreg
auto93	0.80	0.77*	0.68*	0.85v	0.82
autoHorse	0.92	0.90	0.85*	0.95v	0.95v
autoMpg	0.90	0.91	0.90	0.91	0.92
autoPrice	0.90	0.91	0.81*	0.90	0.90
bodyfat	0.97	0.87*	0.97	0.98	0.99
breastTumor	0.27	0.19*	0.16*	0.09*	0.28
cholesterol	0.16	0.04*	0.07*	0.08*	0.16
cpu	0.97	0.97	0.92*	1.00v	0.97
echoMonths	0.71	0.39*	0.72	0.42*	0.68*
elusage	0.85	0.85	0.88v	0.86	0.84
hungarian	0.68	0.55*	0.59*	0.49*	0.58*
lowbwt	0.79	0.62*	0.78	0.60*	0.77
pbc	0.57	0.30*	0.40*	0.32*	0.58
pwLinear	0.89	0.72*	0.83*	0.90	0.86*
quake	0.10	0.08	0.09	0.08	0.06*
sensory	0.47	0.39*	0.57v	0.29*	0.35*

Table 2. Comparing the proposed ensemble with well known regressors

	AverageLRD	Bagging-RepTree	Boosting-RepTree	Bagging-DS	Boosting-DS
auto93	0.80	0.43*	0.26*	0.74*	0.79
autoHorse	0.92	0.89*	0.85*	0.80*	0.90
autoMpg	0.90	0.91	0.89	0.78*	0.90
autoPrice	0.90	0.92	0.90	0.82*	0.91
bodyfat	0.97	0.98	0.98	0.84*	0.97
breastTumor	0.27	0.22*	0.16*	0.23*	0.29
cholesterol	0.16	0.18	0.07*	0.12*	0.14
cpu	0.97	0.96	0.90*	0.87*	0.97
echoMonths	0.71	0.69	0.69	0.69	0.59*
elusage	0.85	0.82*	0.80*	0.84	0.83
hungarian	0.68	0.64*	0.58*	0.60*	0.67
lowbwt	0.79	0.79	0.77	0.78	0.77
pbc	0.57	0.55	0.46*	0.46*	0.53*
pwLinear	0.89	0.91	0.90	0.68*	0.85*
quake	0.10	0.12	0.06	0.09	0.08

References

1. C. Blake & C. Merz, UCI Repository of machine learning databases. Irvine, CA: University of California, Department of Information and Computer Science. http://www.ics.uci.edu/ mlearn/MLRepository.html
2. Breiman L., Bagging Predictors. Machine Learning 24 (1996) 123-140.
3. Gavin Brown, Jeremy Wyatt, and Peter Tino. Managing diversity in regression ensembles. Journal of Machine Learning Research, 6, 2005.
4. Chan Ph., Fan W., Prodromidis A., Stolfo, S., Distributed Data Mining in Credit Card Fraud Detection, IEEE Intelligent Systems on Data Mining, December 1999.
5. Duffy, N. Helmbold, D., Boosting Methods for Regression, Machine Learning, 47, 153-200, 2002
6. Gary William Flake, Steve Lawrence, Efficient SVM Regression Training with SMO, Machine Learning, Volume 46, Issue 1 - 3, Jan 2002, Pages 271 - 290.
7. Fox, J. (1997), Applied Regression Analysis, Linear Models, and Related Methods, ISBN: 080394540X, Sage Pubns.
8. Friedman J. (2002). "Stochastic Gradient Boosting," Computational Statistics and Data Analysis 38(4):367-378.
9. Iba, W., & Langley, P., Induction of one-level decision trees. Proceedings of the Ninth International Machine Learning Conference (1992). Aberdeen, Scotland: Morgan Kaufmann.
10. C. John and L. Trigg, K*: An Instance- based Learner Using an Entropic Distance Measure", Proc. of the 12th International Conference on ML, pp. 108-114, 1995.
11. Kohavi R. (1995). "The Power of Decision Tables." In Proc European Conference on Machine Learning.
12. Mitchell, T., Machine Learning. McGraw Hill (1997).

13. Shevade, S., Keerthi, S., Bhattacharyya C., and Murthy, K. (2000). Improvements to the SMO algorithm for SVM regression. IEEE Transaction on Neural Networks, 11(5):1188-1183.
14. Witten I. & Frank E., Data Mining: Practical Machine Learning Tools and Techniques with Java Implementations, Morgan Kaufmann, San Mateo (2000).

