Approximation and Extrapolation of Convergent and Divergent Sequences and Series
CIRM Luminy - September 28, 2009 - October 2, 2009

# Extrapolation Methods: a tool for accelerating real life problems

R. BERTELLE[a], M.R. RUSSO[a], M. VENTURIN[b]

[a]Department of Pure and Applied Mathematics, University of Padua, Italy [b]Department of Computer Science, University of Verona, Italy

## Introduction

Electronic has a greater and greater role in nowadays life. New products, with better functionality, appear almost every day. Moreover, their lifetimes are extremely short. This requires a constant and fast development of the basic electronic devices, from which the electronic machines are made on. This low time to market necessarily needs device simulations. Fluid simulations are one of the most interesting fields since we live dip into fluids. An aircraft flies into air which may be seen as an inviscid compressible fluid. A wind power plant uses wind to produce electricity. Water, an incompressible fluid, flows inside aqueducts from the spring to our houses. Also our blood may be modeled as a complicated fluid. All these aspects may be optimized and understood better via computer simulations.
The convergence of the simulation algorithms is sometimes very slow. In such cases is essential to improve the rate of convergence and this can be done by vector extrapolation methods.

## Vector Extrapolation Methods

Vector extrapolation methods accelerate slowly or non convergent vector sequences. We consider here the Poliynomial Methods. Let $\mathbf{x}_j$, $j = 0, 1, 2 \ldots$ be a sequence of N-dimensional column vectors with limit $\mathbf{s}$.
A polynomial algorithm finds $\mathbf{s}$ as a weighted average of $k+1$ terms related with the given sequence, where the $k$ independent weights are found by solving a linear system of size $(N \times k)$. In detail this means:

$$\mathbf{u}_j = \mathbf{x}_{j+1} - \mathbf{x}_j \qquad \mathbf{v}_j = \mathbf{u}_{j+1} - \mathbf{u}_j$$
$$U = [\mathbf{u}_0, \mathbf{u}_1, \ldots, \mathbf{u}_{k-1}] \qquad V = [\mathbf{v}_0, \mathbf{v}_1, \ldots, \mathbf{v}_{k-1}]$$

### MPE–Minimal Polynomial Extrapolation:

$$\mathbf{s} = \sum_{j=0}^{k} \gamma_j \mathbf{x}_j \quad \text{with} \quad \gamma_j = \frac{c_j}{\sum_{i=0}^{k} c_i}, \quad j = 0, 1, \ldots, k$$

where the coefficients $c_j$, $j = 0, \cdots, k$ are obtained solving the linear system $U\mathbf{c} = -\mathbf{u}_k$.

### RRE–Reduced Rank Extrapolation:

$$\mathbf{s} = \mathbf{x}_0 + \sum_{i=0}^{k} \xi_i \mathbf{u}_i$$

where the coefficients $\xi_i$, $i = 0, \cdots, k$ are obtained solving the linear system $V\xi = -\mathbf{u}_0$ .

Remarks:

- The explicit knowledge of the sequence generator is not required.
- $\mathbf{s}$ is find solving a relatively small linear system (i.e., $k << N$).

## Semiconductor devices simulation: MOSFET

One of the most important electronic device is the MOSFET (Metal Oxide Field Effect Transistor). To understand its relevance on our life, it suffice to say that a large amount of the hardware inside a computer is based on it. For example, a microprocessor contains about 400 million mosfets. In a very crude way, an n-channel mosfet works as follows. Assume $V_{th} > 0$ be a given threshold voltage.

1. $V_g \leq V_{th}$: the electric field is too low to attract under the oxide a sufficiently large number of source and drain electrons to realize the inversion in the p-silicon region just under the oxide (i.e, the p-silicon becomes an n silicon). Thus, the source and the drain are separated and no current flows throughout the source and drain contacts.
2. $V_g > V_{th}$: the electric field is high enought to create an inversion under the oxide. The drain and the source are joned by the same n-type of semiconductor material (thus, the name n-channel mosfet). A current can flow from source to drain if $V_D - V_S \neq 0$.

