Solving algebraic equation by vector field integration

Olivier Ruatta

10 septembre 2003

Résumé

The present document deals with a new approach to approximate simultaneously all the roots of an univariate polynomial with complex coeffiecients. The main result of the paper is to associate to a monic polynomial f of degree d a vector field on a Zariski open subset of $\mathbb{C}^d$. The only singularities of this vector field (which are attractive) are formed with the roots of the polynomial f we want to solve. We also discus the integrationof this vector field in order to approximate one of this vector and recover an approximation of the all the roots of f.

1 Introduction

In this document, we introduce a new approach to numerical solving of polynomial equations. Our approach consists in contructing a vector field. The fixed points of this vector field are attractive, isolated and are points having the set of roots of the input polynomials as coordinates. Using numerical or symbolic-numeric integration, we compute an approximation of an orbit for this vector field. We show that each orbit has one of the fixed point of the vector field in its closure. So from the approximation of the orbit, we are able to deduce an approximation of one of the fixed point, and equivalently an approximation of all the roots of the polynomial we want to solve.

The two most popular approaches to the problem of giving an approximation of the roots of univariate polynomials are the Shönhage-Weyl approach (see [8] for an account on this approach) and the Smale-Li one (we refer to [3] for this one). Other approaches are known but less used. The method proposed here can be seen as a continuous analog of the Weierstrass method. A relatively close point of view can be found in [1] and [2]. But, to the knowledge of the authors, the approach proposed here is new.

The method proposed here is related to the so-called Weierstrass method (also known as Durand-Kerner or Dochev method). It is a numerical iterative method which allows to approximate simultaneously all the roots of a monic polynomial with complex coefficients. The begining of the work presented here takes its sources in the study of the dynamic of the Weierstrass iteration and, to be more precise, in our study of the global behavior of this method. Since McMullen [7, 5], it is known that there is no generally convergent purely iterative algorithm for finding a root of polynomials of degree 5 or more. Let p be a polynomial, a purely iterative algorithm associated to p is a rational function $T_p(z)$ (depending rationally on the coefficient of p), such that under iteration $T_p^n(z)$ tends to a root of p for most choices of p and p. This result is strongly related to the dynamic on the complex plane. The weierstrass iteration is a multivariate purely iterative algorithm (its a rational map on the coefficients of p with d variables and d coordinate if p is of degree d, i.e. it is a rational map $T_p(z_1, \ldots, z_d) = (T_{1,p}(z_1, \ldots, z_d), \ldots, T_{d,p}(z_1, \ldots, z_d)))$ and is not concerned by the McMullen result. Furthermore, the method gives good experimental results if we do not take numerical problems into account. This leads us to the following question : "Is the Weierstrass method generally conevrgent ?". In other words, does the Weierstrass method converge to the set of the roots for most of the choises of p and sets of initial values $(z_1, \ldots, z_d)$. In the present document we do not study directly this problem which will be the object of futur works. But the method presented here come from our interest in that question and is closely related to this.

The method proposed here is a numerical integration of a special vector field associated to the polynomial we want to solve. This vector field is derived from the iteration function of the Weierstrass method. We call this vector field the Weierstrass vector field. By numerical integration of the Weierstrass vector field we are leaded to a method to approximate simultaneously all roots of a polynomial. It is clear that we obtain different algorithms using different algorithm for the integration of the Weierstrass vector field. The familly obtain by this way is not purely iterative. Nethertheless, each algorithm of this familly can be easilly represented as a "real automata" in the Blum-Cucker-Shub-Smale model [4]. We show that each algorithm is generally convergente, i.e. for almost choices of p and vector of complex values $(z_1, \ldots, z_d)$ the algorithm returns a vector formed with the set of the roots of p. An important property of this familly of algorithms is that for each p with only simple roots we can decide if the method will converge for a given input vector $(z_1, \ldots, z_d)$, since the set of points for which the method is not convergente is an explicit semi-algebraic set. For a rational function $T_p(z)$ which is globally convergente (i.e. for almost choices of z), decide if $T_p^n(z)$ will converge to a root of p is untractable [4]. The approach we describe here presents also interesting theorical developpements, but this will be the objet of an other work.

Our exposition is organized as follows : In the first section we recall the classical Weierstrass method. In the second section we introduce the Weierstrass vector field and discuss some of its properties. Finally, we present a first algorithm based on Euler method.

2 The classical Weierstrass's iteration

In this section we quickly present a construction of the Weierstrass's iteration. We choose the algebraic approach because it is synthetic since we are only interested in polynomials here. There is others approaches, more general but less clearer in our mind. We consider a degree d monic polynomial :

$$f(x) = \prod_{i=1}^{d}(x-\zeta_i) = x^d + \sum_{i=1}^{d} a_i x^{d-i} \in \mathbb{C}[x],$$

with only simple roots, i.e. $\zeta_i \neq \zeta_j$ for all $i \neq j \in \{1,\ldots,d\}$. For all $\boldsymbol{z} = (z_1,\ldots,z_d) \in \mathbb{C}^d$, we denote $\sigma_i(\boldsymbol{z}) = (-1)^i \sum_{1\leqslant j_1<\ldots<j_i\leqslant d} z_{j_1}\cdots z_{j_i}$ the i-th elementary symmetric function (or Viête's function), $\forall i \in \{1,\ldots,d\}$. We the define the following map :

$$\Sigma : \begin{cases} \mathbb{C}^d \longrightarrow \mathbb{C}^d \\ \boldsymbol{z} \longmapsto \Sigma(\boldsymbol{z}) = \begin{pmatrix} \sigma_1(\boldsymbol{z}) - a_1 \\ \vdots \\ \sigma_d(\boldsymbol{z}) - a_d \end{pmatrix} \end{cases} \tag{1}$$