Spectral analysis of short and of non-equidistant time series using the Least–Squares technique

S. Pytharouli, P. Psimoulis, E. Kokkinou, and S. Stiros

Geodesy Lab., Dept. of Civil Engineering,
University of Patras, GR-26500, Patras, Greece
{spitha,stiros}@upatras.gr

Abstract. The Lomb Normalized Periodogram, a Least Squares-based technique and a FORTRAN computer code for spectral analysis of short time series consisting of non–equidistant values are presented. The application of this technique in the spectral analysis of a time series reflecting elevation changes of the deck of a short–span railroad bridge in Central Greece in response to a passing train is also presented. Analysis of data collected with robotic theodolite (RTS) with an updated built-in software indicates that even in the case of short time series (< 130 values) the Lomb Normalized Periodogram can lead to easy, reliable and statistically significant results.

Introduction

Fast Fourier Transforms (FFT) is the most common spectral analysis method, and its basic merit is its great efficiency and the little time needed for the computations. Still, this technique has two requirements which prevent from its use in several engineering and other applications. Time series should consist of equidistant values and should be periodic, ideally infinite in length [1, 2]. While the problem of non-equidistant data can be partly faced by interpolation techniques in cases where the gaps between adjacent values are relatively small, spectral analysis of a short (< 150 values) is a problem that cannot be easily solved using FFT.

A common technique to overcome this problem is adding a series of zeros at the beginning and the end of the time series ("zero padding"), so that a new time series with a minimum length of at least 256 values is formed. However this technique introduces additional noise and may lead to biased results [1]. Such an effect, obviously, is not suitable for instance in the case we investigate the response of an old structure in order to investigate whether or not it is safe to be used by the public (structural health monitoring). The need of an alternative technique to the FFT method for the spectral analysis of short-time series is evident. The Lomb Normalized Periodogram provides a satisfactory solution to this problem, for it also permits statistical check of its results.

The Lomb Normalized Periodogram

Lomb [3] and later Scargle [4] developed an algorithm for the spectral analysis of both evenly and unevenly data, as well as of short time series. This algorithm known as the *Lomb Normalized Periodogram* (LNP). LNP for a specific period T is defined by:

$$P(T) = \frac{1}{2\sigma^2}\left(\frac{\left(\sum_{j=1}^{N}(x_j-\bar{x})\cos\frac{2\pi(t_j-\tau)}{T}\right)^2}{\sum_{j=1}^{N}\left(\cos\frac{2\pi(t_j-\tau)}{T}\right)^2}+\frac{\left(\sum_{j=1}^{N}(x_j-\bar{x})\sin\frac{2\pi(t_j-\tau)}{T}\right)^2}{\sum_{j=1}^{N}\left(\sin\frac{2\pi(t_j-\tau)}{T}\right)^2}\right), \tag{1}$$

where the parameter τ is defined by the equation:

$$\tan\left(\frac{4\pi\tau}{T}\right)=\frac{\sum_{j=1}^{N}\sin\left(\frac{4\pi t_j}{T}\right)}{\sum_{j=1}^{N}\cos\left(\frac{4\pi t_j}{T}\right)}, \tag{2}$$

and

N is the number of points,
t_i is the time at which the displacement i was measured,
$\bar{x}$ is the mean of the data values $\bar{x}=\frac{1}{N}\sum_{i=1}^{N}x_i$, and
σ^2 is the variance of the data values $\sigma^2=\frac{1}{N-1}\sum_{i=1}^{N}(x_i-\bar{x})^2$.

This process is repeated for all values, and hence a spectrogram is produced. In addition, the significance level is defined by equation:

$$z_0 = -\ln\left[1-(1-p)^{\frac{1}{N}}\right], \tag{3}$$

where,

z_0 is the power level above which the value $P(T)$ of the LNP is statistically significant with $(1-p)\times 100\%$ confidence level,
p is the significance level, and
N is the number of data points.

Computations can be made with the FORTRAN-based Normperiod code [5].

Case Study: Spectral Analysis of the RTS monitoring record of a railway bridge

The application and effectiveness of the Lomb periodogram is highlighted in the following example. In order to investigate the response of the midspan of a 30m long steel railway bridge in Central Greece under dynamic loads (passing trains) we carried out a number of experiments. Using GPS (Global Positioning System) and RTS (robotic total station) technology, we recorded the movements

of a control point located on the middle span of the bridge before, during and after a passing train.