## Semiconductor devices simulation: MOSFET

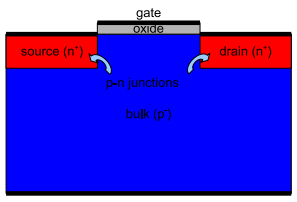



FIGURE 1. $V_g \leq V_{th}$

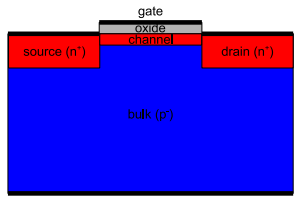
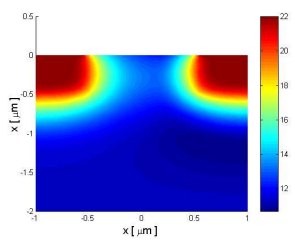
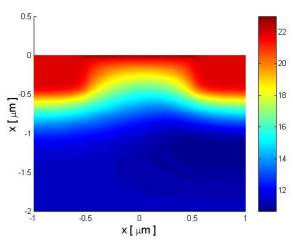



FIGURE 2. $V_g > V_{th}$




## Extrapolation of the Gummel map

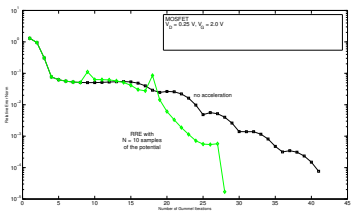
The model of a semiconductor device is a system of three strongly coupled differential equations: the Poisson equation for the potential $\psi$ and the two continuity equations for hole and electron concentrations, $n$ and $p$, respectively. $n = n_i e^{\psi - \phi_n}$ and $p = n_i e^{\phi_p - \psi}$ where $n_i$ is a given reference concentration and $\phi_n$ and $\phi_p$ are the electron and hole quasi-Fermi potentials. The Gummel map is the common, iterative, way used to solve this system. At each step, the Poisson and the continuity equations are solved and the solutions updated until some specified stopping criteria is fullfilled.
For the mosfet case, the number of iteration steps is low for $V_g \approx 0\,V$ and rises with $V_g$. In this latter case, it is interesting to apply some acceleration technique as we can see in the modified Gummel map algorithm.

guess $\psi^{(0)}, \phi_n^{(0)}, \phi_p^{(0)}$
$k = 0$
**while** (some stop criterion is not matched)

1. $-\nabla \cdot (\epsilon \nabla \psi^{(k+1)}) = q[p^{(k)} - n^{(k)} + C]$
2. collect $\psi^{(k+1)}$ and/or extrapolate a new value for $\psi^{(k+1)}$
3. $\nabla \cdot [-q\mu_n n^{(k+1)} \nabla \psi^{(k+1)} + qD_n \nabla n^{(k+1)}] = R(\psi^{(k)}, n^{(k)}, p^{(k)})$
4. collect $\phi_n^{(k+1)}$ and/or extrapolate a new value for $n^{(k+1)}$
5. $\nabla \cdot [q\mu_p p^{(k+1)} \nabla \psi^{(k+1)} + qD_n \nabla p^{(k+1)}] = -R(\psi^{(k)}, n^{(k)}, p^{(k)})$
6. collect $\phi_p^{(k+1)}$ and/or extrapolate a new value for $p^{(k+1)}$
7. $k = k + 1$

**end**

## Numerical Results (1 $\mu m$ channel width )




number $N_\psi$ of consecutive collected $\psi^{(k)}$ vectors for RRE extrapolation

| $N_\psi$ | no extrapolation | 6 | 7 | 8 | 9 | 10 | 11 | 12 |
|---|---|---|---|---|---|---|---|---|
| CPU time [ s ] | 59 | 48 | 37 | 34 | 36 | 41 | 41 | 49 |

## Fluid simulations

The mathematical model for both compressible and incompressible fluids contains continuity and momentum equations and is completed in the compressible fluids by a temperature equation whereas for the incompressible ones by an energy equation.
The numerical solution may be carried out using an iterative procedure, the Characteristic-Based Split (CBS) scheme. It is a fractional time-stepping algorithm based on an original finite difference velocity-projection scheme where the convective terms are treated using the idea of the characteristic-Galerkin method. The number of iteration may be very high and thus we propose an accelerated version of the algorithm.