We $\mathcal{S}_d$ denotes the classical permutation group of the set $\{1,\ldots,d\}$. This group act of $\mathbb{C}^d$ by $\forall \boldsymbol{z} = (z_1,\ldots,z_d) \in \mathbb{C}^d$ and $\tau \in \mathcal{S}_d$, we define $\boldsymbol{z}^\tau = (z_{\tau(1)},\ldots,z_{\tau(d)})$. The following proposition is very classical :

Proposition 1 *Let $\boldsymbol{z} \in \mathbb{C}^d$, then $\boldsymbol{z}$ satisfies the following system :*

$$\Sigma(\boldsymbol{z}) = 0 \tag{2}$$

if and only if there exists $\tau \in \mathcal{S}_d$ such that $\boldsymbol{z}^\tau = (\zeta_1,\ldots,\zeta_d)$. In other words, $\boldsymbol{z}$ is formed with the roots of f.

Th idea of the Weierstrass's method is to apply the Newton's method to Σ in order to find the solution of the system (2). But we will not do it directly since the inversion of the jacobian matrix of Σ can leads to intractable computations. We will rewrite the problem in a different way in order to get an efficient method.

Let $\boldsymbol{z} \in \mathbb{C}^d$, we denote $f_{\boldsymbol{z}}(x) = \prod_{i=1}^{d}(x-z_i)$ and $F(\boldsymbol{z},x) = f_{\boldsymbol{z}}(x) - f(x)$. Clearly $F(\boldsymbol{z},x) = 0 \in \mathbb{C}[x]$ is an equivalent system of the system (2). This is the system we will use to construct the iteration function of the Weierstrass's method. Let us denote $\mathbb{C}[x]_{d-1}$ the $\mathbb{C}$-vector space of polynomial of degree less or equal to $d-1$. It is easy to check that since $f_{\boldsymbol{z}}(x)$ and $f(x)$ are monic then $F(\boldsymbol{z},x) \in \mathbb{C}[x]_{d-1}$. We are interested in two bases of this vector space. The first one is the monomial basis $(1,x,\ldots,x^{d-1})$. We have already seem $\Sigma(\boldsymbol{z})$ is the vector of the coordinate of $F(\boldsymbol{z},x)$ in the monomial basis. If $\boldsymbol{z}$ has all is coordinates distinct, then an other interesting basis of $\mathbb{C}[x]_{d-1}$ is the Lagrange basis $(L_{z_i}(x))_{i\in\{1,\ldots d\}}$, associated with the coordinate of $\boldsymbol{z}$.

We now assume that $\boldsymbol{z}$ has all his coordinates distinct. To compute the iteration point of the Newton's method applied to F at $\boldsymbol{z}$ (x is not consider as a variable here), we have to compute the vector $\boldsymbol{u} \in \mathbb{C}^d$ such that :

$$\boldsymbol{u} = DF(.,x)(\boldsymbol{z})^{-1}F(\boldsymbol{z},x).$$

It is to say that $\boldsymbol{u}$ must satisfy the following equality :

$$DF(.,x)(\boldsymbol{z})\boldsymbol{u} = F(\boldsymbol{z},x). \tag{3}$$

We will now express the terms of that equality in the Lagrange's basis. To begin, by the Lagrange's interpolation formula we have :

$$F(\boldsymbol{z},x) = \sum_{i=1}^{d} F(\boldsymbol{z},z_i)L_{z_i}(x) = -\sum_{i=1}^{d} f(z_i)L_{z_i}(x). \tag{4}$$

This is by definition of $F(\boldsymbol{z},x)$. In other hand, we have :

$$\frac{\partial}{\partial z_j}F(\boldsymbol{z},x) = \frac{\partial}{\partial z_j}f_{\boldsymbol{z}}(x) = \frac{\partial}{\partial z_j}\prod_{k=1}^{d}(x-z_k),$$

and finally we obtain :

$$\frac{\partial}{\partial z_j}F(\boldsymbol{z},x) = -\prod_{k\neq j}(x-z_k). \tag{5}$$

By expressing the right term of the equality (5) in the Lagrange's basis (this possible because $\frac{\partial}{\partial z_i}F(\boldsymbol{z},x) \in \mathbb{C}[x]_{d-1}$), we have :

$$\frac{\partial}{\partial z_j}F(\boldsymbol{z},x) = -\prod_{k\neq j}(z_j-z_k)L_{z_i}(x) \tag{6}$$

Form equation (6), we deduce that :

$$DF(\boldsymbol{z},x)\boldsymbol{u} = \sum_{i=1}^{d} -\prod_{k\neq i}(x-z_k)u_i = \sum_{i=1}^{d} -(\prod_{k\neq i}(z_i-z_k)u_i)L_{z_i}(x) \tag{7}$$

