Computing Gradients of Inverse Problems in ODE Models

Juan Alfredo Gómez [1]   *Aymée Marrero* [2]

[1] Center of Mathematics and Theoretical Physics, ICIMAF
Havana, Cuba

[2] Faculty of Mathematics and Computer Sciences
University of Havana, Cuba

Abstract
The aim of this paper is to state and prove some basic formulas related with the numerical solution of the parameter estimation problem in nonlinear dynamical models. We consider a general approach to solve the inverse problem and prove several gradient and hessian's formulae for the continuous problem and for some of its discrete approximations. Regular and stiff schemes for ODE models with constant or variable step size policy are included.

1 General Approach

1.1 The parameter estimation problem

We consider the following continuous optimization problem:

$$\begin{array}{ll} \min & J(u) = \sum\limits_{i=1}^{s} \varphi_i \left[z_i(\tau_i), \bar{z}_i \right], \\ \text{s.t.} & \dot{x}(t) = f\left(x(t), u, t\right),\ t \in [0,T], \\ & x_0 = l(u), \\ & z(t) = g\left(x(t), u, t\right),\ t \in [0,T], \\ & 0 \le \tau_i < \tau_{i+1} \le T,\ i = 1, ..., s-1, \end{array} \tag{1}$$

where: $x \in \Re^n,\ u \in \Re^m,\ z \in \Re^p,\ \varphi_i : \Re^p \times \Re^p \to \Re,\ f : \Re^n \times \Re^m \times [0,T] \to \Re^n,\ l : \Re^m \to \Re^n,\ g : \Re^n \times \Re^m \times [0,T] \to \Re^p$. This means that we are modelling a dynamical process by a $n-$dimensional system of nonlinearordinary differential equations, which depends on an unknown $m-$vector of parameters u. To this end, a set of data (measurements) $\{\bar{z}_1, \bar{z}_2, ..., \bar{z}_s\}$ of the observed $p-$vector variable $z(.)$, at s different instants of time $\{\tau_1, \tau_2, ..., \tau_s\}$, is given and we should minimize J, a sum of functions depending on model and observed vectors (most frequently it is a quadratic function of the residuals $(z_i(\tau_i) - \bar{z}_i)$). We make the usual assumptions that $\tau_s = T$ and the model functions f, g, l, φ_i are continuously differentiable.

Parameter estimation problems for ODE systems (or ODE inverse problems) is a classical matter and have been considered by many authors. Several methods and points of view were proposed and special structures or statistical concepts were exploited (see for example [1], [3], [13] [19]). New schemes of numerical integration were also designed in order to deal with stiff ODE (see [6], [9] and [4]), appearing frequently in chemical reaction models (see also [20], [22]) and other important fields of applications.

In this article we are mainly interested in the numerical computation of the solution. It is clear that the problem (1) only can be approximately solved since, most time, the exact solution of the nonlinear differential equation can not be exactly calculated and the optimization algorithms are iterative process in character. Therefore, we decided not to try to solve the continuous problem (1), but to transform it in such a way that we obtain a simpler problem which gives us a satisfactory approximated solution.

1.2 Problem transformation

The general idea is to use numerical integration schemes as constraints, instead of the differential equation, transforming the continuous problem into a discrete one which can be solved in an easier way. This transformation depends directly on the numerical scheme of integration that is used. As a general example, we can consider the following discrete problem in which the system of ordinary differential equations is substituted by a multi-step scheme of variable order Q_i and variable step size h_i :

$$\begin{array}{ll} \min & J_k(u) = \sum_{i=0}^{k} \tilde{\varphi}_i \left[z_i, \tilde{z}_i \right], \\ \text{s.t.} & x_{i+1} = x_i + h_i F_i \left(x_i, x_{i-1}, ..., x_{i-Q_i+1}; u \right), \ i = 0, 1, 2, ..., k-1, \\ & x_0 = l(u), \\ & z_i = g_i \left(x_i, u \right), \ i = 0, 1, 2, ..., k. \end{array} \tag{2}$$

As a rule, the partition of integration:

$$\begin{aligned} t_0 &= 0, \ t_k = T \\ t_{i+1} &= t_i + h_i, \ i = 0, 1, ..., k-1, \end{aligned}$$

should contains the set of measurement times $\{\tau_1, \tau_2, ..., \tau_s\}$ and hence, to each measurement index $j \in \{1, 2, ..., s\}$ corresponds an integration index $i \in \{0, 1, 2, ..., k\}$. If we define the index correspondence:

$$\mathcal{I}(i) = \begin{cases} j & \text{if } i \text{ corresponds to } j, \\ 0 & \text{otherwise,} \end{cases} \quad , \ i = 0, 1, ..., k \tag{3}$$

and denote by M the set of integration indexes corresponding to measurements:

$$M = \{i \in \{0, 1, 2, ..., k\} \mid \mathcal{I}(i) \neq 0\},$$

then, the functions $\tilde{\varphi}_i$ in $J_k(u)$ are defined by:

$$\begin{aligned} \tilde{\varphi}_i[z_i, \tilde{z}_i] &= \varphi_{\mathcal{I}(i)}[z_i, \tilde{z}_i]\, \mathbf{1}_M[i], \\ \tilde{z}_i &= \bar{z}_{\mathcal{I}(i)}, \end{aligned}$$

where $\mathbf{1}_M$ is the characteristic function of the set M and $\varphi_0, \bar{z}_0$ can be defined arbitrarily.

The functions F_i can be taken in several forms. For example, the Adams integration formulae may enter in its definition, as a linear Predictor-Corrector scheme:

$$F_i\left(x_i, x_{i-1}, ..., x_{i-Q_i+1}, u\right) = K_0^{Q_i+1} f\left(y_{i+1}, u, t_{i+1}\right) + \sum_{j=1}^{Q_i} K_j^{Q_i+1} f\left(x_{i-j+1}, u, t_{i-j+1}\right), \tag{4}$$

$$y_{i+1} = x_i + h_i \sum_{j=1}^{Q_i} \Pi_j^{Q_i} f\left(x_{i-j+1}, u, t_{i-j+1}\right),\ i = 0, 1, ..., k-1. \tag{5}$$

where Π_j^Q, $K_j^Q$are the coefficients of the $Q-$order Adams-Bashford and Adams-Moulton schemes, respectively (see, for example, [23]).

The order policy of the scheme can be defined in many ways. For example, increase Q_i, step by step, until a given maximum order for the Corrector is attained $Q_{i_0} + 1 = Q_{\max}$, and then remain on it, $Q_i = Q_{\max} - 1$ for $i \geq i_0$:

$$Q_i = \min\left\{i + 1, Q_{\max} - 1\right\},\ i = 0, 1, ...k-1.$$

The substitution we made is some kind of "direct method" approach for the solution of the inverse problem.

1.3 Integration scheme for stiff problems

In 1976, Enright and Henrici proposed a family of implicit multistep-multiderivative nonlinear $Q-$order schemes with uniform step length, having several theoretical and practical advantages [4]. They are specially adapted for stiff problems and, for our purpose, we used the following simpler and better known second order formula, for stationary systems:

$$\begin{aligned} x_{i+1} = x_i + h \sum_{j=1}^{Q} \rho_j^{Q+2} f(x_{i-j+1}, u) + h\rho_0^{Q+2} f\left(x_{i+1}, u\right) + \\ + h^2 \rho_0^{Q+2} D_x f\left(x_{i+1}, u\right) . f\left(x_{i+1}, u\right). \end{aligned} \tag{6}$$

In order to decrease the complexity of the calculation of the resulting nonlinear equation and of the gradient computation, we considered also some semi-implicit variants of such scheme. The idea was to improve the corrector evaluation, using the Enright's second order formula as a recorrector. Some details of the variants and gradient formulas can be seen below. For results on numerical experiments see [17].

