Chapter 1

Ordinary Differential Equations

In mathematics, an *ordinary differential equation* is a *differential equation* containing one or more functions of one independent variable and its derivatives (of any order).[1] This mathematical tool is widely used to describe dynamical systems (physical, social, economical, biological, ...). The first mathematicians to study and apply ordinary differential equations include important names like Newton, Euler, Leibniz, the Bernoulli family, Riccati, Clairaut and d'Alembert. An exmple of ODE is probably on of the most known formulas in the world, that is trivially the Newton's second law of motion:

$$\mathcal{F}(x) = m\ddot{x} \tag{1.1}$$

1.1 An introduction to Initial Value Problems

Before getting into any numerical solution scheme let us briefly recall the essential formal definitions and properties of ordinary differential equations and initial value problems.

Definition 1 (Ordinary differential equation)**.** An *ordinary differential equation* (ODE) is an equation for a function $y(t)$, defined on an interval $I \subset \mathbb{R}$ and with values in the real or complex numbers or in the space $\mathbb{R}^d$ ($\mathbb{C}^d$), of the form:

$$F\left(t, y(t), y'(t), y''(t) \ldots, y^{(n)}\right) = 0 \tag{1.2}$$

Here F represents an arbitrary function of its arguments. The *order* n of a differential equation is the highest derivative which occurs. If the dimension d of the value range of y is higher than one, we talk about *systems of differential equations*.

Definition 2. An *explicit* differential equation of first order is a equation of the form:

$$y'(t) = f\left(t, y(t)\right) \tag{1.3}$$

or shortly:

$$y' = f\left(t, y\right) \tag{1.4}$$

A differential equation of order n is called explicit, if it is of the form:

$$y^{(n)}(t) = F\left(t, y(t), y'(t), y''(t) \ldots, y^{(n-1)}(t)\right) \tag{1.5}$$

Definition 3. A differential equation of the form (1.3) is called *autonomous*, if the right hand side f is not explicitly dependent on t, namely:

$$y'(t) = f\left(y(t)\right) \tag{1.6}$$

[1] Notice that the term *ordinary* is used in contrast with the term *partial* so to specify that the differential equations has only *one* independent variable.

Lemma 0.1. *Every differential equation of higher order can be written as a system of first-order differential equations. If the equation is explicit, then the system is explicit.*

Proof. By the introduction of additional variables $y_0(t) = y(t)$, $y_1(t) = y'(t)$ to $y_{n-1}(t) = y^{(n-1)}(t)$, each differential equation of order n can be transformed into a system of n differential equations of first order. This system has the form:

$$\left\{\begin{array}{c} y_0'(t) - y_1(t) \\ y_1'(t) - y_2(t) \\ \vdots \\ y_{n-2}'(t) - y^{(n-2)}(t) \\ F\left(t, y_0(t), y_1(t), \ldots, y_{n-1}(t), y_{n-1}'(t)\right) \end{array}\right\} = \left\{\begin{array}{c} 0 \\ 0 \\ 0 \\ 0 \\ 0 \end{array}\right\} \tag{1.7}$$

In the case of an explicit equation, the system has the form:

$$\left\{\begin{array}{c} y_0'(t) \\ y_1'(t) \\ \vdots \\ y_{n-2}'(t) \\ y_{n-1}'(t) \end{array}\right\} = \left\{\begin{array}{c} y_1(t) \\ y_2(t) \\ \vdots \\ y_{n-1}(t) \\ F\left(t, y_0(t), y_1(t), \ldots, y_{n-1}(t)\right) \end{array}\right\} \tag{1.8}$$

□

We will see that most of times the ODEs do not come alone. Almost always they are coupled with an *initial condition*, thus forming the so-called *initial value problem*. The initial condition give information about the value of function y at a given point t_0 in the domain