By substitution of the terms of the equation (3) with the right terms of the equations (4) and (7), we deduce that :

$$\sum_{i=1}^{d} -(\prod_{k\neq i}(z_i-z_k)u_i)L_{z_i}(x) = -\sum_{i=1}^{d} f(z_i)L_{z_i}(x). \tag{8}$$

By identification we finally obtain :

$$u_i = \frac{f(z_i)}{\prod_{k\neq i}(z_i-z_k)} \tag{9}$$

We denote by $I_i^W(\boldsymbol{z}) = z_i - \frac{f(z_i)}{\prod\limits_{k\neq i}(z_i - z_k)}$ and by $I^W(\boldsymbol{z}) = (I_1^W(\boldsymbol{z}), \ldots, I_d^W(\boldsymbol{z}))$. The map I^W is called the Weierstrass's iteration function. Since apply this iteration function is equivalent to apply thee Newton's iteration to Σ, we have the following theorem :

Théorème 1 *Let $\boldsymbol{z} \in \mathbb{C}^d$ a point with all its coordinates distinct. The sequence $\boldsymbol{z}^{(i+1)} = I^W(\boldsymbol{z}^{(i)})$, with $\boldsymbol{z}^{(0)} = \boldsymbol{z}$ is a locally quadratically convergent method to a vector formed with the roots of the polynomial f.*

Remarque 1 Since at each step of the method the coordinates of the iteration point can be computed separately without inverse numerically the jacobian matrix of Σ, the Weierstrass's method as a great numerical behavior.

The local behavior of this method is known for a long time. But its global behavior is widely unknown. The study of the global behavior of this method was the initial motivation of that work. A future work will be dedicated to this topics. For our approach of this problem, we consider the geometry of this iteration which leads us to a new method for solving polynomial. All what follow is dedicated to this new method. We will give some geometrical material before to explain this new method.

3 Weierstrass's map and Weierstrass vector field

3.1 The Weierstrass's map

Définition 1 *We define the Weierstrass's map as follows :*

$$W : \left\{ \begin{array}{c} \mathbb{C}^d \longrightarrow \mathbb{C}^d \\ \boldsymbol{z} \longmapsto W(\boldsymbol{z}) = \begin{pmatrix} \sigma_1(\boldsymbol{z}) \\ \vdots \\ \sigma_d(\boldsymbol{z}) \end{pmatrix} \end{array} \right. .$$

Let $f(x) = x^d + \sum_{i=1}^d a_i x^{d-i} \in \mathbb{C}[x]$, we denote $\boldsymbol{f} = (a_1, \ldots, a_d) \in \mathbb{C}^d$ the vector representing $f(x)$. For clear reasons, we will call space of roots the source of W and space of polynomials its goal. We denote $\Delta = \{\boldsymbol{z} \in \mathbb{C}^d | \exists i \neq j \in \{1, \ldots, d\} \text{ with } \boldsymbol{z}_i = \boldsymbol{z}_j\}$ and $\boldsymbol{U} = \mathbb{C}^d \backslash \Delta$. It is known that $\mathcal{D} = W(\Delta)$ is the discriminant locus of the degree d polynomials. We denote $\boldsymbol{V} = \mathbb{C}^d \backslash \mathcal{D}$. Improperly, we still denote W the restriction $W|_{\boldsymbol{U}} : \boldsymbol{U} \to \boldsymbol{V}$. We have :

Proposition 2 *The map $W : \boldsymbol{U} \to \boldsymbol{V}$ defines a differential covering from $\boldsymbol{U}$ to $\boldsymbol{V}$ of degree $d!$.*

There is a natural action of the symmetric group $\mathcal{S}_d$ on $\boldsymbol{U}$ induced by its action on $\mathbb{C}^d$. We easily have :

Proposition 3 *Let $\boldsymbol{z}$ and $\boldsymbol{y} \in \boldsymbol{U}$, then $W(\boldsymbol{z}) = W(\boldsymbol{y})$ if and only if there exists $\tau \in \mathcal{S}_d$ such that $\boldsymbol{y} = \boldsymbol{z}^\tau$.*

The commutation of this group action with the map W implies the following theorem :

Théorème 2 *There exists an only homeomorphism $\mathcal{W} : \boldsymbol{U}/\mathcal{S}_d \to \boldsymbol{V}$.*

It is easy to see that $\boldsymbol{U}$ and $\boldsymbol{V}$ are differential varieties ant that $\mathcal{W}$ is a diffeomorphism. The differentials structures will play an important role.

3.2 The Weierstrass's field associated to a polynomial

Let $\boldsymbol{f} \in \boldsymbol{U}$ (this implies that the corresponding polynomial has only simple roots), the aim is to define a vector field on $\boldsymbol{U}$ by lifting by W a vector field defined on $\boldsymbol{V}$. In order to avoid some singularities, we will again need to make restrictions on the applications and on the vector field.