2 Theoretical results

2.1 General gradient formula

From practical experience, it is a common opinion that, no matter the algorithm we are using to solve a nonlinear optimization problem, the better the gradient is computed the better the optimal solution of the problem is (approximately) calculated. For this reason we recommend to avoid finite differences for computing gradients in this approach, and deduce and implement exact formulas, based in the following general lemma:

Lemma 1: Let X, U, Y, Z be normed spaces, $(\bar{x}, \bar{u}) \in X \times U$, $F : \mathcal{U} \to Y$ and $G : \mathcal{U} \to Z$ continuously Frechet differentiable operators defined in a neighborhood $\mathcal{U}$ of $(\bar{x}, \bar{u})$. Suppose there exists an implicit function:

$$x = x(u) : V \to X, \tag{7}$$

defined in neighborhood V of $\bar{u}$, continuously Frechet differentiable in V, which satisfies:

$$x(\bar{u}) = \bar{x}, \tag{8}$$

$$G(x(u), u) = \theta, \ \forall u \in V, \tag{9}$$

where θ denotes the zero element of the normed space Z. Then, the composite functional:

$$\mathcal{F} : V \to Y, \ \mathcal{F}(u) = F(x(u), u) \tag{10}$$

is Frechet differentiable in $\bar{u}$ and the following gradient formulae holds:

$$D_u\mathcal{F}(\bar{u}) = D_x F(\bar{x}, \bar{u}) \circ M + D_u F(\bar{x}, \bar{u}), \tag{11}$$

where $M \in \mathrm{L}(U, X)$ is any solution of the, so called, sensitivity equation:

$$D_x G(\bar{x}, \bar{u}) \circ M = -D_u G(\bar{x}, \bar{u}), \tag{12}$$

and also

$$D_u\mathcal{F}(\bar{u}) = \bar{p} \circ D_u G(\bar{x}, \bar{u}) + D_u F(\bar{x}, \bar{u}), \tag{13}$$

where $\bar{p} \in \mathrm{L}(Z, Y)$ is any solution of the, so called, adjoint (or conjugate) equation:

$$p \circ D_x G(\bar{x}, \bar{u}) = -D_x F(\bar{x}, \bar{u}), \tag{14}$$

and the symbol $\mathrm{L}(A, B)$ denotes the normed space of continuous linear operators from A to B.

Proof. By assumptions, $\mathcal{F}$ becomes Frechet differentiable and the chain rule gives:

$$D_u\mathcal{F}(\bar{u}) = D_xF(\bar{x}, \bar{u}) \circ D_u\bar{x}(u) + D_uF(\bar{x}, \bar{u}), \tag{15}$$

Using now (14) we can write:

$$D_u\mathcal{F}(\bar{u}) = -\bar{p} \circ D_xG(\bar{x}, \bar{u}) \circ D_ux(\bar{u}) + D_uF(\bar{x}, \bar{u}). \tag{16}$$

Denote $\mathcal{G}(u) = G(x(u), u)$. Using (9) and again the chain rule, we obtain the identity:

$$D_u\mathcal{G}(\bar{u}) = D_xG(\bar{x}, \bar{u}) \circ D_ux(\bar{u}) + D_uG(\bar{x}, \bar{u}) = \theta, \ \forall u \in V, \tag{17}$$

and then, setting $M = D_u\bar{x}(u)$, from (15) and (17) we have (11) and (12). In addition, from (17) we also have:

$$D_uG(\bar{x}, \bar{u}) = -D_xG(\bar{x}, \bar{u}) \circ D_ux(\bar{u}), \tag{18}$$

Substituying (18) in (16) we obtain (13). ■

Remark 1. Observe that if we define a Lagrangian-type operator as usual:

$$\begin{aligned} \mathcal{L}(x, u, p) &= F(x, u) + p \circ G(x, u), \\ x &\in X, \ u \in U, \ p \in \mathrm{L}(Z, Y), \end{aligned} \tag{19}$$

then, the equation (13) can be written as follows:

$$D_u\mathcal{F}(\bar{u}) = D_u\mathcal{L}(\bar{x}, \bar{u}, \bar{p}), \tag{20}$$

where $\bar{p}$ is any solution of the equation (14), which in turn can be written:

$$D_x\mathcal{L}(\bar{x}, \bar{u}, p) = 0. \tag{21}$$

Hence, this gives the following useful formulation of the Lemma 1.

Lemma 2: Under the conditions above, the gradient of the function $\mathcal{F}(u) = F(x(u), u)$, at the point $(\bar{x}, \bar{u}) = (x(\bar{u}), \bar{u})$, can be calculated through the following algorithm:

1) Write down the Lagrangian functional $\mathcal{L}(x, u, p)$,

2) Compute the partial derivative $D_x\mathcal{L}(x, u, p)$ at the point $(\bar{x}, \bar{u}, p)$,

3) Find a solution $\bar{p} \in \mathrm{L}(Z, Y)$ of the equation (21),

4) Fix $p = \bar{p}$ and compute the partial derivative $D_u\mathcal{L}(x, u, \bar{p})$ at $(\bar{x}, \bar{u})$,

5) Set $D_u\mathcal{F}(\bar{u}) = D_u\mathcal{L}(\bar{x}, \bar{u}, \bar{p})$.

Remark 2. The advantage of using formula (13) instead of (11) or viceversa, depends on the dimension of the spaces involved. Note that we do not need to know neither the implicit function $x(u)$ nor its derivative. But not always the equations (14) or (12) can be solved. A sufficient condition for the existence of a solution of (14) is that the linear operator $D_x G(\bar{x}, \bar{u})$ be right-invertible. Left-invertibility of $D_x G(\bar{x}, \bar{u})$ is the analogous condition for a solution of (12). Nevertheless, even when the equations (14) and (12) are not solvable, function $\mathcal{F}(u)$ is still Frechet differentiable but, in that case, we only can use the implicit formula (15).

We will see now some applications of this Lemma for gradient computation. The problem of computing gradients in situations similar to the one we considered here is a classical matter for one step schemes (see, for example, [21]). Recently, it has been analyzed in relation to automatic differentiation for Runge-Kutta schemes in ODE models [2] and for finite differences schemes in PDE models [5], including applications to optimal control problems. The algorithm given by Lemma 2 is a general solution for the gradient computation of discrete and continuous dynamical models, as the following examples show.

2.2 Gradient for continuous models

2.2.1 The continuous problem transformation

In the continuous problem (1), we consider:

$$x(.) \in \mathcal{C}^{1,n},\ z(.) \in \mathcal{C}^{0,p},\ u \in \Re^m, \tag{22}$$

where $\mathcal{C}^{r,k} = \mathcal{C}^r([0,T], \Re^k)$ denotes the Banach space of $r-$times continuously differentiable functions on $[0,T]$, with image into $\Re^k$, and the uniform $r-$norm:

$$\|x\|_{\mathcal{C}^{r,k}} = \sum_{j=0}^{r} \max_{t\in[0,T]} \left\| \frac{d^j x(t)}{dt^j} \right\|_{\Re^k},$$

with the usual convention $\frac{d^0 x(t)}{dt^0} = x(t)$.

The differential constraint of (1), together with the initial condition, are equivalent to the integral equation:

$$x(t) = l(u) + \int_0^t f(x(\tau), u, \tau) d\tau, \tag{23}$$

therefore, defining the functional:

$$F(x,u) = \sum_{i=1}^{s} \Phi_i(x,u)\ ,\ F : \mathcal{C}^{1,n} \times \Re^m \to \Re, \tag{24}$$

where the $\Phi_i : \mathcal{C}^{1,n} \times \Re^m \to \Re$ are given by the continuously differentiable functions:

$$\Phi_i(x,u) = \varphi_i[g(x(\tau_i),u,\tau_i),\bar{z}_i],\ i=1,...,s,$$

and introducing the operator:

$$G(x,u)(t) = x(t) - l(u) - \int_0^t f(x(\tau),u,\tau)d\tau\ ,\ G : \mathcal{C}^{1,n} \times \Re^m \to \mathcal{C}^{1,n}, \qquad (25)$$

we can rewrite the problem (1) in the following equivalent form:

$$\begin{array}{ll} \min & J(u) = F(x,u), \\ \text{s.t.} & G(x,u) = \theta_n. \\ & (x,u) \in X \times U, \end{array} \qquad (26)$$

where, with the notation of the Lemma 1, we have:

$$X = \mathcal{C}^{1,n},\ Y = \Re,\ Z = \mathcal{C}^{1,n}\ ,\ U = \Re^m, \qquad (27)$$

and θ_n is the null function in $\mathcal{C}^{1,n}$.

2.2.2 Application of the Lemma

In order to apply Lemma (1) we compute the Frechet differentials in a generic point $(\bar{x},\bar{u}) \in X \times U$. For the functional F, a direct computation yields the formula:

$$D_{(x,u)}F(\bar{x},\bar{u})(\eta,\mathfrak{v}) = \sum_{i=1}^{s} D_z\varphi_i[g(\bar{x}(\tau_i),\bar{u},\tau_i),\overline{z}_i]\,[g_x(\bar{x}(\tau_i),\bar{u},\tau_i)\eta(\tau_i) + g_u(\bar{x}(\tau_i),\bar{u},\tau_i)\mathfrak{v}\ ]\,, \qquad (28)$$

for all $(\eta,\mathfrak{v}) \in \mathcal{C}^{1,n} \times \Re^m$, since g and φ_i are continuously differentiable functions and the evaluation in τ_i is a linear operator.