Definition 4 (Initial value problem)**.** Given a point $(t_0, u_0) \in \mathbb{R} \times \mathbb{R}^d$. Furthermore, let the function $f(t, u)$ with values in $\mathbb{R}^d$ be defined in a neighborhood $I \times U \subset \mathbb{R} \times \mathbb{R}^d$ of the initial value. Then an *initial value problem* (IVP) is defined as follows: find a function $y(t)$, such that:

$$\begin{cases} y'(t) = f(t, y(t)) \\ y(t_0) = y_0 \end{cases} \tag{1.9}$$

Definition 5 (Local solution)**.** We call a continuously differentiable function $y(t)$ with $u(t_0) = y_0$ a *local solution* of the IVP, if there exists a neighborhood J of the point in time t_0 in which $y(t)$ and $f(t, y(t))$ are defined and if the equation $y(t_0) = y_0$ holds for all $t \in J$.

Definition 6 (Linear ODE)**.** A differential equation is said to be *linear* if F can be written as a linear combination of the derivatives of y:

$$y^{(n)}(t) = \sum_{i=1}^{n-1} a_i(t) y^{(i)} + r(t) \tag{1.10}$$

with $a_i(t)$ and $r(t)$ continuous functions in t. If $r(t) = 0$ the we call the linear differential equation *homogeneous* otherwise we call it *inhomogeneous*.

1.1.1 Well-posedness of the Initial Value Problems

Definition 7. A mathematical problem is called well-posed if the following *Hadamard conditions* are satisfied:

- A solution exists.
- The solution is unique.
- The solution is continuously dependent on the data.

The third condition is often dropped and substituted with the so-called Lipschitz continuity, which is more a quantitative condition.

Definition 8. The function $f(t, y)$ satisfies on its domain $D = I \times \Omega \subset \mathbb{R} \times \mathbb{R}^d$ an uniformly continuous Lipschitz condition if it is Lipschitz continuous with regard to y. In other words it exists a positive constant L, such that:

$$\forall t \in I; x, y \in \Omega : |f(t, x) - f(t, y)| \leq L|x - y| \tag{1.11}$$

It satisfies a local Lipschitz condition if the same holds true for all compact subsets of D.

As we will see the *Peano's existence theorem* proofs that if f is a continuous map than there exists at least a solution to the ODE. It must be pointed out that the Peano theorem does not proof that the solution is unique.

Theorem 1 (Peano's existence theorem)**.** *Let the function $f(t, y)$ be continuous on the closed set*

$$\overline{D} = \left\{(t, u) \in \mathbb{R} \times \mathbb{R}^d \;\middle|\; |t - t_0| \leq \alpha, |y - y_0| \leq \beta\right\} \tag{1.12}$$

where $\alpha, \beta > 0$. Then there exists a solution $y \in \mathbb{C}^1(I)$ on the interval $I = [t_0 - T, t_0 + T]$ with:

$$T = \min\left(\alpha, \frac{\beta}{M}\right), M = \max_{(t,u)\in\overline{D}} |f(t, u)| \tag{1.13}$$

Theorem 2 (Peano's continuation theorem)**.** *Let the assumptions of Peano's existence theorem hold. Then, the solution can be extended to an interval $I_m = [t_-, t_+]$ such that the points $(t_-, u(t_-))$ and $(t_+, u(t_+))$ are on the boundary of D. Neither the values of t, nor of $y(t)$ need to be bounded as long as f remains bounded.*

To proof the uniqueness of solution we employ the aforementioned *Lipschitz continuity* together with the *Grönwall's lemma.*

Lemma 2.2 (Grönwall's theorem)**.** *Let be $w(t)$, $a(t)$ and $b(t)$ be nonnegative, integrable functions, such that $a(t)w(t)$ is integrable. Furthermore, let $b(t)$ be monotonically nondecreasing and let $w(t)$ satisfy the integral inequality:*

$$w(t) \leq b(t) + \int_{t_0}^{t} a(s)w(s)ds, \qquad t \geq t_0 \tag{1.14}$$

Then, for almost all $t \geq t_0$ there holds:

$$w(t) \leq b(t) \exp\left(\int_{t_0}^{t} a(s)ds\right) \tag{1.15}$$

The existence and uniqueness of solution can be now proofed thanks to the *Picard-Lindelöf theorem* (also called *Picard's existence theorem*, *Cauchy-Lipschitz theorem*, or *existence and uniqueness theorem*):