Définition 2 *We define $C_W^{\boldsymbol{f}} = \{\boldsymbol{g} \in \boldsymbol{V} | \exists t \in [1, \infty] \, \mathrm{with} (1-t)\boldsymbol{g} + t\boldsymbol{f} \in \mathcal{D}\}$. This set is the set of polynomials $\boldsymbol{g}$ such that the line segment joining $\boldsymbol{g}$ to $\boldsymbol{f}$ cut the discriminant locus. This is a semi-algebraic set of real codimension 1. This set is called the Weierstrass's cone.*

Définition 3 *We define $\mathcal{V}_{\boldsymbol{f}} := \boldsymbol{V} \backslash C_W^{\boldsymbol{f}}$, $C_{\boldsymbol{f}}^W = W^{-1}(C_W^{\boldsymbol{f}})$ and $\mathcal{U}_{\boldsymbol{f}} = \boldsymbol{U} \backslash C_{\boldsymbol{f}}^W$.*

Remarque 2 The set $\mathcal{U}_{\boldsymbol{f}}$ is connected. To see this, it is useful to remark that $\mathcal{V}_{\boldsymbol{f}}$ is stared relatively to $\boldsymbol{f}$.

Again, we still denote W for the restriction $W|_{\mathcal{U}_{\boldsymbol{f}}} : \mathcal{U}_{\boldsymbol{f}} \to \mathcal{V}_{\boldsymbol{f}}$, which is also a differential covering. In the same way than in the previous sub-section, we have a natural action of $\mathcal{S}_d$ on $\mathcal{U}_{\boldsymbol{f}}$ which commutes with W. Let $\boldsymbol{g} \in \mathcal{V}_{\boldsymbol{f}}$, the vector $\overrightarrow{\boldsymbol{g}\boldsymbol{f}}$ can be view as an element of $T_g\mathcal{V}_{\boldsymbol{f}}$. This leads us to define a vector field on $\mathcal{V}_{\boldsymbol{f}}$ as follows :

Définition 4 *We define the following vector field :*

$$\Phi : \left\{ \begin{array}{c} \mathcal{V}_{\boldsymbol{f}} \to T\mathcal{V}_{\boldsymbol{f}} \\ \boldsymbol{g} \longmapsto \Phi_{\boldsymbol{f}}(\boldsymbol{g}) = (\boldsymbol{g}, \overrightarrow{\boldsymbol{g}\boldsymbol{f}}) \end{array} \right. .$$

To simplify the notations, we will often denote $\Phi_{\boldsymbol{f}}(\boldsymbol{g}) = \overrightarrow{\boldsymbol{g}\boldsymbol{f}}$. This vector field allows us to define a vector field on $\mathcal{U}_{\boldsymbol{f}}$ as the pull-back of Φ by W :

Définition 5 *We define the Weierstrass's vector field as follows :*

$$\Psi : \left\{ \begin{array}{c} \mathcal{U}_{\boldsymbol{f}} \longrightarrow T\mathcal{U}_{\boldsymbol{f}} \\ \boldsymbol{z} \longmapsto \Psi_{\boldsymbol{f}}(\boldsymbol{z}) = (\boldsymbol{z}, \mathrm{DW}(\boldsymbol{z})^{-1}\Phi_{\boldsymbol{f}}(W(\boldsymbol{z}))) \end{array} \right. .$$

Again, to simplify, we will often denote $\Psi(\boldsymbol{z}) = W^\star\Phi(\boldsymbol{z}) = \mathrm{DW}(\boldsymbol{z})^{-1}\Phi(W(\boldsymbol{z}))$. The Weiersttrass's vector field will leads us to a new method to solve an univariate polynomial. The results of the section 2 allows us to give an explicit formula for Ψ. This is the goal of the following proposition :

Proposition 4 *For all $\boldsymbol{z} \in \mathcal{U}_{\boldsymbol{f}}$, we have :*

$$\Psi(\boldsymbol{z}) = \left(\frac{-f(z_1)}{\prod\limits_{j\neq 1}(z_1 - z_j)}, \ldots, \frac{-f(z_d)}{\prod\limits_{j\neq 1}(z_d - z_j)} \right). \tag{10}$$

Démonstration Fist, remark that for all $\boldsymbol{g} \in \mathcal{V}_{\boldsymbol{f}}$, we have $\overrightarrow{\boldsymbol{g}\boldsymbol{f}} = (g_1 - a_1, \ldots, g_n - a_n)$, in a way that $\Phi(W(\boldsymbol{z})) = (\sigma_1(\boldsymbol{z}) - a_1, \ldots, \sigma_d(\boldsymbol{z}) - a_d)$. So, we have $\Psi(\boldsymbol{z}) = W^\star \Phi(\boldsymbol{z}) = \mathrm{DW}(\boldsymbol{z})^{-1}\Phi(W(\boldsymbol{z}))$, which is exactly the expression of the vector $\boldsymbol{u}$ in the equation (2). The equation (10) is directly deduced from this fact. □

3.3 Geometry of the Weierstrass's vector field

In this sub-section, we treat some geometrical aspect of the Weierstrass's vector field. Our approach will use the ordinary differential equations associated to the vector fields Φ and Ψ :

$$\dot{\gamma} = \Phi(\gamma) \tag{11}$$

$$\dot{\lambda} = \Psi(\lambda) \tag{12}$$

The geometry associated to the Weierstrass's vector field share a lot of properties with the field Φ, this explains why we begin by the study of Φ.