Now we shall prove that $(x,u) \to D_{(x,u)}F(x,u)$ is a continuous operator at any point $(\bar{x},\bar{u})$ of $\mathcal{C}^{1,n} \times \Re^m$ using standard arguments of uniform continuity. In fact, consider the compact set:

$$R = \text{Im}(\bar{x}) = \{x \in \Re^n \mid \exists\, t \in [0,T] : x = \bar{x}(t)\}\,,$$

its corresponding $\delta-$neighborhood:

$$R^\delta = \left\{x \in \Re^n \mid d(x,R) = \min_{y \in R} \|x-y\|_{\Re^n} \le \delta\right\},$$

and the unit closed ball at $\bar{u}$:

$$B = \{u \in \Re^m \mid \|u-\bar{u}\|_{\Re^m} \le 1\}.$$

By the uniform continuity of the function $(x,u) \to D_z\varphi_i[g(x,u,\tau_i),\bar{z}_i]g_x(x,u,\tau_i)$ in the compact set $R^\delta \times B$, for all $i = 1,...,s$, and for all $\varepsilon > 0$, there exists $\delta_1 \in (0,1)$, such that, for arbitrary $(x_1,u_1),(x_2,u_2) \in R^\delta \times B$, satisfying $\|x_1 - x_2\|_{\mathcal{C}^{1,n}} + \|u_1 - u_2\|_{\Re^m} < \delta_1$, we have:

$$|D_z\varphi_i[g(x_1,u_1,\tau_i),\bar{z}_i]g_x(x_1,u_1,\tau_i) - D_z\varphi_i[g(x_2,u_2,\tau_i),\bar{z}_i]g_x(x_2,u_2,\tau_i)| < \frac{\varepsilon}{s}.$$

Taking $x \in \mathcal{C}^{1,n}$ such that $\|x - \bar{x}\|_{\mathcal{C}^{1,n}} < \min\{\delta, \frac{\delta_1}{2}\}$, and $\|u - \bar{u}\|_{\Re^m} < \frac{\delta_1}{2}$, then:

$$(x(\tau_i),u) \in R^\delta \times B \text{ and } \| x(\tau_i) - \bar{x}(\tau_i)\|_{\Re^n} + \|u_1 - u_2\|_{\Re^m} < \delta_1,$$

and we obtain:

$$\begin{aligned}
&|D_xF(x,u)(\eta) - D_xF(\bar{x},\bar{u})(\eta)| \leq \\
&\leq \sum_{i=1}^{s} \|D_z\varphi_i[g(x(\tau_i),u,\tau_i),\bar{z}_i]g_x(x(\tau_i),u,\tau_i) - \\
&-D_z\varphi_i[g(\bar{x}(\tau_i),\bar{u},\tau_i),\bar{z}_i]g_x(\bar{x}(\tau_i),\bar{u},\tau_i)\|_{\Re^p} \|\eta(\tau_i)\|_{\Re^n} \leq \\
&\leq \varepsilon \ \|\eta\|_{\mathcal{C}^{1,n}} .
\end{aligned}$$

Analogously, a similar inequality can be proved for D_uF, and therefore:

$$\left\|D_{(x,u)}F(x,u) - D_{(x,u)}F(\bar{x},\bar{u})\right\|_{(\mathcal{C}^{0,p})^*} \leq \varepsilon,$$

which implies continuity of $D_{(x,u)}F(x,u)$.

On the other hand, the partial derivatives of G have the form:

$$D_xG(\overline{x},\overline{u})(\eta)(t) = \eta(t) - \int_0^t f_x(\overline{x}(\tau),\overline{u},\tau)\eta(\tau)d\tau \tag{29}$$

$$D_uG(\overline{x},\overline{z},\overline{u})(\mathfrak{v})(t) = -\left[D_ul(\overline{u}) + \int_0^t f_u(\overline{x}(\tau),\overline{u},\tau)d\tau\right]\mathfrak{v} \tag{30}$$

and the Frechet differential:

$$D_{(x,u)}G(\overline{x},\overline{u})(\eta,\mathfrak{v})(t) = \eta(t) - D_ul(\overline{u})\mathfrak{v} - \int_0^t (f_x(\overline{x}(\tau),\overline{u},\tau)\eta(\tau) + f_u(\overline{x}(\tau),\overline{u},\tau)\mathfrak{v} \ ) \, d\tau \tag{31}$$

To prove the continuity of $D_{(x,u)}G$, we proceed in the same way as before, using the uniform continuity of $f_x(.)$, $f_u(.)$ and $D_ul(.)$ over $R^\delta \times B \times [0,T]$ and choosing appropiate bounds. We obtain:

$$\left\|\left[D_{(x,u)}G(x,u) - D_{(x,u)}G(\bar{x},\bar{u})\right](\eta,\mathfrak{v})\right\| \ \leq \varepsilon\left(\|\eta\|_{\mathcal{C}^{1,n}} + \|\mathfrak{v}\|_{\Re^m}\right),$$

which gives the continuity of $D_{(x,u)}G$.

We still need that the Frechet partial differential $D_xG(\bar{x},\bar{u})$ should be a homeomorphism because, in that case, we can apply the implicit function theorem (see [11] or [7]) and Lemma 1. In fact, $D_xG(\bar{x},\bar{u})$ is a bijective, linear and continuous operator, since for fixed $(\bar{x},\bar{u})$, the equation:

$$y(t) = \eta(t) - \int_0^t f_x(\overline{x}(\tau),\overline{u},\tau)\eta(\tau)d\tau,\ t \in [0,T],$$

is a Volterra integral equation of the second type, with bounded kernell, and therefore, for every function $y(t) \in \mathcal{C}^{1,n}$, there exists a unique solution $\eta(t) \in \mathcal{C}^{1,n}$ of the integral equation (see [11] or [7]). Hence, the inverse operator exists and is continuous. This proves that $D_xG(\bar{x},\bar{u})$ is a homeomorphism.

Then, by the implicit function theorem, there exists a neighborhood $V \subset \Re^m$ of $\bar{u}$ where the implicit function is defined and is Frechet differentiable:

$$x = x(u)\ ,\ x : V \to \mathcal{C}^1([0,T],\Re^n). \tag{32}$$

By the above Lemma, the composite function:

$$\mathcal{F} : V \to \Re,\ \mathcal{F}(u) = F(x(u),u)$$

is Frechet differentiable in $\overline{u}$, we have the two formulae:

$$D_u\mathcal{F}(\overline{u}) = D_xF(\bar{x},\bar{u})D_ux(\bar{u}) + D_uF(\bar{x},\bar{u}), \tag{33}$$

$$D_u\mathcal{F}(\overline{u}) = p \circ D_uG(\overline{x},\overline{u}) + D_uF(\bar{x},\bar{u}), \tag{34}$$

and $p \in \left(\mathcal{C}^{1,n}\right)^*$is an arbitrary solution of the equation:

$$p \circ D_xG(\overline{x},\overline{u}) = -D_xF(\overline{x},\overline{u}). \tag{35}$$

Note that (33) and (34) are two equivalent, but different expresions of the continuous gradient $\nabla J(\bar{u}) = D_u\mathcal{F}(\overline{u})$. The first one uses the $n \times m-$matrix $D_ux(\bar{u})$, so called "sensitivity matrix", and the second one uses the solution p of the adjoint equation, when it happens to exist.

2.2.3 Gradient expression using sensitivity matrix

In this case, the formula (33) is not so useless as it seems in a first sight. We can use the sensitivity equation (12) which allows us to compute the sensitivity matrix $D_u x(\bar{u})$, and it is possible to give a direct proof of this. In fact, as we already saw in (23), the function $x(t) = x(u)(t)$ is a solution of the integral equation:

$$x(t) = l(u) + \int_0^t f(x(\tau), u, \tau) d\tau, \ t \in [0, T], \tag{36}$$

and we proved that the operator $u \to x(u)$, from $\Re^m$ to $\mathcal{C}^{1,n}$, is defined in a neighborhood V of $\bar{u}$ and is Frechet differentiable at $\bar{u}$. Therefore, the Frechet differential operator $D_u x(\bar{u})$, defined in V, satisfies the integral equation:

$$D_u x(\bar{u})(t) = D_u l(\bar{u}) + \int_0^t \left[f_x(\bar{x}(\tau), \bar{u}, \tau) D_u x(\bar{u})(\tau) + f_u(\bar{x}(\tau), \bar{u}, \tau) \right] d\tau, \ t \in [0, T]. \tag{37}$$

where this last formula has been obtained computing the Frechet derivative of the operator given by the right hand side of (36). Then, the sensitivity matrix $M(t) = D_u x(\bar{u})(t)$ can be found solving the equivalent matrix differential equation:

$$\begin{array}{l} \frac{dM(t)}{dt} = f_x(\bar{x}(t), \bar{u}, t) M(t) + f_u(\bar{x}(t), \bar{u}, t), \ t \in [0, T], \\ M(0) = D_u l(\bar{u}). \end{array} \tag{38}$$

The final expresion for the gradient, when we use (33), is the following:

$$\nabla J(\bar{u}) = \sum_{i=1}^{s} \left[D_x \Phi_i(\bar{x}(\tau_i), \bar{u}) M(\tau_i) + D_u \Phi_i(\bar{x}(\tau_i), \bar{u}) \right], \tag{39}$$

or

$$\nabla J(\bar{u}) = \sum_{i=1}^{s} D_z \varphi_i [g(\bar{x}(\tau_i), \bar{u}, \tau_i), \overline{z}_i] \left[g_x(\bar{x}(\tau_i), \bar{u}, \tau_i) M(\tau_i) + g_u(\bar{x}(\tau_i), \bar{u}, \tau_i) \right], \tag{40}$$

where the matrix $M(t)$ is the solution of (38).