Data used in this study consist of RTS recordings of the vertical movements-response of the bridge to a passing train, Fig. 1(top). What is evident is that this time series consists of three parts: a first and third part reflecting measurement noise (apparent displacements before and after the passing of the train, indicative of the accuracy level in our data) and a middle section indicating a significant oscillation with duration of several tens of seconds. Analysis of this last section can permit to define the dynamic characteristics of this structure.

Spectral analysis was not possible using FFT, for the available time series was too short (approximately 120 values). Zero padding could be used, but it would lead to biased results, unacceptable for this particular case. Furthermore, data were not sampled at a constant rate, and hence a transformation of the available time series to a new one, based on interpolation techniques would be necessary before any processing. This would also lead to additional noise. For this reason we used the method of the Lomb Normalized Periodogram and the *Normperiod* code. The result of this spectral analysis is shown in Fig. 1(bottom). A dominant frequency equal to 0.45Hz was revealed. This frequency probably corresponds to the interaction between train and bridge deck [6].

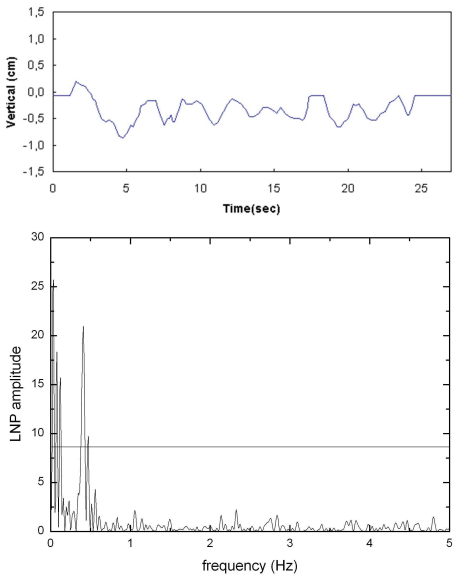


Fig. 1. Top: Elevation changes recorded by robotic theodolite (RTS) at the deck of a short-span railroad bridge in central Greece.
Bottom: The corresponding frequency spectrum using the *Normperiod* code. Straight line represents the 95% confidence level. A dominant frequency of 0.45Hz is revealed. Frequencies at the left edge of spectrum are not statistically significant (edge effect; [7]).

Conclusions

The above discussion indicates that the least-squares-based LNP is a powerful tool for the spectral analysis of short (< 150 values) and discontinuous time series without any interpolations, hence leading to low-noise results. In addition, the confidence levels of computed spectra can be determined. LNP and Normperiod code can therefore permit spectral analysis for data for which no reliable spectral analysis was possible in the past.

References

1. Press, W.H., Teukolsky, S.A., Vellerling, W.T., Flannery, B.P.: Numerical Recipies in C. The Art of Scientific Computing. Cambridge University Press (1988)
2. Proakis, J.G., Manolakis, D.: Digital Signal Processing. Prentice Hall, New York (1996)
3. Lomb, N.R.: Least–squares frequency analysis of unequally spaced data. Astrophysics and Space Science **39** (1976) 447–462
4. Scargle, J.D.: Studies in astronomical time series analysis. II. statistical aspects of spectral analysis of unevenly spaced data. The Astrophysical Journal **263** (1982) 179–184
5. Pytharouli, S.I., Stiros, S.: Spectral analysis of unevenly spaced or discontinuous data using the *normperiod* code. Computers and Structures (2007) in press,doi:10.1016/j.compstruc.2007.02.022.
6. Kokkinou, E., Psimoulis, P., Pytharouli, S., Stiros, S.: Smaller Alignment Index (SALI): Monitoring of the gorgopotamos steel railway bridge using a robotic total station (rts). In: Proceedings of the Conference 8th HSTAM International Congress on Mechanics Libration Point Orbits and Applications. (2007)
7. Pytharouli, S.I., Kontogianni, V.A., Stiros, S.C.: Kinematics of two deep seated landslides in greece. Proceedings of the Institute of Civil Engineers - Geotechnical Engineering **60**(GE3) (2007) 835–853

A Generalization of a Class of Matrices: Analytic Inverse and Determinant

F.N. Valvi

Department of Mathematics, University of Patras,
GR-26500 PATRAS, GREECE, fvalvi@upatras.gr

Abstract. The analytic form of the inverse for a class of matrices is given, The class represents a generalization of already known classes of matrices with elements defined by $4n-2$ parameters. Its inverse is found to be a lower Hessenberg matrix with elements expresssed analytically by these parameters. The analytic expression of the determinant is also provided and the numerical complexity in evaluating the inverse is discussed.