3.3.1 Geometry associated to Φ

We recall that the orbit of a point $\boldsymbol{g} \in \mathcal{V}_{\boldsymbol{f}}$ is the graph of the maximal solution $\gamma : I \to \mathcal{V}_{\boldsymbol{f}}$ of the differential equation (11) such that there exists $t_0 \in I$ satisfying $\gamma(t_0) = \boldsymbol{g}$. The dynamic of this vector field is particularly simple :

Proposition 5 *The orbit of $\boldsymbol{g} \in \mathcal{V}_{\boldsymbol{f}}$ for Φ is :*

- *if the semi-line of origin $\boldsymbol{f}$ passing on $\boldsymbol{g}$ does not cut the discriminant, then the orbit is the semi-line itself (without $\boldsymbol{f}$) which is the graph of the application* $\gamma : \left\{ \begin{array}{c}]-\infty, 1[\to \mathcal{V}_{\boldsymbol{f}} \\ t \mapsto (1-t)\boldsymbol{g} + t\boldsymbol{f} \end{array} \right.$ *(maximal solution of (11)),*
- *if the semi-line of origin $\boldsymbol{f}$ passing on $\boldsymbol{g}$ cut the discriminant, we denote $\boldsymbol{h}_{\boldsymbol{g}}$ this intersection and the orbit is the opened line segment of extremities $\boldsymbol{f}$ and $\boldsymbol{h}_{\boldsymbol{g}}$ which is the graph of the application* $\gamma : \left\{ \begin{array}{c}]0, 1[\to \mathcal{V}_{\boldsymbol{f}} \\ t \mapsto (1-t)h_{\boldsymbol{g}} - t\boldsymbol{f} \end{array} \right.$ *(maximal solution of (11)),*
- *the set $\{\boldsymbol{f}\}$ since $\boldsymbol{f}$ is a singularity (the only one).*

Démonstration We consider the associated Cauchy problem :

$$\left\{ \begin{array}{c} \dot{\gamma} = \Phi(\gamma), \\ \gamma(0) = \boldsymbol{g} \in \mathcal{V}_{\boldsymbol{f}}. \end{array} \right.$$

One can remark that if $\boldsymbol{g} \in \mathcal{V}_{\boldsymbol{f}}$, then $\gamma : t \in [0,1[\mapsto (1-t)\boldsymbol{g} + t\boldsymbol{f} \in \mathcal{V}_{\boldsymbol{f}}$ is a solution of this Cauchy problem, but not maximal. The rest of the proof consists in the maximal extension of the solutions if $\boldsymbol{g} \neq \boldsymbol{f}$ and to remark that $\Phi(\boldsymbol{f}) = 0$. $\square$

We have easily the following proposition :

Proposition 6 *If $\gamma : I \to \mathcal{V}_{\boldsymbol{g}}$ is a maximal solution of the equation (11), then* 1 *is on the closure of I and* $\lim_{t\to 1} \gamma(t) = \boldsymbol{f}$.

Remarque 3 We will generally use weaker results than those two proposition. In fact we will mainly use that the solution γ of (11) with the initial condition $\gamma(0) = \boldsymbol{g} \in \mathcal{V}_{\boldsymbol{f}}$ is defined on $[0,1[$ and that $\lim_{t\to 1} \gamma(t) = \boldsymbol{f}$ since it is given by $\gamma(t) = (1-t)\boldsymbol{g} + t\boldsymbol{f}$.

3.3.2 Geometry associate to Ψ

Proposition 7 *Let $\boldsymbol{z} \in \mathcal{U}_{\boldsymbol{f}}$, then if $\lambda : I(\ni 0) \to \mathcal{U}_{\boldsymbol{f}}$ is a solution of the ordinary differential equation (12) with the initial condition $\lambda(0) = \boldsymbol{z}$, then $\gamma = W \circ \lambda$ is a solution of the ordinary differential equation (11) with the initial condition $\gamma(0) = W(\boldsymbol{z})$.*

Démonstration Let $\lambda : I \to \mathcal{U}_{\boldsymbol{f}}$ be the solution of the ordinary differential equation (12) with the initial condition $\lambda(0) = \boldsymbol{z}$. Denote $\gamma = W \circ \lambda$, then by the chain rule, we have $\dot\gamma(t) = \mathrm{DW}(\lambda(t))\dot\lambda(t) = \mathrm{DW}(\lambda(t))\Psi(\lambda(t))$ which gives, by definition of Ψ, $\dot\gamma(t) = \mathrm{DW}(\lambda(t))\,\mathrm{DW}(\lambda(t))^{-1}\Phi(W(\lambda(t))) = \Phi(\gamma(t))$. The initial condition is easy to check. $\square$

Proposition 8 *Let $\boldsymbol{z} \in \mathcal{U}_{\boldsymbol{f}}$, then λ the solution of the ordinary differential equation (12) with initial condition $\lambda(0) = \boldsymbol{z}$ is defined on $[0,1[$ and $\lim_{t\to 1} \lambda(t) \in W^{-1}(\boldsymbol{f})$.*

Démonstration From proposition 16, $\gamma = W \circ \lambda$ is a solution of the ordinary differential equation (11) and so is defined on $[0,1[$ by proposition 14. Then λ is defined on $[0,1[$. Furthermore from proposition 15 we have $\lim_{t\to 1} \gamma(t) = \boldsymbol{f}$ and then $\lim_{t\to 1} \lambda(t) \in W^{-1}(\boldsymbol{f})$. $\square$

4 Numerical solver of univariate polynomial equations

In this section we are interested in the possibility to design algorithms to solve univariate polynomial equation by numerical integration of the Weierstrass's vector field.