2.2.4 Gradient expression using adjoint equation

In order to apply the second formula (34), we must find a solution of the adjoint equation. This a non trivial task, and we shall see that it yields a more complicated expression for the continuous gradient. Then, this second formula is used only in some particular cases. Nevertheless, as we shall show below, for the gradient of *discrete time* dynamical models the adjoint form is always preferred.

We start using the expression of $D_x F$ from (28) in (35), obtaining:

$$p \circ D_x G(\overline{x}, \overline{u})(\eta) = \sum_{i=1}^{s} D_z \varphi_i [g(\overline{x}(\tau_i), \overline{u}, \tau_i), \bar{z}_i] g_x(\overline{x}(\tau_i), \overline{u}, \tau_i) \eta(\tau_i). \tag{41}$$

According to the Riesz theorem, which characterizes continuous linear functionals over $\mathcal{C}^{0,n}$, we will try to find the solution $p \in \left(\mathcal{C}^{1,n}\right)^*$ of (41) in the following form:

$$p[y] = \int_0^T y(\tau) d\psi(\tau),$$

where the integral is considered in the Riemann-Stieljes sense, and the bounded variation function $\psi(t)$ will be conveniently chosen. Note that the solution does not necessarily exist, since $\left(\mathcal{C}^{0,n}\right)^*$ is strictly contained in $\left(\mathcal{C}^{1,n}\right)^*$, but if it do exists, then it should be unique, since $D_x G(\overline{x}, \overline{u})$ is an homeomorphism.

The right hand side of the equality (41) suggests that ψ must have jump discontinuities at the points $\{\tau_i,\ i = 1, ..., s\}$. Recalling that $\tau_s = T$, we define:

$$\begin{array}{l} p[y] = \sum_{i=1}^{s-1} \left[\psi(\tau_i + 0) - \psi(\tau_i - 0)\right]^{\mathrm{T}} y(\tau_i) - \psi(\tau_s - 0) y(\tau_s) + \\ \qquad + \int_0^{\tau_1} \frac{d\psi(\tau)}{d\tau} y(\tau) d\tau + \sum_{i=1}^{s-1} \int_{\tau_i}^{\tau_{i+1}} \frac{d\psi(\tau)}{d\tau} y(\tau) d\tau, \end{array}$$

where $\psi(t)$ is supposed to have continuous derivative at each open subinterval $(0, \tau_1)$, $(\tau_i, \tau_{i+1}),\ i = 1, ..., s$.

Using (29) and notations:

$$\begin{array}{l} \bar{f}(t) = f(\overline{x}(t), \overline{u}, t), \\ \bar{g}(t) = g(\overline{x}(t), \overline{u}, t), \end{array} \tag{42}$$

in (41) we have:

$$\begin{array}{l} p \circ D_x G(\overline{x}, \overline{z}, \overline{u})(\eta) = \sum_{i=1}^{s-1} \left[\psi(\tau_i + 0) - \psi(\tau_i - 0)\right]^{\mathrm{T}} \left[\eta(\tau_i) - \int_0^{\tau_i} \bar{f}_x(\tau)\eta(\tau) d\tau\right] - \\ -\psi^{\mathrm{T}}(\tau_s - 0) \left[\eta(\tau_s) - \int_0^{\tau_s} \bar{f}_x(\tau)\eta(\tau) d\tau\right] + \int_0^{\tau_1} \frac{d\psi^{\mathrm{T}}(\tau)}{d\tau} \left[\eta(\tau) - \int_0^{\tau} \bar{f}_x(\sigma)\eta(\sigma) d\sigma\right] d\tau + \\ + \sum_{i=1}^{s-1} \int_{\tau_i}^{\tau_{i+1}} \frac{d\psi^{\mathrm{T}}(\tau)}{d\tau} \left[\eta(\tau) - \int_0^{\tau} \bar{f}_x(\sigma)\eta(\sigma) d\sigma\right] d\tau = \sum_{i=1}^{s} D_z \varphi_i[\bar{g}(\tau_i), \bar{z}_i] \bar{g}_x(\tau_i) \eta(\tau_i). \end{array} \tag{43}$$

The last equality allows us to define the values of the jumps of ψ at the points of discontinuity:

$$\begin{array}{rcl} \psi(\tau_i + 0) - \psi(\tau_i - 0) & = & D_z \varphi_i[g(\overline{x}(\tau_i), \overline{u}, \tau_i), \bar{z}_i] g_x(\overline{x}(\tau_i), \overline{u}, \tau_i),\ i = 1, ..., s, \\ \psi(\tau_s + 0) & = & \psi(T + 0) = 0. \end{array} \tag{44}$$

Simplifying and introducing the terms $\pm \psi^{\mathrm{T}}(\tau_s + 0) \int_0^{\tau_s} \bar{f}_x(\tau)\eta(\tau) d\tau$ in (43), we obtain the identity:

$$\begin{array}{l} -\sum_{i=1}^{s} \left[\psi(\tau_i + 0) - \psi(\tau_i - 0)\right]^{\mathrm{T}} \int_0^{\tau_i} \bar{f}_x(\tau)\eta(\tau) d\tau + \\ +\psi^{\mathrm{T}}(\tau_s + 0) \int_0^{\tau_s} \bar{f}_x(\tau)\eta(\tau) d\tau + \int_0^{\tau_1} \frac{d\psi^{\mathrm{T}}(\tau)}{d\tau} \left[\eta(\tau) - \int_0^{\tau} \bar{f}_x(\sigma)\eta(\sigma) d\sigma\right] d\tau + \\ + \sum_{i=1}^{s-1} \int_{\tau_i}^{\tau_{i+1}} \frac{d\psi^{\mathrm{T}}(\tau)}{d\tau} \left[\eta(\tau) - \int_0^{\tau} \bar{f}_x(\sigma)\eta(\sigma) d\sigma\right] d\tau = 0. \end{array} \tag{45}$$

Integrating by parts we have:

$$\begin{array}{l}\int_0^{\tau_1} \frac{d\psi^{\mathrm{T}}(\tau)}{d\tau}\left(\int_0^{\tau} \bar{f}_x(\sigma)\eta(\sigma)d\sigma\right)d\tau = \\ = \psi(\tau_1 - 0)\left(\int_0^{\tau_1} \bar{f}_x(\sigma)\eta(\sigma)d\sigma\right)d\tau - \int_0^{\tau_1} \psi(\tau)\bar{f}_x(\tau)\eta(\tau)d\tau\end{array}$$

and

$$\begin{array}{l}\int_{\tau_i}^{\tau_{i+1}} \frac{d\psi^{\mathrm{T}}(\tau)}{d\tau}\left(\int_0^{\tau} \bar{f}_x(\sigma)\eta(\sigma)d\sigma\right)d\tau = \\ = \psi(\tau_{i+1} - 0)\left(\int_0^{\tau_{i+1}} \bar{f}_x(\sigma)\eta(\sigma)d\sigma\right)d\tau - \psi(\tau_i + 0)\left(\int_0^{\tau_i} \bar{f}_x(\sigma)\eta(\sigma)d\sigma\right)d\tau - \\ - \int_{\tau_i}^{\tau_{i+1}} \psi(\tau)\bar{f}_x(\tau)\eta(\tau)d\tau.\end{array}$$