Introduction

In [1] a class of matrices $K_n = [a_{ij}]$ with elements

$$a_{ij} = \begin{cases} 1, & i \leqslant j, \\ a_j, & i > j, \end{cases}$$

is treated. A generalization of this class is presented in [2] by the matrix $G_n = [b_{ij}]$, where

$$b_{ij} = \begin{cases} b_j, & i \leqslant j, \\ a_j, & i > j. \end{cases}$$

In this paper, we consider a more extended class of matrices, M, and we deduce in analytic form its inverse and determinant. The class under consideration is defined by the Hadamard product of G_n and a matrix L, which results from G_n first by assigning the values $a_i = l_{n-i+1}$ and $b_i = k_{n-i+1}$ to the latter in order to get a matrix K, say, and then by the relation $L = PK^{\mathrm{T}}P$, where $P = [p_{ij}]$ is the permutation matrix with elements

$$p_{ij} = \begin{cases} 1, & i = n - j + 1, \\ 0, & \text{otherwise.} \end{cases}$$

The so constructed class is defined by $4n - 2$ parameters and its inverse has a lower Hessenberg analytic expression. By assigning particular values to these parameters, a great variety of test matrices occurs.

It is worth noting that the classes L and G_n that produce the class $M = L \circ G_n$ belong to the extended DIM classes presented in [3], as well as to the categories of the upper and lower Brownian matrices, respectively, as they have been defined in [4].

The Class of Matrices and its Inverse

Let $M = [m_{ij}]$ be the matrix with elements

$$m_{ij} = \begin{cases} k_i b_j, & i \leqslant j, \\ l_i a_j, & i > j, \end{cases} \tag{1}$$

that is,

$$M = \begin{bmatrix} k_1b_1 & k_1b_2 & k_1b_3 & \dots & k_1b_{n-1} & k_1b_n \\ l_2a_1 & k_2b_2 & k_2b_3 & \dots & k_2b_{n-1} & k_2b_n \\ l_3a_1 & l_3a_2 & k_3b_3 & \dots & k_3b_{n-1} & k_3b_n \\ \dots & & & & & \\ l_{n-1}a_1 & l_{n-1}a_2 & l_{n-1}a_3 & \dots & k_{n-1}b_{n-1} & k_{n-1}b_n \\ l_na_1 & l_na_2 & l_na_3 & \dots & l_na_{n-1} & k_nb_n \end{bmatrix}.$$

If $M^{-1} = [\mu_{ij}]$ is its inverse, the following expressions determine its elements

$$\mu_{ij} = \begin{cases} \dfrac{k_{i+1}b_{i-1} - l_{i+1}a_{i-1}}{c_{i-1}c_i}, & i = j = 2, 3, \dots, n-1, \\ \dfrac{k_2}{c_0c_1}, & i = j = 1, \\ \dfrac{b_{n-1}}{c_{n-1}c_n}, & i = j = n, \\ (-1)^{i+j} \dfrac{d_{j-1}g_i \prod\limits_{\nu=j+1}^{i-1} f_\nu}{\prod\limits_{\nu=j-1}^{i} c_\nu}, & i > j, \\ -\dfrac{1}{c_i}, & i = j-1, \\ 0, & i < j-1, \end{cases} \tag{2}$$

where

$$\begin{cases} c_i = k_{i+1}b_i - l_{i+1}a_i, & i = 1, 2, \dots, n-1, \quad c_0 = k_1, \quad c_n = b_n, \\ d_i = a_{i+1}b_i - a_ib_{i+1}, & i = 1, 2, \dots, n-2, \quad d_0 = a_1, \\ f_i = l_ia_i - k_ib_i, & i = 2, 3, \dots, n-1, \\ g_i = k_{i+1}l_i - k_il_{i+1}, & i = 2, 3, \dots, n-1, \quad g_n = l_n, \end{cases} \tag{3}$$

with

$$\prod_{\nu=j+1}^{i-1} f_\nu = 1 \quad \text{whenever} \quad i = j+1, \tag{4}$$

and with the obvious assumption

$$c_i \neq 0, \quad i = 0, 1, 2, \dots, n. \tag{5}$$

The Determinant of M

The determinant of M takes the form

$$\det(M) = k_1 b_n (k_2 b_1 - l_2 a_1) \dots (k_n b_{n-1} - l_n a_{n-1}) .$$

Evidently, M is singular if $c_i = 0$ for some $i \in \{0, 1, 2, \dots, n\}$.