4.1 Euler's method based algorithm

The algorithm proposed here is based on the numerical integration of ordinary differential equation associated to the Weierstrass's vector field by the means of the Euler's method. We call our method the Euler-Weierstrass method. The Euler method makes an approximation of the solution of an ordinary differential equation by a piecewise linear application on a compact interval. Because the solutions of the equation (2) are defined on $[0,1[$, we need to make the approximation on an interval of shape $I_\varepsilon = [0, 1-\varepsilon]$ where $\varepsilon \in]0,1[$ is a input parameter. A good choice for the value of this parameter will be discuss later. An other important parameter is the size of the step used for the discretization of I_ε.

Let $N \in \mathbb{N}$, we denote $\delta_\varepsilon = \frac{1-\varepsilon}{N}$. We consider the subdivision of I_ε by intervals of shape $[i\delta_\varepsilon, (i+1)\delta_\varepsilon]$, $i \in \{0, \ldots, N-1\}$. We define $\lambda_{\varepsilon,N}(x) = \lambda_{\varepsilon,N}(i\delta_\varepsilon) + (x - i\delta_\varepsilon)\Psi(\lambda_{\varepsilon,N}(i\delta_\varepsilon))$ on $[i\delta_\varepsilon, (i+1)\delta_\varepsilon]$. A classical result of numerical analysis say that the sequence $\lambda_{\varepsilon,N}$ converges uniformly to λ (λ denote the solution of (12) with initial condition $\lambda(0) = \lambda_{\varepsilon,N}(0)$) when $N \to \infty$. This leads us to consider the following algorithm :

Algorithm :

- Inputs : f, $\boldsymbol{z} \in \mathcal{U}_{\boldsymbol{f}}$, $\varepsilon \in]0,1[$ and $N \in \mathbb{N}$.
- $\lambda_0 = \boldsymbol{z}, \delta = \frac{1-\varepsilon}{N}$.
- for i from 0 to N-1 do
  $\lambda_{i+1} = \lambda_i + \delta\Psi(\lambda_i)$
- end for.
- Output : λ_N.

Because $\lambda_{\varepsilon,N}$ is an approximation of λ on I_ε, λ_N will be an approximation of $\lambda(1-\varepsilon)$, where λ is the solution o th ordinary differential equation with initial condition $\lambda(0) = \boldsymbol{z}$. So, if ε is small enough (in a way to be precised), λ_N will be an approximation of $\mathrm{llim}_{t\to 1} \lambda(t) \in W^{-1}(\boldsymbol{f})$, i.e. λ_N will be an approximation of a vector made with the roots of $\boldsymbol{f}$.

Hereafter, we will explain the links with a well known method : the modified Weierstrass method.

4.2 Links between Euler-Weierstrass method and modified Weierstrass's method

The modified Weierstrass's method is a numerical path following method using the classical Weierstrass's method as a correction operator. This method proceed as follow :

For $\boldsymbol{g} \in \mathcal{U}_{\boldsymbol{f}}$, we consider $h : [0,1] \to \mathcal{V}_{\boldsymbol{f}}$ defined by $h(t) = (1-t)\boldsymbol{g} + t\boldsymbol{f}$. We subdivide $[0,1]$ by the intervals $[i\delta, (i+1)\delta]$ where $\delta = \frac{1}{N}$, $i \in \{0, \ldots, N-1\}$ and $N \in \mathbb{N}$. We assume to know the roots of $\boldsymbol{g}$ (in reality, we choose a vector $\boldsymbol{z} \in \mathcal{U}_{\boldsymbol{f}}$ and we take $\boldsymbol{g} = W(\boldsymbol{z})$). We take the vector $\boldsymbol{z}^{(0)} = \boldsymbol{z}$ as an input of the classical Weierstrass's method to compute an approximation $\boldsymbol{z}^{(1)}$ of a vector made with the roots of $h(\delta)$. We use $\boldsymbol{z}^{(1)}$ as an input of the classical Weierstrass's method to compute an approximation $\boldsymbol{z}^{(2)}$ of a vector made with the coefficient of $h(2\delta)$, and so on. At the end of the process, we have a vector $\boldsymbol{z}^{(N)}$ which i an approximation of a vector made with the roots of $h(N\delta) = h(1) = \boldsymbol{f}$.

This method can be view as a kind of Euler's method, but it does not be applied to Ψ but to Φ and the iteration points are pulled-back on $\mathcal{U}_{\boldsymbol{f}}$ by the used of the Weierstrass's method. Our method consist to apply the Euler's method directly to Ψ.