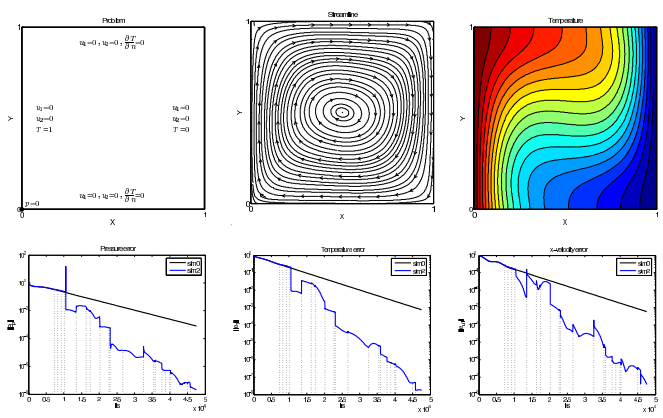
## Extrapolation of CBS scheme

There are no general guidelines to accelerate the convergence of evolutive problems that converge to a stationary solution. So, we adopt the following strategies:

- We select the variables to check where the extrapolation procedure should be done. Regardless this choice, all the solution variables are extrapolated.
- We skip the first iterations of the iterative process since they are, reasonable, the most far away from the stationary solution
- We collect consecutive iterations characterized by the fact that the maximum error for each variable between two consecutive iterations occurs at the same mesh nodes. The errors between consecutive iterations are proportional and hence it is reasonably that it appears at the same nodes of the mesh. This way to check the extrapolation point is very less, memory and time, consuming.
- We perform extrapolation only on the last part of the collected iterations.
- Finally, since the extrapolated solution, typically, is not a feasible solution for the evolution problem (i.e., it does not satisfy the PDE equations), we give some iterations to relax towards a feasible solution before looking back again to original sequence for finding another starting point before collecting further iterations.

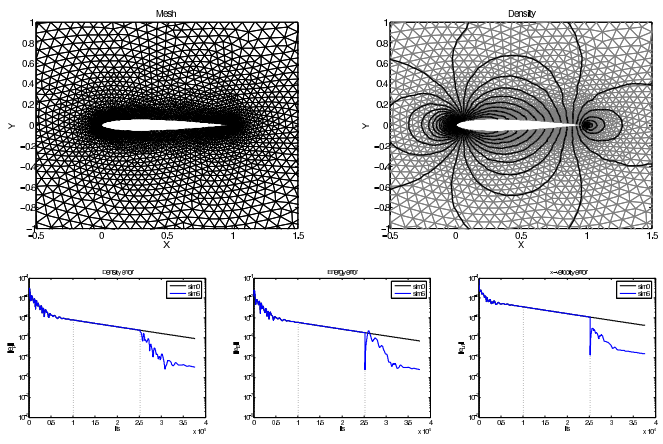
## Buoyancy–Driven Convection Problem

In this problem the Navier–Stokes equations are coupled with the temperature equation. The local temperature difference induces a local density difference within the fluid and this produce the fluid motion.




## NACA0012 Airfoil Problem

This problem refers to an inviscid compressible flow over a NACA0012 airfoil test problem at Mach number 0.25.




## References

[1] R. BERTELLE, M.R. RUSSO, *An approach to the Gummel map by vector extrapolation methods*, Numer. Algor. 45 (2007).

[2] R. BERTELLE, M.R. RUSSO, M. VENTURIN, *An accelerated algorithm for Navier-Stokes equations*, (2009), submitted

[3] C. BREZINSKI, M. REDIVO ZAGLIA, *Extrapolation methods. Theory and practice.*, North–Holland Publishing Co., Amsterdam, (1991).

[4] M. MORANDI-CECCHI, M. VENTURIN, *Characteristic-based split (CBS) algorithm finite element modelling for shallow waters in the Venice lagoon*, Int. J. Numer. Meth. Engrg. 66 (2006).

[5] A. SIDI, *Efficient Implemenation of Minimal Polynomial and Reduced Rank Extrapolation Methods*, J. Comput. Applied Math. 36 (1991)