Numerical Complexity

The inverse of the matrix M is given explicitly by the expressions (2). However, a careful reader could easily derive the recursive algorithm that gives the elements under the main diagonal of M^{-1}. In particular,

$$\mu_{i,i-1} = - \frac{d_{i-2} g_i}{c_{i-2} c_{i-1} c_i}, \quad i = 2, 3, \dots, n,$$

$$\mu_{i,i-s-1} = - \left(\frac{d_{i-s-2} f_{i-s}}{d_{i-s-1} c_{i-s-2}} \right) \mu_{i,i-s}, \quad i = 3, 4, \dots, n, \quad s = 1, 2, \dots, i-2,$$

where the c_i, d_i, f_i, and g_i are given by the relations (3). By use of the above algorithms, the estimation of the whole inverse of the matrix M is carried out in $2n^2 + 11n - 19$ multiplications/divisions, since the coefficient of μ_{ij} depends only on the second subscript, and in $5n - 9$ additions/subtractions.

References

1. Milnes, H. W.: A note concerning the properties of a certain class of test matrices. Math. Comp. **22** (1968) 827–832
2. Herbold, R. J.: A generalization of a class of test matrices. Math. Comp. **23** (1969) 823–826
3. Carayannis, G., Kalouptsidis, N., & Manolakis, D. G.: Fast recursive algorithms for a class of linear equations. IEEE Trans. Acoust. Speech Sig. Proc. **30** (1982) 227–239
4. Gover, M. J. C., & Barnett, S.: Brownian matrices: properties and extensions. Int. J. Systems Sci. **17** (1986) 381–386

On Two Brownian–type Matrices with Explicit Hessenberg Inverses

F.N. Valvi[1] and V.S. Geroyannis[2]

[1] Department of Mathematics, University of Patras,
GR-26500 PATRAS, GREECE, fvalvi@upatras.gr
[2] Department of Physics, University of Patras,
GR-26500 PATRAS, GREECE, vgeroyan@upatras.gr

Abstract. In this paper, we present explicit inverses for two Brownian–type matrices, which are defined as Hadamard products of certain already known matrices. The matrices under consideration are defined by $3n-1$ parameters and their lower–Hessenberg–form inverses are expressed analytically in terms of these parameters. Such matrices are useful in the theory of digital signal processing and in the theory and applications of test matrices, i.e., matrices with known explicit inverses, which are thus appropriate for testing matrix inversion algorithms.

Introduction

[1] gives the explicit inverse of a matrix $G_n = [\beta_{ij}]$ with elements

$$\beta_{ij} = \begin{cases} b_j, & i \leqslant j, \\ a_j, & i > j. \end{cases}$$

[2] gives the explicit inverses of two symmetric matrices $K = [\kappa_{ij}]$ and $N = [\nu_{ij}]$ with elements

$$\kappa_{ij} = k_i \quad \text{and} \quad \nu_{ij} = k_j, \quad i \leqslant j,$$

respectively. K is a special case of Brownian–type matrix and G_n is a lower Brownian matrix as defined in [3]. Earlier, [4] has used the term "pure Brownian matrix" for the type of the matrix K; and [5] has treated the so-called "diagonal innovation matrices" (DIM), special cases of which are the matrices K and N.

In the present paper, we consider two matrices A_1 and A_2 defined by

$$A_1 = K \circ G_n \quad \text{and} \quad A_2 = N \circ G_n,$$

where the symbol $\circ$ denotes the Hadamard product. These matrices have the explicit forms

$$A_1 = \begin{bmatrix} k_1b_1 & k_1b_2 & k_1b_3 & \dots & k_1b_{n-1} & k_1b_n \\ k_1a_1 & k_2b_2 & k_2b_3 & \dots & k_2b_{n-1} & k_2b_n \\ k_1a_1 & k_2a_2 & k_3b_3 & \dots & k_3b_{n-1} & k_3b_n \\ \dots & & & & & \\ k_1a_1 & k_2a_2 & k_3a_3 & \dots & k_{n-1}b_{n-1} & k_{n-1}b_n \\ k_1a_1 & k_2a_2 & k_3a_3 & \dots & k_{n-1}a_{n-1} & k_nb_n \end{bmatrix} \tag{1}$$