Theorem 3 (Picard-Lindelöf theorem)**.** *Let $f(t, y)$ be continuous on a cylinder $D = (t, y) \in \mathbb{R} \times \mathbb{R}^d \;\middle|\; |t - t_0| \leq a, |y - y_0| \leq b$. Let f be bounded such that there is a constant $M = \max_D |f|$ and satisfy the Lipschitz condition with constant L on D. Then the IVP:*

$$\begin{cases} y'(t) = f(t, y(t)) \\ y(t_0) = y_0 \end{cases} \tag{1.16}$$

is uniquely solvable on the interval $I = [t_0 - T, t_0 + T]$ where $T = \min\{a, b/M\}$.

1.2 Numerical Solutions of Initial Value Problems

In most cases an analytical solution to IVPs cannot be found or it is simply impractical (complex integrals appear in the solution). Thus, a set of numerical schemes for solving IVPs are now presented. It must be pointed out that all the schemes can be naturally extended to systems of differential equations. Moreover, since higher order differential equations can be rewritten as a system of first order differential equations, we will only concentrate on numerical methods for these last ones.

Definition 9 (Time step)**.** On a time interval $I = [t_0, t_0 + T]$, we define a partitioning in n subintervals, also known as *time steps*. The time steps $I_k = [t_{k-1}, t_k]$ have the step size $h_k = t_k - t_{k-1}$. A partitioning in n time steps implies $t_n = T$. The term k-th time step is used for both the interval I_k and for the point in time t_k, but it should always be clear through context which one is meant. Very often, we will consider evenly spaced time steps, in which case we denote the step size by h and $h_k = h$ for all k.

Numerical methods can be subdivided into two main categories: *one-step methods* and *multi-step methods*. In the first category the step y_{k+1} is determined only by y_k, where as in the second category the the step y_{k+1} is determined also by $y_{k-1}, \ldots, y_{k-n}$.

Definition 10 (Explicit method (forward integration))**.** An *explicit method* is a method which, given y_0 at t_0 computes a sequence of approximations $y_1, \ldots, y_n$ to the solution of an IVP in the time steps $t_1, \ldots, t_n$ using an update formula of the form:

$$y_k = y_{k-1} + h_k F(t_{k-1}, y_{k-1}, \ldots, y_{k-p}) \tag{1.17}$$

where F is called *increment function* and p is the order is the numerical multi-step method. If $p = 1$ we call the method as one-step method.

Definition 11 (Implicit method (backward integration))**.** An *implicit method* is a method which, given y_0 at t_0 computes a sequence of approximations $y_1, \ldots, y_n$ to the solution of an IVP in the time steps $t_1, \ldots, t_n$ using an update formula of the form:

$$y_k = y_{k-1} + h_k F(t_k, y_k, \ldots, y_{k-p}) \tag{1.18}$$

Notice that the increment function F depends on y_k and the previous equation must be solved for y_k.

1.2.1 Euler Method

The first and simplest on-step numerical method is the Euler method. There are many ways of deriving this method. To derive this method let us consider the following IVP:

$$\begin{cases} y'(t) = f(t, y) \\ y(t_0) = y_0 \end{cases} \tag{1.19}$$

The truncated Taylor series of the solution $y(t)$ centered in t_0, that is:

$$y(t_0 + h) = y_0 + hy_1 + o(h^2) \tag{1.20}$$

where $y_1(t = t_0) = y'(t = t_0)$ and since $y(t)$ is considered to be the solution is the IVPs $y_1(t = t_0) = f(t_0, y_0)$. We get an approximation of $y(t_0 + h)$, namely:

$$\tilde{y}(t_0 + h) = y_0 + hy_1 \tag{1.21}$$

Notice that the difference between $\tilde{y}(t_0 + h)$ and $y(t_0 + h)$ is proportional to h^2.