4.3 Convergence of the method

We consider λ a solution on $[0, 1[$ of :

$$\left\{ \begin{array}{c} \dot{\lambda} = \Phi(\lambda) \\ \lambda(0) = \boldsymbol{z}^{(0)} \in \mathcal{U}. \end{array} \right.$$

We know that $\lim_{t \to 1} \lambda(t) \in W^{-1}(\boldsymbol{f})$. For $\varepsilon \in]0, 1[$, we define $I_\varepsilon = [0, 1-\varepsilon]$. The aim here is to study the global error of the Euler's method applied to Φ on I_ε.

Let $N \in \mathbb{N}\backslash\{0\}$ and $\delta = \frac{1-\varepsilon}{N}$, we consider the subdivision $t_0 = 0 < t_1 = \delta < \cdots < t_i = i\delta < \cdots < t_N = 1 - \varepsilon$ of I_ε. We recall that the consistency error of the Euler's method applied to Φ is $e_n = \lambda(t_{n+1}) - (\lambda(t_n) + \delta\Phi(\lambda(t_n)))$. We have the following proposition :

Proposition 9 *We have $e_n = \frac{\delta^2}{2}\ddot{\lambda}(t_n) + \boldsymbol{o}(\delta^2)$ and consequently, the method is consistent.*

Démonstration The Taylor expansion of $\lambda(t_{n+1})$ is $\lambda(t_{n+1}) = \lambda(t_n) + \delta\dot{\lambda}(t_n) + \frac{\delta^2}{2}\ddot{\lambda}(t_n) + \boldsymbol{o}(\delta^2)$. Because $\dot{\lambda}(t_n) = \Phi(\lambda(t_n))$, we have $e_n = \frac{\delta^2}{2}\ddot{\lambda}(t_n) + \boldsymbol{o}(\delta^2)$. It is easy to remark that when $N \to +\infty$, $\delta \to 0$ and so $e_n \to 0$ and the method is consistent. □

In the following, we will need the following lemma :

Lemme 1 *(mutivariate discrete Gronwall's lemma)*

Let (θ_n) and (ε_n) be sequance in $\mathbb{C}^d$ and $\mu \in [0, +\infty[$ such that :

$$\|\theta_{n+1}\| \leqslant (1+\mu)\|\theta_n\| + \|\varepsilon_n\|.$$

Then we have $\|\theta_n\| \leqslant \boldsymbol{e}^{n\mu}\|\theta_0\| + \sum_{0 \leqslant i \leqslant n-1} \boldsymbol{e}^{(n-i-1)\mu}\|\varepsilon_i\|.$

Démonstration By induction on n. Trivially, we have $\|\theta_0\| \leqslant \|\theta_0\|$. Assume that the property is true for rank n. Then by hypothesis, we have :

$$\|\theta_{n+1}\| \leqslant (1+\mu)\|\theta_n\| + \|\varepsilon_n\|$$

but $(1+\mu) \leqslant \boldsymbol{e}^\mu$ and by induction we have :

$$\|\theta_{n+1}\| \leqslant \boldsymbol{e}^\mu(\boldsymbol{e}^{n\mu}\|\theta_0\| + \sum_{0 \leqslant i \leqslant n-1} \boldsymbol{e}^{(n-i-1)\mu}\|\varepsilon_i\|) + \|\varepsilon_n\|,$$

but $\|\varepsilon_i\| \leqslant \boldsymbol{e}^{(n-1)\mu}\|\varepsilon_i\|$, then :

$$\|\theta_{n+1}\| \leqslant \boldsymbol{e}^{(n+1)\mu}\|\theta_0\| + \sum_{0\leqslant i\leqslant n-1} \boldsymbol{e}^{(n-i)\mu}\|\varepsilon_i\| + \boldsymbol{e}^{(n-1)\mu}\|\varepsilon_i\|,$$

and then we have :

$$\|\theta_{n+1}\| \leqslant \boldsymbol{e}^{n\mu}\|\theta_0\| + \sum_{0\leqslant i\leqslant n} \boldsymbol{e}^{(n-i-1)\mu}\|\varepsilon_i\|.$$

□

This lemma will allow us to study the stability of the Euler's method applied to Φ and to determine the nature of the global error of this method on I_ε.

Let λ_0 and $\tilde{\lambda}_0 \in \mathcal{U}$. We consider :

- λ and $\tilde{\lambda}$ solutions of $\dot{\lambda} = \Phi(\lambda)$ with initial conditions respectively $\lambda(0) = \lambda_0$ and $\tilde{\lambda}(0) = \tilde{\lambda}_0$,
- (λ_n) and $(\tilde{\lambda}_n)$ sequences defined by $\lambda_{n+1} = \lambda_n + \delta\Phi(\lambda_n)$ and $\tilde{\lambda}_{n+1} = \tilde{\lambda}_n + \delta\Phi(\tilde{\lambda}_n) + \varepsilon_n$ respectively.