and

$$A_2 = \begin{bmatrix} k_1b_1 & k_2b_2 & k_3b_3 & \dots & k_{n-1}b_{n-1} & k_nb_n \\ k_2a_1 & k_2b_2 & k_3b_3 & \dots & k_{n-1}b_{n-1} & k_nb_n \\ k_3a_1 & k_3a_2 & k_3b_3 & \dots & k_{n-1}b_{n-1} & k_nb_n \\ \dots & & & & & \\ k_{n-1}a_1 & k_{n-1}a_2 & k_{n-1}a_3 & \dots & k_{n-1}b_{n-1} & k_nb_n \\ k_na_1 & k_na_2 & k_na_3 & \dots & k_na_{n-1} & k_nb_n \end{bmatrix}. \tag{2}$$

In the following sections, we give the explicit inverses and determinants of these matrices.

The Inverse and Determinant of A_1

The inverse of A_1 is a lower Hessenberg matrix expressed analytically by the $3n-1$ parameters defining A_1. In particular, the inverse $A_1^{-1} = [\alpha_{ij}]$ of A_1 has elements given by the relations

$$\alpha_{ij} = \begin{cases} \dfrac{k_{i+1}b_{i-1} - k_{i-1}a_{i-1}}{c_{i-1}c_i}, & i = j \neq 1, n, \\ \dfrac{k_2}{k_1c_1}, & i = j = 1, \\ \dfrac{b_{n-1}}{c_{n-1}c_n}, & i = j = n, \\ (-1)^{i+j}\dfrac{d_{j-1}g_i\displaystyle\prod_{\nu=j+1}^{i-1} k_\nu f_\nu}{\displaystyle\prod_{\nu=j-1}^{i} c_\nu}, & i - j \geqslant 1, \\ -\dfrac{1}{c_i}, & j - i = 1, \\ 0, & j - i > 1, \end{cases} \tag{3}$$

where

$$\begin{cases} c_i = k_{i+1}b_i - k_ia_i, & i = 1, 2, \dots, n-1, \quad c_0 = 1, \quad c_n = b_n, \\ d_i = k_{i+1}b_ia_{i+1} - k_ia_ib_{i+1}, & i = 1, 2, \dots, n-2, \quad d_0 = a_1, \\ f_i = a_i - b_i, & i = 2, 3, \dots, n-1, \\ g_i = k_{i+1} - k_i, & i = 2, 3, \dots, n-1, \quad g_n = 1, \end{cases} \tag{4}$$

with

$$\prod_{\nu=j+1}^{i-1} k_\nu f_\nu = 1 \quad \text{if} \quad i = j+1,$$

and with the obvious assumptions

$$k_1 \neq 0 \quad \text{and} \quad c_i \neq 0, \quad i = 1, 2, \ldots, n.$$

The determinant of A_1 takes the form

$$\det(A_1) = k_1 b_n (k_2 b_1 - k_1 a_1)(k_3 b_2 - k_2 a_2) \ \ldots \ (k_n b_{n-1} - k_{n-1} a_{n-1}).$$

Evidently, A_1 is singular if $k_1 = 0$ or, using the relations (4), $c_i = 0$ for some $i \in \{1, 2, \ldots, n\}$.

The Inverse and Determinant of A_2

In the case of A_2, its inverse $A_2^{-1} = [\alpha_{ij}]$ is a lower Hessenberg matrix with elements given by the relations

$$\alpha_{ij} = \begin{cases} \dfrac{k_{i-1} b_{i-1} - k_{i+1} a_{i-1}}{c_{i-1} c_i}, & i = j \neq 1, n, \\ \dfrac{1}{c_1}, & i = j = 1, \\ \dfrac{k_{n-1} b_{n-1}}{k_n c_{n-1} c_n}, & i = j = n, \\ (-1)^{i+j} \dfrac{d_{j-1} g_i \displaystyle\prod_{\nu=j+1}^{i-1} k_\nu f_\nu}{\displaystyle\prod_{\nu=j-1}^{i} c_\nu}, & i - j \geqslant 1, \\ -\dfrac{1}{c_i}, & j - i = 1, \\ 0, & j - i > 1, \end{cases} \tag{5}$$

