



POLITECNICO
MILANO 1863

Anderson Acceleration for Fixed-Point Iterations

Martino Ischia

Supervisor: Prof. Formaggia

January 2021

Abstract

The goal of this project is twofold.
Firstly, develop a C++ interface for accelerating a converging sequence of vectors. It should be at the same time open for extensions, in case the user wants to implement its own algorithm, and efficient in the solution of large systems.
Secondly, making use of said interface to implement Anderson acceleration strategy and apply it on ...

The reader not interested in the C++ implementation, exposed in section 2, can safely skip it.

Contents

1	Introduction	2
1.1	Anderson Algorithm	2
2	C++ interface for a generic accelerator	3
3	FEM application	3
	Bibliography	3

1 Introduction

A fixed-point problem consists in finding the point (a vector in \mathbb{R}^n) that satisfies the following:

$$x = g(x)$$

where g is a function from \mathbb{R}^n to \mathbb{R}^n .

It is clearly equivalent to the problem of finding the roots of a generic function, that is often tackled by employing Newton-Raphson iterative algorithm. In many practical problems, though, the cost of computing the Jacobian of a function is not practical, not to mention that there might be problems in starting the iterations from a proper guess.

Considering the fixed-point form and solving through fixed-point iterations has also some limitations, the main one being a slow convergence: in fact most of the times the convergence is linear.

Several strategies have been described in the literature to improve the speed of convergence of a vector sequence: we could refer to them as acceleration methods. This project focuses on one of them, proposed by Anderson [1] in 1965.

1.1 Anderson Algorithm

There are many equivalent ways to formulate Anderson acceleration method. I report the one in Walker and Ni [3].

Several adjustment can be made to address specific problems, but

Given x_0 and $m \geq 1$

Set $x_1 = g(x_0)$

for $k = 1, 2, \dots$ *until convergence* **do**

Set $m_k = \min\{m, k\}$

Set $F_k = (f_{k-m_k}, \dots, f_k)$, where $f_i = g(x_i) - x_i$

Determine $\alpha^{(k)} = (\alpha_0^{(k)}, \dots, \alpha_{m_k}^{(k)})^T$, subject to $\sum_{i=0}^{m_k} \alpha_i = 1$, that solves

$$\min_{\alpha = (\alpha_0, \dots, \alpha_{m_k})^T} \|F_k \alpha\|_2$$

Set $x_{k+1} = (1 - \beta) \sum_{i=0}^{m_k} \alpha_i^{(k)} x_{k-m_k+i} + \beta \sum_{i=0}^{m_k} \alpha_i^{(k)} g(x_{k-m_k+i})$

end for

The new value is obtained through a linear combination of the previous iterates and their evaluations. The coefficients are found through a minimization problem, in this case written as a constrained

problem, but an unconstrained form is usually chosen for the implementation [2][3].

Notice that the algorithm depends on two parameters: β , a relaxation parameter that has also a special interpretation when the algorithm is seen as a multisecant method [2], and m , a memory parameter, that represents the number of previous iterates considered in the calculations.

2 C++ interface for a generic accelerator

aaaa

3 FEM application

aaa

Bibliography

- [1] Donald Anderson. “Iterative Procedures for Nonlinear Integral Equations”. In: *J. ACM* 12 (Oct. 1965), pp. 547–560. DOI: 10.1145/321296.321305.
- [2] Haw-ren Fang and Yousef Saad. “Two classes of multisecant methods for nonlinear acceleration”. In: *Numerical Linear Algebra with Applications* 16 (Mar. 2009), pp. 197–221. DOI: 10.1002/nla.617.
- [3] Homer Walker and Peng Ni. “Anderson Acceleration for Fixed-Point Iterations”. In: *SIAM J. Numerical Analysis* 49 (Aug. 2011), pp. 1715–1735. DOI: 10.2307/23074353.