If we consider a generic interval $I = [t_0, t_0 + T]$, we can generalize the equation (1.21) to the k-th step and repeat it on each k-th subinterval $I_k = [t_{k-1}, t_k]$. Thus, in general the k-th step of the *explicit Euler method* can be written as:

$$y(t_k) = y_{k-1} + hf(t_{k-1}, y_{k-1}), \quad k = 1, \ldots, T/h \tag{1.22}$$

whereas the the k-th step of the *implicit Euler method* can be written as:

$$y(t_k) = y_{k-1} + hf(t_k, y_k), \quad k = 1, \ldots, T/h \tag{1.23}$$

It can be prrofed that both explicit Euler method and implicit Euler method converges to the exact solutionas $h \to 0$ and the error at any time $t \in I = [t_0, t_0 + T]$ can be bounded by Ch, where C is a positive constant. Since C is proportional to e^{LT}, where the Lipschitz constant L of f may be very large. Such problem, which are commonly referred to be *stiff*, are characterized by a large changes in at very different time scales and high sensitivity to changes in the initial condition. If we apply the Euler method to a stiff problems it turns out that the explicit method is not able to properly find the solution due to its *stability*. On the other hand implicit Euler method works large number of applications with a rate of converge of $o(h)$.

1.2.2 Runge-Kutta Methods

Even if the Euler method works fine for most of applications, if the dimension d is large, the end time T is large, the error tolerance ε is small or more importantly the ODE is stiff we might want to improve the solution method. To pursue these improvements we can use the so-called *Runge-Kutta methods*.

The generic *explicit Runge-Kutta method* is a one-step method with the representation:

$$\begin{aligned} g_i &= y_k + h \sum_{j=1}^{i-1} a_{ij} k_j \\ k_i &= f(hc_i, g_i) \\ g_{k+1} &= y_k + h \sum_{i=1}^{s} b_i k_i \end{aligned} \tag{1.24}$$

where $i = 1, \ldots, s$. In this method the values hc_i are the quadrature points on the interval $[0, h]$. The values k_i are approximations to function values of the integrand in these points and the values gi constitute approximations to the solution $y(hc_i)$ in the quadrature points. This method uses s intermediate values and is thus called an s-stage method.

Definition 12 (Butcher tableau)**.** It is customary to write Runge-Kutta methods in the form of a *Butcher tableau*, containing only the coefficients of equation **??** in the following matrix form:

$$\begin{array}{c|ccccc} 0 & & & & & \\ c_2 & a_{21} & & & & \\ c_3 & a_{31} & a_{32} & & & \\ \vdots & \vdots & \vdots & \ddots & & \\ c_s & a_{s1} & a_{s2} & \ldots & a_{s,s-1} & \\ \hline & b_1 & b_2 & \ldots & b_{s-1} & b_s \end{array} \tag{1.25}$$

Modified Euler Method The modified Euler method is a variation of the Euler method of the following form:

$$\begin{aligned} k_1 &= f(t_k, y_k) \\ k_2 &= f\left(t_k + \frac{1}{2}h, y_k + \frac{1}{2}hk_1\right) \\ y_{k+1} &= y_k + hk_2 \end{aligned} \qquad \begin{array}{c|cc} 0 & & \\ 1/2 & 1/2 & \\ \hline & 0 & 1 \end{array} \tag{1.26}$$

Heun Method The Heun method is a method of the following form:

$$\begin{aligned} k_1 &= f(t_k, y_k) \\ k_2 &= f\left(t_k + h, y_k + hk_1\right) \\ y_{k+1} &= y_k + h\left(\frac{1}{2}k_1 + \frac{1}{2}k_2\right) \end{aligned} \qquad \begin{array}{c|cc} 0 & & \\ 1 & 1 & \\ \hline & 1/2 & 1/2 \end{array} \tag{1.27}$$