If the sequence ε_n is such that $\|\varepsilon_n\|$ is small enough $\forall n \in \{1,\ldots,N\}$, then there exists a compact $K \subset \mathcal{U}$ such that $\lambda(t)$ and $\tilde{\lambda}(t) \in K$ for all $t \in [0,1[$ and λ_n and $\tilde{\lambda}_n \in K$ for all $n \in \{0,\ldots,N\}$. Because Φ is $\mathcal{C}^\infty$ on K it is lipschitzian of Lipschitz constant $\Lambda = \max\limits_{\boldsymbol{z}\in K}\{D\Phi(\boldsymbol{z})\}$. We have the following theorem :

Théorème 3 *With the above notations, we have :*

$$\|\lambda_n - \tilde{\lambda}_n\| \leqslant \boldsymbol{e}^{\Lambda(1-\varepsilon)}(\|\lambda_0 - \tilde{\lambda}_0\| + \sum_{0\leqslant n\leqslant N}\|\varepsilon_n\|.$$

Démonstration It is suffisent to remark that $n\delta \leqslant (1-\varepsilon)$ and that $(n-i-1)\delta \leqslant (1-\varepsilon)$ for all $n \in \{1,\ldots,N\}$ and $i \in \{1,\ldots,n-1\}$ and to apply the Gronwall lemma to the sequences $\theta_n = \lambda_n - \tilde{\lambda}_n$ and ε_n with $\mu = \Lambda\delta$. □

By choosing $\tilde{\lambda}_n = \tilde{\lambda}(t_n) + e_n$ with $\tilde{\lambda}_0 = \lambda_0$ (and by definition of e_n), we have :

$$\|\lambda(1-\varepsilon) - \lambda_N\| \leqslant \boldsymbol{e}^{\Lambda(1-\varepsilon)} \sum_{0\leqslant n\leqslant N}\|e_n\|.$$

Because $e_n = \ddot{\lambda}(t_n)\delta^2 + \boldsymbol{o}(\delta^2)$ on $[0, 1-\varepsilon]$, there exists $\gamma \in [0,+\infty[$ such that $\|e_n\| \leqslant \gamma\delta^2$. This leads us to the following proposition :

Proposition 10 *There exists* $C \in [0,+\infty[$ *such that* $\|\lambda(1-\varepsilon) - \lambda_N\| \leqslant C/N$.

Démonstration We have $\|e_n\| \leqslant \gamma\delta^2$, then $\sum\limits_{0\leqslant n\leqslant N}\|e_n\| \leqslant N\gamma\delta^2$. But $\delta^2 = \frac{(1-\varepsilon)^2}{N^2}$, then $\sum\limits_{0\leqslant n\leqslant N}\|e_n\| \leqslant \frac{\gamma(1-\varepsilon)^2}{N}$ and $\|\lambda(1-\varepsilon) - \lambda_N\| \leqslant \frac{\boldsymbol{e}^{\Lambda(1-\varepsilon)}\gamma(1-\varepsilon)^2}{N}$. □

This result just show that for N suffisently large, the global error of the Euler's method applied to Φ on I_ε behave as $1/N$. But, at this time, we are not able to to give a uniform bound for C.

5 Conclusions and perspectives

In this paper we propose a new approach to univariate polynomial solving. The method proposed consist in :

- construct a vector field which admits as only singularities (attractive) vector made with the set of roots of the polynomial we want to solve,
- integrate this vector field and reciver an approximation of one of the singularity of this vector field, it is to say an approximation of all the roots of the polynomial we want to soolve.

We propose a effective solver based on the Euler method to integrate the Weierstrass vector field and we proved the convergence of that method. This new appoach open severale perspectives :

- we can use other numerical method than Euler one,
- we can also use based on symbolic-numeric method in order to integrate the Weierstrass vector field as the one proposed by Van der Hoeven [6],
- the Weierstrass method had been generalized in multivariate context [9, 11, 10] together with the generalization of the Weierstrass map, a work in progress try to generalize this approach to this contex.

Ackwolegdement : The author want to thank Bernard Mourrain and Joris Van der Hoeven for the time they spent to discuss about this approach and Carlo Traverso for is support.

Références

[1] Ruben Airapetyan, Alexander Ramm, and Alexandra Smirnova. Continuous analog of the gauss-newton method. *Mathematical Models and Methods in Applied Sciences*, 9(3) :463–474, 1999.

[2] Ruben Airapetyan, Alexander Ramm, and Alexandra Smirnova. Continuous methods for solving nonlinear ill-posed problems. *Operator theory and its applications*, pages 111–137, 2000.

[3] Eugene L. Allgower and Kurt Georg. *Numerical Path Following*. Springer, 1990.

[4] Lenore Blum, Felipe Crucker, Michael Shub, and Steve Smale. *Complexity and Real Computation*. Springer, 1998.

[5] Peter Doyle and Curtis McMullen. Solving the quintic by iteration. *Acta Mathematica*, 163(3-4) :151–180, 1989.

[6] Van der Hoeven Joris. Effective analytic function. In *MEGA 2003*, 2003.

[7] Curtis McMullen. Families of rational maps and iterative root-finding algorithm. *Ann. of Math.*, 125(3) :467–493, 1987.

[8] V.Y. Pan. Solving a polynomial equation : Some history and recent progress. *SIAM Rev.*, 39(2) :187–220, 1997.

[9] Olivier Ruatta. A multivariate weierstrass iterative rootfinder. In *issac*, London, Ontario, 2001. ACM press.

[10] Olivier Ruatta. *Dualité algébrique, structures et applications.* PhD thesis, Université de d'Aix-Marseille II, 2002.

[11] Olivier Ruatta and Bernard Mourrain. Relation between roots and coefficients, interpolation and application to system solving. *Journal of Symbolic Computation*, (33) :679–699, 2002.