Using these last expressions in (45):

$$\begin{array}{l}-\sum_{i=1}^{s}\left[\psi(\tau_i + 0) - \psi(\tau_i - 0)\right]^{\mathrm{T}} \int_0^{\tau_i} \bar{f}_x(\tau)\eta(\tau)d\tau + \\ +\psi^{\mathrm{T}}(\tau_s + 0)\int_0^{\tau_s} \bar{f}_x(\tau)\eta(\tau)d\tau + \int_0^{\tau_1} \frac{d\psi^{\mathrm{T}}(\tau)}{d\tau}\eta(\tau)d\tau - \\ -\psi(\tau_1 - 0)\left(\int_0^{\tau_1} \bar{f}_x(\sigma)\eta(\sigma)d\sigma\right)d\tau + \int_0^{\tau_1} \psi(\tau)\bar{f}_x(\tau)\eta(\tau)d\tau + \\ + \sum_{i=1}^{s-1}\int_{\tau_i}^{\tau_{i+1}} \frac{d\psi^{\mathrm{T}}(\tau)}{d\tau}\eta(\tau)d\tau - \\ - \sum_{i=1}^{s-1}\left[\psi(\tau_{i+1} - 0)\left(\int_0^{\tau_{i+1}} \bar{f}_x(\sigma)\eta(\sigma)d\sigma\right)d\tau - \psi(\tau_i + 0)\left(\int_0^{\tau_i} \bar{f}_x(\sigma)\eta(\sigma)d\sigma\right)d\tau\right] + \\ + \sum_{i=1}^{s-1}\int_{\tau_i}^{\tau_{i+1}} \psi(\tau)\bar{f}_x(\tau)\eta(\tau)d\tau,\end{array}$$

and simplifying, we obtain:

$$\begin{array}{l}\int_0^{\tau_1}\left[\frac{d\psi^{\mathrm{T}}(\tau)}{d\tau} + \psi(\tau)\bar{f}_x(\tau)\right]\eta(\tau)d\tau + \\ + \sum_{i=1}^{s-1}\int_{\tau_i}^{\tau_{i+1}}\left[\frac{d\psi^{\mathrm{T}}(\tau)}{d\tau} + \psi^{\mathrm{T}}(\tau)\bar{f}_x(\tau)\right]\eta(\tau)d\tau = 0.\end{array}$$

Since η is arbitrary, we have that ψ must be the solution of the equation:

$$\frac{d\psi^{\mathrm{T}}(t)}{d\tau} = -\psi^{\mathrm{T}}(t)f_x(\bar{x}(t), \bar{u}, t), \tag{46}$$

at each subinterval $(0, \tau_1)$, (τ_i, τ_{i+1}), $i = 1, ..., s$, and satisfies the initial (or jump) conditions given by (44). The exact values of ψ at the points of discontinuities: $\psi(\tau_i)$, $i = 1, ..., s$, are not important for gradient calculation, but only the values of the finite jumps $[\psi(\tau_i + 0) - \psi(\tau_i - 0)]$. The function ψ can be defined left or right continuous at those points, arbitrarily, but for the extreme points 0 and $\tau_s = T$, it is usual to consider ψ as right continuos and left continuous, respectively, i.e.

$$\begin{array}{rcl}\psi(0) & = & \psi(0 + 0), \\ \psi(\tau_s) & = & \psi(\tau_s - 0).\end{array}$$

Substituting the expression of p in (34) we have:

$$
\begin{aligned}
D_u\mathcal{F}(\bar{u})(\mathfrak{v}) = &- \textstyle\sum_{i=1}^{s}\left[\psi(\tau_i+0)-\psi(\tau_i-0)\right]^{\mathrm{T}}\left[D_u l(\overline{u}) + \int\limits_0^{\tau_1}\bar{f}_u(\tau)d\tau\right]\mathfrak{v}+ \\
&+\psi^{\mathrm{T}}(\tau_s-0)\left[D_u l(\overline{u}) + \int\limits_0^{\tau_s}\bar{f}_u(\tau)d\tau\right]\mathfrak{v} - \int_0^{\tau_1}\frac{d\psi^{\mathrm{T}}(\tau)}{d\tau}\left[D_u l(\overline{u}) + \int\limits_0^{\tau}\bar{f}_u(\sigma)d\sigma\right]\mathfrak{v}\,d\tau- \\
&-\textstyle\sum_{i=1}^{s-1}\int_{\tau_i}^{\tau_{i+1}}\frac{d\psi^{\mathrm{T}}(\tau)}{d\tau}\left[D_u l(\overline{u}) + \int\limits_0^{\tau}\bar{f}_u(\sigma)d\sigma\right]\mathfrak{v}\,d\tau + \sum\limits_{i=1}^{s} D_z\varphi_i[\bar{g}(\tau_i),\bar{z}_i]\bar{g}_u(\tau_i)\mathfrak{v},
\end{aligned}
$$

where we used again the notations (42).

Integrating by parts and simplifying, as we did above, we obtain the final expression for the gradient:

$$
\begin{aligned}
\nabla J(\bar{u}) = D_u\mathcal{F}(\bar{u}) = \bar{\psi}^{\mathrm{T}}(0)D_u l(\overline{u}) + \textstyle\int_0^{\tau_1}\bar{\psi}^{\mathrm{T}}(\tau)f_u(\overline{x}(t),\overline{u},t)d\tau+ \\
+\textstyle\sum_{i=1}^{s-1}\int_{\tau_i}^{\tau_{i+1}}\bar{\psi}^{\mathrm{T}}(\tau)f_u(\overline{x}(t),\overline{u},t)d\tau + \sum\limits_{i=1}^{s} D_z\varphi_i[g(\overline{x}(\tau_i),\overline{u},\tau_i),\bar{z}_i]g_u(\overline{x}(\tau_i),\overline{u},\tau_i),
\end{aligned}
\tag{47}
$$

where $\bar{\psi}(t)$ is the solution of the adjoint equation (46) at each subinterval $(0,\tau_1)$, (τ_i,τ_{i+1}), $i=1,...,s$, with the initial (jump) conditions given by (44).

The advantage of the expression (47), in comparison with (39), is that the adjoint equation (46) is a vector instead of a matrix differential equation, i.e. we have n differential equations for the adjoint function, instead of n^2 equations, given by (38), for the sensitivity matrix. As drawbacks, we can mention a more difficult integration of the system (46) because, during computation, we have to keep in mind the discontinuities, and also the more complex form of the gradient, which some times is an obstacle for further theoretical developments.

2.3 Gradient for discrete models with fixed steplengths

2.3.1 Gradient for multistep explicit models

We consider now the discrete problem (2). First, we suppose that the integration step lengths h_i are given and fixed, for all $i=0,1,...,k-1$. A usual example is when a uniform step length $h_i=h$, $\forall i$, is chosen. In the next section we analyze a general model for automatic step length control.

Introduce the notations:

$$\mathbf{x} = (x_0,x_1,...,x_k)^{\mathrm{T}},$$

$$\mathbf{x}_i = (x_0,x_1,...,x_{i+1-Q_i})^{\mathrm{T}},\ i=0,1,...,k-1,$$

$$\Phi_i(x_i,u) = \tilde{\varphi}_i[g_i(x_i,u),\bar{z}_{\mathcal{I}(i)}],\ i=0,1,...,k,$$

$$\tilde{\varphi}_i[z_i,\tilde{z}_i] = \varphi_{\mathcal{I}(i)}[z_i,\bar{z}_{\mathcal{I}(i)}]\,\mathbf{1}_M[i],\ i=0,1,...,k,$$

$$\tilde{\Phi}_j(x_j,u)=\varphi[g_{\mathcal{J}(j)}(x_{\mathcal{J}(j)},u),\bar{z}_j],\ j=1,2,...,s,$$

where $\mathcal{I}:\{0,1,...,k\}\to\{0,1,...,s\}$ was defined in (3) and $\mathcal{J}:\{1,...,s\}\to\{0,1,...,k\}$ is a left inverse of $\mathcal{I}$ over $\{1,...,s\}$, i.e. $\mathcal{J}(j)$ is equal to the unique $i\in\{0,...,k\}$ such that $\mathcal{I}(i)=j$. We also introduce the operators:

$$F(\mathbf{x},u)=\sum_{i=0}^{k}\Phi_i(x_i,u)=\sum_{j=1}^{s}\tilde{\Phi}_j(x_j,u),$$

$$G(\mathbf{x},u)=(x_0-l(u),x_1-x_0-h_0F_0(\mathbf{x}_0,u),...,x_k-x_{k-1}-h_{k-1}F_{k-1}(\mathbf{x}_{k-1},u))^{\mathrm{T}},$$

and it is easy to see that the problem (2) can be written in the following equivalent form:

$$\begin{array}{ll}\min & J_k(u)=F(\mathbf{x},u),\\ \text{s.t.} & G(\mathbf{x},u)=0_{kn}.\\ & (\mathbf{x},u)\in X\times U,\end{array}\qquad(48)$$

where:

$$X=\Re^{k\times n},\ U=\Re^{m}.$$

If the functions $\tilde{\varphi}_i, g_i$ and F_i are defined for all $(x,u)\in X\times U$ and are continuously differentiable then, the operators F and G are also continuously differentiable and we have the following expressions for the partial derivatives:

$$D_{\mathbf{x}}F(\mathbf{x},u)=\left[\begin{array}{cccc}D_x\Phi_0(x_0,u) & D_x\Phi_1(x_1,u) & \cdots & D_x\Phi_k(x_k,u)\end{array}\right],$$

$$D_uF(\mathbf{x},u)=\sum_{i=0}^{k}D_u\Phi_i(x_i,u),$$

$$D_{\mathbf{x}}G(\mathbf{x},u)=\left[\begin{array}{cccc}I & 0 & \cdots & 0\\ -I-h_0D_{x_0}F_0(\mathbf{x}_0,u) & I & \cdots & 0\\ -h_1D_{x_0}F_1(\mathbf{x}_1,u) & -I-h_1D_{x_1}F_1(\mathbf{x}_1,u) & \cdots & 0\\ \cdots & \cdots & \cdots & \cdots\\ -h_{k-1}D_{x_0}F_{k-1}(\mathbf{x}_{k-1},u) & -h_{k-1}D_{x_1}F_{k-1}(\mathbf{x}_{k-1},u) & \cdots & I\end{array}\right],$$

$$D_uG(\mathbf{x},u)=\left[\begin{array}{c}-D_ul(u)\\ -h_0D_uF_0(\mathbf{x}_0,u)\\ \vdots\\ -h_{k-1}D_uF_{k-1}(\mathbf{x}_{k-1},u)\end{array}\right].$$

Furthermore, it is clear that the difference scheme defines an implicit continuously differentiable function $\mathbf{x}(u)$, such that $G(\mathbf{x}(u),u)=0$, for all $u\in U$. We can apply

Lemma 1 and then, the composite function $\mathcal{F}(u) = F(\mathbf{x}(u), u)$ is differentiable at any $\bar{u} \in U$, and the following formulae hold:

$$\nabla J_k(\bar{u}) = D_{\mathbf{x}} F(\bar{\mathbf{x}},\bar{u})\, D_u \mathbf{x}(\bar{u}) + D_u F(\bar{\mathbf{x}}, \bar{u}), \tag{49}$$

$$\nabla J_k(\bar{u}) = \bar{\mathbf{p}} \circ D_u G(\bar{\mathbf{x}}, \overline{u}) + D_u F(\bar{\mathbf{x}}, \bar{u}), \tag{50}$$

where we denote $\bar{\mathbf{x}} = \mathbf{x}(\bar{u})$. The $(k+1) * n \times m-$matrix $D_u \mathbf{x}(\bar{u})$ is the solution of the algebraic matrix system:

$$D_{\mathbf{x}} G(\bar{\mathbf{x}}, \bar{u}) D_u \mathbf{x}(\bar{u}) + D_u G(\bar{\mathbf{x}}, \bar{u}) = \theta_{(k+1)n \times m}$$

and $\bar{\mathbf{p}}$ is the solution of the adjoint vector system:

$$\mathbf{p}^{\mathrm{T}} D_{\mathbf{x}} G(\bar{\mathbf{x}}, \bar{u}) = -D_{\mathbf{x}} F(\bar{\mathbf{x}}, \bar{u}).$$

If we define $D_u \mathbf{x}(\bar{u}) = (M_0, M_1, ..., M_k)^{\mathrm{T}}$, $M_i \in \Re^{n \times m}$, and $\bar{\mathbf{p}} = (p_0, p_1, ..., p_k)^{\mathrm{T}}$, $p_i \in \Re^n$, it is easy to see that both auxiliary systems can be rewritten in the following form:

$$\begin{aligned} M_0 &= D_u l(\bar{u}), \\ M_{i+1} &= M_i + h_i \left[\sum_{j=1}^{Q_i} D_{x_{i-j+1}} F_i(\bar{\mathbf{x}}_i, \bar{u}) M_{i-j+1} + D_u F_i(\bar{\mathbf{x}}_i, \bar{u}) \right],\ i = 0, 1, ..., k-1, \end{aligned} \tag{51}$$

$$\begin{aligned} p_k^{\mathrm{T}} &= -D_x \Phi_k(x_k, \bar{u}) \\ p_j^{\mathrm{T}} &= p_{j+1}^{\mathrm{T}} + \textstyle\sum_{i=j}^{k-1} p_{i+1}^{\mathrm{T}} h_i D_{x_i} F_i(\bar{\mathbf{x}}_i, \bar{u}) - D_x \Phi_j(\bar{x}_j, \bar{u}),\ j = k-1, ..., 0. \end{aligned} \tag{52}$$

and then, the final expressions for both gradients follows:

$$\nabla J_k(\bar{u}) = \sum_{i=0}^{k} \left[D_x \Phi_i(\bar{x}_i, \bar{u}) M_i + D_u \Phi_i(\bar{x}_i, \bar{u}) \right], \tag{53}$$

$$\nabla J_k(\bar{u}) = -p_0^{\mathrm{T}} D_u l(\bar{u}) - \sum_{i=0}^{k-1} p_{i+1}^{\mathrm{T}} h_i D_u F_i(\bar{\mathbf{x}}_i, \bar{u}) + \sum_{i=0}^{k} D_u \Phi_i(\bar{x}_i, \bar{u}). \tag{54}$$

Now it is clear that, in the case of discrete dynamical models, formula (54) is always preferred against (53) since the discrete adjoint system (52) is $n-$dimensional and the discrete sensitivity system (51) is $n \times m-$dimensional. In fact, the total number of dot products in (52) and (54) is less than $(k * n + (k+1) * m)$, while the number of dot products in (51) and (53) is greater than $((k-1)^2 * n^2 + k * n * m)$, and this means a great handicap in computational efforts.

2.3.2 Gradient for implicit and semi-implicit models

Up to now, we constraint ourselves to explicit schemes, but similar formulas can be derived for implicit schemes, following the above general procedure. For example, if the right hand side function depends on $x_{i+1}, F(x_{i+1}, x_i, ..., u)$, the only difference appearing is that both auxiliary systems become implicit schemes, but the gradient expressions are not more complex and can be easily obtained. The results for the particular case of Enright's scheme and a its semi-implicit variant for stationary ODE models are the following:

The implicit model is:

$$\begin{array}{ll} \min & J(u) = \sum\limits_{i=0}^{k} \tilde{\varphi}_i \left[z_i, \bar{z}_i\right], \\ \text{s.t.} & x_{i+1} = x_i + h \sum\limits_{j=1}^{Q_i} \rho_j^{Q_i+2} f_{i-j+1}(x_{i-j+1}, u) + h\rho_0^{Q_i+2} f_{i+1}\left(x_{i+1}, u\right) + \\ & + h^2 \rho_0^{Q_i+2} D_x f_{i+1}\left(x_{i+1}, u\right) . f_{i+1}\left(x_{i+1}, u\right),\ i = 0, 1, 2, ..., k-1, \\ & x_0 = l(u), \\ & z_i = g_i\left(x_i, u\right),\ i = 0, 1, 2, ..., k. \end{array}$$

and the corresponding gradient formula follows:

$$\begin{aligned} \nabla J(\bar{u}) \quad = \quad & -\sum_{i=1}^{k} h \left\{ p_{i+1}^{\mathrm{T}} \rho_0^{Q_i+2} D_u f(x_{i+1}, u) + p_{i+1}^{\mathrm{T}} h \rho_0^{Q_i+2} [D_{xu} f\left(x_{i+1}, u\right) f\left(x_{i+1}, u\right) + \right. \\ & \left. + D_x f\left(x_{i+1}, u\right) D_u f\left(x_{i+1}, u\right)] \right\} - \sum_{r=0}^{k-1} \sum_{i=r}^{Q_r - 1} h p_{i+1}^{\mathrm{T}} \rho_{i+1-r}^{Q_i+2} D_u f\left(x_r, u\right) - \\ & - \sum_{i=0}^{k} \lambda_i^{\mathrm{T}}\ D_u^{\mathrm{T}} g\left(x_i, u\right) - p_0^{\mathrm{T}} D_u^{\mathrm{T}} l(\bar{u}), \end{aligned}$$

where the multipliers p_i and λ_i are the solutions of the following implicit discrete system:

$$\begin{aligned} & p_i^{\mathrm{T}} \left[I - h.\rho_0^{Q_i+2} D_x f_i\left(x_i, u\right) - h^2 \rho_0^{Q_i+2} \left(D_{xx} f_i(x_i, u) . f_{i+1}(x_i, u) + D_x^2 f_i(x_i, u) \right) \right] = \\ & p_{i+1}^{\mathrm{T}} + \lambda_i^{\mathrm{T}}\ D_x^{\mathrm{T}} g_i\left(x_i, u\right) + \textstyle\sum_{r=0}^{k-1} \sum_{j=1}^{Q_i-1} h \left[q_{j+1}^{\mathrm{T}} \pi_{j+i-1}^{Q_i} + q_{j+1}^{c\mathrm{T}} K_{j+i-1}^{Q_i+1} + p_{j+1}^{\mathrm{T}} \rho_{j+i-1}^{Q_i+2} \right] D_x f_i\left(x_i, u\right), \\ & \qquad\qquad\qquad\qquad\qquad\qquad\qquad\qquad\qquad\qquad\qquad i = 0, 1, ..., k-1, \end{aligned}$$

$$p_k^{\mathrm{T}} \left[I - h.\rho_0^{Q_i+2} D_x f_k\left(x_k, u\right) - h^2 \rho_0^{Q_i+2} \left(D_{xx} f_k(x_k, u) . f_k(x_k, u) + D_x^2 f_k(x_k, u) \right) \right] = \lambda_k^{\mathrm{T}} D_x g_k(x_k, u)$$

$$\lambda_i^{\mathrm{T}} = -D_x \varphi(z_i, u), i = 0, 1, ..., k$$

The semi-implicit model is a predictor-corrector-recorrector scheme:

$$
\begin{aligned}
\min \quad & J(u) = \sum_{i=0}^{k} \tilde{\varphi}_i \left[z_i, \bar{z}_i \right], \\
\text{s.t.} \quad & y_{i+1} = x_i + h \sum_{j=1}^{Q_i} \pi_j^{Q_i} f_{i-j+1}(x_{i-j+1}, u), \\
& y_{i+1}^c = x_i + h\ K_0^{Q_i+1} f_{i+1}(y_{i+1}, u) + h \sum_{j=1}^{Q_i} K_j^{Q_i+1} f_{i-j+1}(x_{i-j+1}, u), \\
& x_{i+1} = x_i + h \sum_{j=1}^{Q_i} \rho_j^{Q_i+2} f_{i-j+1}(x_{i-j+1}, u) + h\rho_0^{Q_i+2} f_{i+1}\left(x_{i+1}, u\right) + \\
& + h^2 \rho_0^{Q_i+2} D_x f_{i+1}(y_{i+1}^c, u).f_{i+1}(y_{i+1}^c, u), \quad i = 0, 1, 2, ..., k-1, \\
& x_0 = l(u), \\
& z_i = g_i\left(x_i, u\right), \ i = 0, 1, 2, ..., k.
\end{aligned}
$$

and the corresponding gradient formula follows:

$$
\begin{aligned}
& \nabla J(\bar{u}) = -\sum_{i=1}^{k} h \left[q_{i+1}^{c\mathrm{T}} K_0^{Q_i+1} D_u f(y_{i+1}, u) + p_{i+1}^{\mathrm{T}} \rho_0^{Q_i+2} D_u f(x_{i+1}, u) + \right. \\
& \left. p_{i+1}^{\mathrm{T}} h \rho_0^{Q_i+2} \left(D_{xu} f(y_{i+1}^c, u) f(y_{i+1}^c, u) + D_x f(y_{i+1}^c, u) D_u f(y_{i+1}^c, u) \right) \right] - \\
& - \sum_{r=0}^{k-1} \sum_{i=r}^{Q_r-1} h \left[q_{i+1}^{\mathrm{T}} \pi_{i+1-r}^{Q_i} + q_{i+1}^{c\mathrm{T}} K_{i+1-r}^{Q_i+1} + p_{i+1}^{\mathrm{T}} \rho_{i+1-r}^{Q_i+2} \right] D_u f\left(x_r, u\right) - \\
& - \sum_{i=0}^{k} \lambda_i^{\mathrm{T}}\ D_u^{\mathrm{T}} g\left(x_i, u\right) - p_0^{\mathrm{T}} D_u^{\mathrm{T}} l(\bar{u})
\end{aligned}
$$

where the multipliers p_i, q_i, q_i^c and λ_i are the solutions of the following simpler equations:

$$
\begin{aligned}
& p_i^{\mathrm{T}} \left[I - h.\rho_0^{Q_i+2} D_x f_i\left(x_i, u\right) \right] = q_{i+1}^{\mathrm{T}} + q_{i+1}^{c\mathrm{T}} + p_{i+1}^{\mathrm{T}} + \lambda_i^{\mathrm{T}}\ D_x g_i^{\mathrm{T}}\left(x_i, u\right) + \\
& \sum_{r=0}^{k-1} \sum_{j=1}^{Q_i-1} h \left[q_{j+1}^{\mathrm{T}} \pi_{j+i-1}^{Q_i} + q_{j+1}^{c\mathrm{T}} K_{j+i-1}^{Q_i+1} + p_{j+1}^{\mathrm{T}} \rho_{j+i-1}^{Q_i+2} \right] D_x f_i\left(x_i, u\right), \\
& \qquad\qquad i = 0, 1, ..., k-1
\end{aligned}
$$

$$
p_k^{\mathrm{T}} = \lambda_k^{\mathrm{T}} D_x g_k^{\mathrm{T}}(x_k, u)
$$

$$
q_{i+1}^{\mathrm{T}} = h \cdot q_{i+1}^{c\mathrm{T}} K_0^{Q_i+1} D_x f_{i+1}(y_{i+1}, u)\ ,\ i = 0, 1, ..., k-1,
$$

$$
q_{i+1}^{c\mathrm{T}} = h^2.p_{i+1}^{\mathrm{T}} \rho_0^{Q_i+2} \left(D_{xx} f_{i+1}(y_{i+1}^c, u).f_{i+1}(y_{i+1}^c, u) + D_x^2 f_{i+1}(y_{i+1}^c, u) \right), \quad i = 0, 1, ..., k-1,
$$

$$
\lambda_i^{\mathrm{T}} = -D_x \varphi(z_i, u)\ ,\ i = 0, 1, ..., k.
$$

2.4 Gradient for discrete models with automatic steplength control

All the common used software for numerical ODE integration contains an automatic step length selection, which takes into account the behavior of the right hand side during computation. The step length control is always based in an estimation of the local error and its selection is performed making this error estimate less than a given tolerance. At each integration step, several trial step lengths can be tested until a convenient one is found, i.e. one whose estimated error is small enough. In what follows, we propose a model for automatic steplength selection which, in our opinion, cover almost all the common used ideas for step control policy.

Suppose we are integrating the ODE system of the continuous problem (1). We denote by $h_{i,j}$ the $j-th$ trial steplength at integration step i. We assume that the maximum number of trials is equal to $j_{\max}$, and it is the same number for all integration steps. This is not a heavy assumption because, at the beginning of the computation, an initial steplength $h_{i,1}$, satisfying $h_{\min} < h_{i,1} < T$, is given and succesive decreasing lengths $h_{i,j}$ are tested until we have a success or a failure. In most cases, a failure ocurred when $h_{i,j}$ becomes less than a prescribed minimum length $h_{\min}$. Therefore, the maximum number of trials is bounded above and satisfies:

$$j_{\max} \leq Int\left[\frac{T}{h_{\min}}\right] + 1,$$

where $Int[x]$ denotes the greater integer less than or equal to x. On the other hand, in practice, the number of tested trials j_i is different at each iteration i and very frequently j_i is less than $j_{\max}$. But the assumption $j_i = j_{\max},\ i = 0, 1, ..., k-1$, has only theoretical meaning, as it can be seen later, and it simplifies further analysis. In addition we suppose the integration process is always successful because, otherwise, the computation is stopped and neither gradient formula nor convergence analysis have any sense. We call this the "always successful" assumption.