where

$$\begin{cases} c_i = k_i b_i - k_{i+1} a_i, & i = 1, 2, \ldots, n-1, \quad c_0 = 1, \quad c_n = b_n, \\ d_i = k_i b_i a_{i+1} - k_{i+1} a_i b_{i+1}, & i = 1, 2, \ldots, n-2, \quad d_0 = a_1, \\ f_i = a_i - b_i, & i = 2, 3, \ldots, n-1, \\ g_i = k_i - k_{i+1}, & i = 2, 3, \ldots, n-1, \quad g_n = 1, \end{cases} \tag{6}$$

with

$$\prod_{\nu=j+1}^{i-1} k_\nu f_\nu = 1 \quad \text{if} \quad i = j + 1,$$

and with the obvious assumptions

$$k_n \neq 0 \quad \text{and} \quad c_i \neq 0, \quad i = 1, 2, \ldots, n.$$

The determinant of A_2 has the form

$$\det(A_2) = k_n b_n \left(k_1 b_1 - k_2 a_1\right)\left(k_2 b_2 - k_3 a_2\right)\ldots\left(k_{n-1} b_{n-1} - k_n a_{n-1}\right),$$

which shows in turn that the matrix A_2 is singular if $k_n = 0$ or, adopting the conventions (6), $c_i = 0$ for some $i \in \{1, 2, \ldots, n\}$.

Numerical Complexity

The relations (3) and (5) lead to recurrence formulae, by which the inverses A_1^{-1} and A_2^{-1}, respectively, are computed in $O(n^2)$ multiplications/divisions and $O(n)$ additions/substractions. In fact, the recursive algorithm

$$\alpha_{i,i+1} = -1/c_i, \quad i = 1, 2, \ldots, n-1,$$

$$\alpha_{ii} = -\alpha_{i,i+1} + \frac{b_{i-1} g_i}{c_{i-1} c_i}, \quad i = 2, 3, \ldots, n-1, \quad \alpha_{11} = \frac{k_2}{k_1 c_1}, \quad \alpha_{nn} = \frac{b_{n-1}}{c_{n-1} c_n},$$

$$\alpha_{i,i-1} = -\frac{d_{i-2} g_i}{c_{i-2} c_{i-1} c_i}, \quad i = 2, 3, \ldots, n,$$

$$\alpha_{i,i-s-1} = -\left(\frac{d_{i-s-2} k_{i-s} f_{i-s}}{d_{i-s-1} c_{i-s-2}}\right) \alpha_{i,i-s}, \quad i = 3, 4, \ldots, n, \quad s = 1, 2, \ldots, i-2,$$

where the c_i, d_i, f_i, and g_i are given by the relations (4), computes A_1^{-1} in $5n^2/2 + 5n/2 - 6$ mult/div (since the coefficients of $\alpha_{i,i-s}$ depends only on the second subscript) and $5n - 9$ add/sub.

For the computation of A_2^{-1} the above algorithms change only in the estimation of the diagonal elements, for which we have

$$\alpha_{ii} = -\alpha_{i,i+1} + \frac{a_{i-1} g_i}{(c_{i-1} c_i)}, \quad i = 2, 3, \ldots, n-1, \quad \alpha_{11} = -\alpha_{12}, \quad \alpha_{nn} = \frac{k_{n-1} b_{n-1}}{k_n c_{n-1} c_n},$$

where the c_i, d_i, f_i, and g_i are given by the relations (6). Therefore, considering the relations (4) and (6), it is clear that the number of mult/div and add/sub in computing A_2^{-1} is the same with that of A_1^{-1}.

References

1. Herbold, R. J.: A generalization of a class of test matrices. Math. Comp. **23** (1969) 823–826
2. Valvi, F. N.: Explicit presentation of the inverses of some types of matrices. J. Inst. Maths. Applics. **19** (1977) 107–117
3. Gover, M. J. C., & Barnett, S.: Brownian matrices: properties and extensions. Int. J. Systems Sci. **17** (1986) 381–386
4. Picinbono, B.: Fast algorithms for Brownian matrices. IEEE Trans. Acoust. Speech Sig. Proc. **31** (1983) 512–514
5. Carayannis, G., Kalouptsidis, N., & Manolakis, D. G.: Fast recursive algorithms for a class of linear equations. IEEE Trans. Acoust. Speech Sig. Proc. **30** (1982) 227–239

Author Index