Three stage Runge-Kutta Method The three stage Runge-Kutta is a method of the following form:

$$\begin{aligned} k_1 &= f(t_k, y_k) \\ k_2 &= f\left(t_k + \frac{1}{2}h, y_k + \frac{1}{2}hk_1\right) \\ k_3 &= f\left(t_k + h, y_k - hk_1 + 2hk_2\right) \\ y_{k+1} &= y_k + h\left(\frac{1}{6}k_1 + \frac{4}{6}k_2 + \frac{1}{6}k_3\right) \end{aligned} \qquad \begin{array}{c|ccc} 0 & & & \\ 1/2 & 1/2 & & \\ 1 & -1 & 2 & \\ \hline & 1/6 & 4/6 & 1/6 \end{array} \tag{1.28}$$

Four stage (classical) Runge-Kutta Method The calssical Runge-Kutta is a method of the following form:

$$\begin{aligned} k_1 &= f(t_k, y_k) \\ k_2 &= f\left(t_k + \frac{1}{2}h, y_k + \frac{1}{2}hk_1\right) \\ k_3 &= f\left(t_k + \frac{1}{2}h, y_k + \frac{1}{2}hk_2\right) \\ k_4 &= f\left(t_k + h, y_k + hk_3\right) \\ y_{k+1} &= y_k + h\left(\frac{1}{6}k_1 + \frac{2}{6}k_2 + \frac{2}{6}k_3 + \frac{1}{6}k_4\right) \end{aligned} \qquad \begin{array}{c|cccc} 0 & & & & \\ 1/2 & 1/2 & & & \\ 1/2 & 0 & 1/2 & & \\ 1 & 0 & 0 & 1 & \\ \hline & 1/6 & 2/6 & 2/6 & 1/6 \end{array} \tag{1.29}$$

All the presented method are *explicit Runge-Kutta methods*. If we deal with *stiff* equations the *implicit Runge-Kutta method* should be preferred.

The generic *implicit Runge-Kutta method* is a one-step method with the representation:

$$\begin{aligned} g_i &= y_k + h\sum_{j=1}^{s} a_{ij}k_j \\ k_i &= f(hc_i, g_i) \\ g_{k+1} &= y_k + h\sum_{i=1}^{s} b_i k_i \end{aligned} \tag{1.30}$$

The butcher tableau for (1.30) will be then of the form:

$$\begin{array}{c|cccc} c_1 & a_{11} & a_{12} & \dots & a_{1s} \\ c_2 & a_{21} & a_{22} & \dots & a_{2s} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ c_s & a_{s1} & a_{s2} & \dots & a_{ss} \\ \hline & b_1 & b_2 & \dots & b_s s \end{array} \tag{1.31}$$

Appendix A

A.1 Explicit matrix inverse

Here we provide the explicit expression for the inverse of matrix in **??**,

$$\begin{bmatrix} \boldsymbol{I} & 0 & 0 \\ 0 & \boldsymbol{M}(\boldsymbol{q}) & -\frac{\partial \boldsymbol{\phi}}{\partial \boldsymbol{q}}^{\intercal} \\ 0 & -\frac{\partial \boldsymbol{\phi}}{\partial \boldsymbol{q}} & 0 \end{bmatrix}^{-1} = \begin{bmatrix} \boldsymbol{I} & 0 & 0 \\ 0 & \boldsymbol{X}_{11} & \boldsymbol{X}_{12} \\ 0 & \boldsymbol{X}_{21} & \boldsymbol{X}_{22} \end{bmatrix},$$

where the quantities $\boldsymbol{X}_{11} \in \mathbb{R}^{n \times n}$, $\boldsymbol{X}_{12} \in \mathbb{R}^{n \times m}$, $\boldsymbol{X}_{21} \in \mathbb{R}^{m \times n}$, $\boldsymbol{X}_{22} \in \mathbb{R}^{m \times m}$ are defined as follows,

$$\begin{aligned}
\boldsymbol{X}_{11} &= \boldsymbol{M}(\boldsymbol{q})^{-1} - \boldsymbol{M}(\boldsymbol{q})^{-1} \frac{\partial \boldsymbol{\phi}}{\partial \boldsymbol{q}}^{\intercal} \left(\frac{\partial \boldsymbol{\phi}}{\partial \boldsymbol{q}} \boldsymbol{M}(\boldsymbol{q})^{-1} \frac{\partial \boldsymbol{\phi}}{\partial \boldsymbol{q}}^{\intercal} \right)^{-1} \frac{\partial \boldsymbol{\phi}}{\partial \boldsymbol{q}} \boldsymbol{M}(\boldsymbol{q})^{-1}, \\
\boldsymbol{X}_{12} &= -\boldsymbol{M}(\boldsymbol{q})^{-1} \frac{\partial \boldsymbol{\phi}}{\partial \boldsymbol{q}}^{\intercal} \left(\frac{\partial \boldsymbol{\phi}}{\partial \boldsymbol{q}} \boldsymbol{M}(\boldsymbol{q})^{-1} \frac{\partial \boldsymbol{\phi}}{\partial \boldsymbol{q}}^{\intercal} \right)^{-1}, \\
\boldsymbol{X}_{21} &= -\left(\frac{\partial \boldsymbol{\phi}}{\partial \boldsymbol{q}} \boldsymbol{M}(\boldsymbol{q})^{-1} \frac{\partial \boldsymbol{\phi}}{\partial \boldsymbol{q}}^{\intercal} \right)^{-1} \frac{\partial \boldsymbol{\phi}}{\partial \boldsymbol{q}} \boldsymbol{M}(\boldsymbol{q})^{-1}, \\
\boldsymbol{X}_{22} &= -\left(\frac{\partial \boldsymbol{\phi}}{\partial \boldsymbol{q}} \boldsymbol{M}(\boldsymbol{q})^{-1} \frac{\partial \boldsymbol{\phi}}{\partial \boldsymbol{q}}^{\intercal} \right)^{-1}.
\end{aligned}$$

Notice that as expected $\boldsymbol{X}_{12} = \boldsymbol{X}_{21}^{\intercal}$, moreover in the inversion formula appears the expression,

$$\left(\frac{\partial \boldsymbol{\phi}}{\partial \boldsymbol{q}} \boldsymbol{M}(\boldsymbol{q})^{-1} \frac{\partial \boldsymbol{\phi}}{\partial \boldsymbol{q}}^{\intercal} \right)^{-1}$$

which is exactly the one in **??** for which we must guarantee the invertibility. Notice also that in the case of a single constraint, i.e., $\phi(\boldsymbol{q}) \in \mathbb{R}$ the following expression can be simplified as follows,

$$\boldsymbol{M}(\boldsymbol{q})^{-1} \frac{\partial \boldsymbol{\phi}}{\partial \boldsymbol{q}}^{\intercal} \left(\frac{\partial \boldsymbol{\phi}}{\partial \boldsymbol{q}} \boldsymbol{M}(\boldsymbol{q})^{-1} \frac{\partial \boldsymbol{\phi}}{\partial \boldsymbol{q}}^{\intercal} \right)^{-1} \frac{\partial \boldsymbol{\phi}}{\partial \boldsymbol{q}} \boldsymbol{M}(\boldsymbol{q})^{-1} = \frac{\boldsymbol{M}(\boldsymbol{q})^{-1} \frac{\partial \boldsymbol{\phi}}{\partial \boldsymbol{q}}^{\intercal} \frac{\partial \boldsymbol{\phi}}{\partial \boldsymbol{q}} \boldsymbol{M}(\boldsymbol{q})^{-1}}{\frac{\partial \boldsymbol{\phi}}{\partial \boldsymbol{q}} \boldsymbol{M}(\boldsymbol{q})^{-1} \frac{\partial \boldsymbol{\phi}}{\partial \boldsymbol{q}}^{\intercal}}.$$

Bibliography