We denote by $e_{i,j} \in \Re$, the local error estimate corresponding to $h_{i,j}$. The accepted step at i is one of the $h_{i,j}$ and is denoted by h_i. Analogously, the corresponding accepted error, denoted by e_i, is one of the trial error estimates $e_{i,j}$. At each step i, the initial trial $h_{i,1}$ is constructed as a product of a function of the last accepted error times the last accepted step:

$$h_{i,1} = R(e_{i-1})h_{i-1},\ i = 1, 2, ..., k-1,$$

with the exception of $i = 0$, for which is supposed that it is a function of both, the initial state and the parameter vector:

$$h_{0,1} = Q(x_0, u).$$

After an unsuccessful trial step $h_{i,j}$, which means that its corresponding error estimate $e_{i,j}$ is not less than or equal to some given tolerance, say $\varepsilon > 0$, the next trial is chosen multiplying $h_{i,j}$ by a function of $e_{i,j}$:

$$h_{i,j+1} = r(e_{i,j})h_{i,j},\ i = 0, 1, ..., k-1;\ j = 1, 2, ..., j_{\max} - 1.$$

The error estimate can be obtained in several forms. For example, recomputing with half of the current steplength or using another scheme of higher order, etc... In most cases, it can be considered as a function of the vector $\mathbf{x}_i = (x_i, x_{i-1}, ..., x_{i-Q_i})$, the parameter vector u and the trial steplength $h_{i,j}$:

$$e_{i,j} = E(\mathbf{x}_i, u, h_{i,j}),\ i = 0, 1, ..., k-1;\ j = 1, 2, ..., j_{\max} - 1.$$

Finally, our model requires the explicit link between $h_{i,j}$ and h_i, and between $e_{i,j}$ and e_i. For this end, we define the following sequence of cartesian products of sets:

$$\begin{aligned} [0,\varepsilon]^{(1)} &= [0,\varepsilon], \\ [0,\varepsilon]^{(j+1)} &= [0,\varepsilon]^c \times [0,\varepsilon]^{(j)},\ j = 1, ..., j_{\max}, \end{aligned}$$

where $[0,\varepsilon]^c$ denotes the complement set $\Re \backslash [0,\varepsilon]$. it is easy to see that, under the "always successful" assumption, for each i there exists one and only one index $j_i \in \{1, ..., j_{\max}\}$ such that, the j_i-dimensional array $(e_{i,1}, ..., e_{i,j_i})$ belongs to the set $[0,\varepsilon]^{(j_i)}$. Note that the "accepted" index j_i corresponds exactly to the first occasion when the error $e_{i,j}$ is less than or equal to ε. Hence,

$$h_i = \sum_{j=1}^{j_{\max}} h_{i,j} \mathbf{1}_{[0,\varepsilon]^{(j)}} (e_{i,1}, ..., e_{i,j}),\ i = 0, 1, ..., k-1,$$

and

$$e_i = \sum_{j=1}^{j_{\max}} e_{i,j} \mathbf{1}_{[0,\varepsilon]^{(j)}} (e_{i,1}, ..., e_{i,j}),\ i = 0, 1, ..., k-1,$$

where $\mathbf{1}_A(x) = 1$, for $x \in A$, and $\mathbf{1}_A(x) = 0$, otherwise. Observe that:

$$\mathbf{1}_{[0,\varepsilon]^{(j)}} (e_{i,1}, ..., e_{i,j}) = \begin{cases} 1 & \text{if } e_{i,l} \notin [0,\varepsilon], \text{ for } l < j, \text{ and } e_{i,j} \in [0,\varepsilon], \\ 0 & \text{otherwise,} \end{cases}$$

and we can even write an exact formula for the accepted index at step i :

$$j_i = \sum_{j=1}^{j_{\max}} j\, \mathbf{1}_{[0,\varepsilon]^{(j)}} (e_{i,1}, ..., e_{i,j}),\ i = 0, 1, ..., k-1.$$

Then, one model for parameter estimation, using ODE integrators with automatic steplength control, is the following:

$$
\begin{array}{ll}
\min & J_{h,k}(u) = \sum\limits_{i=0}^{k} \tilde{\varphi}_i\left[z_i, \bar{z}_i\right], \\
\text{s.t.} & x_{i+1} = x_i + h_i F_i\left(x_i, x_{i-1}, ..., x_{i+1-Q_i}; u\right),\ i = 0, 1, 2, ..., k-1, \\
& x_0 = l(u), \\
& h_{0,1} = Q(x_0, u), \\
& h_{i,j+1} = r(e_{i,j}) h_{i,j},\ i = 0, 1, ..., k-1;\ j = 1, 2, ..., j_{\max} - 1, \\
& e_{i,j} = E(\mathbf{x}_i, u, h_{i,j}),\ i = 0, 1, ..., k-1;\ j = 1, 2, ..., j_{\max} - 1, \\
& e_i = e_{j_i} = \sum_{j=1}^{j_{\max}} e_{i,j} \mathbf{1}_{[0,\varepsilon]^{(j)}}\left(e_{i,1}, ..., e_{i,j}\right),\ i = 0, 1, ..., k-1, \\
& h_{i,1} = R(e_{i-1}) h_{i-1},\ i = 1, 2, ..., k-1, \\
& h_i = h_{j_i} = \sum_{j=1}^{j_{\max}} h_{i,j} \mathbf{1}_{[0,\varepsilon]^{(j)}}\left(e_{i,1}, ..., e_{i,j}\right),\ i = 0, 1, ..., k-1, \\
& z_i = g_i\left(x_i, u\right),\ i = 0, 1, 2, ..., k.
\end{array}
\tag{55}
$$

Here, all the vectors $\mathbf{x} = (x_0, ..., x_k), \boldsymbol{\eta} = (e_{0,1}, ..., e_{k-1,j_{\max}}), \mathbf{e} = (e_0, ..., e_{k-1}), \mathbf{H} = (h_{0,1}, ..., h_{k-1,j_{\max}})$, $\mathbf{h} = (h_0, ..., h_{k-1})$ enter as "state" variables and u is the unknown parameter vector. Assuming continuous differentiability for all the functions $\tilde{\varphi}_i, F_i, l, Q, r, E, R, g_i$, with respect to their arguments, it is not difficult to prove that Lemma 1 can be applied again. Then, the gradient $\nabla J_{h,k}(u)$ can be computed following the same algorithm given above. We shall show only the landmarks of the gradient calculation, in adjoint form, and leave the details to the reader.

Keeping the same notation as before, the Lagrangian function can be written as follows:

$$
\begin{array}{l}
\mathcal{L}(\mathbf{x}, \boldsymbol{\eta}, \mathbf{e}, \mathbf{H}, \mathbf{h}, u) = \\
= \sum\limits_{i=0}^{k} \Phi_i(x_i, u) + \sum\limits_{i=0}^{k-1} p_{i+1}^{\mathrm{T}}\left[x_{i+1} - x_i - h_i F_i\left(x_i, x_{i-1}, ..., x_{i+1-Q_i}; u\right)\right] + p_0^{\mathrm{T}}[x_0 - l(u)] + \\
+ \alpha_{0,1}\left[h_{0,1} - Q(x_0, u)\right] + \sum\limits_{i=0}^{k-1} \sum\limits_{j=1}^{j_{\max}} \alpha_{i,j+1}\left[h_{i,j+1} - r(e_{i,j}) h_{i,j}\right] + \\
+ \sum\limits_{i=0}^{k-1} \sum\limits_{j=1}^{j_{\max}} \gamma_{i,j}\left[e_{i,j} - E(\mathbf{x}_i, u, h_{i,j})\right] + \sum\limits_{i=0}^{k-1} \beta_i\left[e_i - \sum_{j=1}^{j_{\max}} e_{i,j} \mathbf{1}_{[0,\varepsilon]^{(j)}}\left(e_{i,1}, ..., e_{i,j}\right)\right] + \\
+ \sum\limits_{i=0}^{k-1} \alpha_{i,1}\left[h_{i,1} - R(e_{i-1}) h_{i-1}\right] + \sum\limits_{i=0}^{k-1} \delta_i\left[h_i - \sum_{j=1}^{j_{\max}} h_{i,j} \mathbf{1}_{[0,\varepsilon]^{(j)}}\left(e_{i,1}, ..., e_{i,j}\right)\right],
\end{array}
$$

where $p_i, \alpha_{i,j}, \gamma_{i,j}, \beta_i$ and δ_i are the adjoint variables. Computing derivatives with respect to each one of the state variables, and equating to zero, we obtain the following adjoint (backward) system of difference equations:

$$
\begin{array}{l}
p_k^{\mathrm{T}} = -D_x \Phi_k(x_k, u), \\
p_t^{\mathrm{T}} = p_{t+1}^{\mathrm{T}} + \sum\limits_{i=t}^{k-1} p_{i+1}^{\mathrm{T}} h_i D_{x_t} F_i(\mathbf{x}_i, u) - D_x \Phi_t(x_t, u) + \\
\qquad + \sum\limits_{i=t}^{k-1} \sum\limits_{j=1}^{j_{\max}} \gamma_{i,j} D_{x_t} E(\mathbf{x}_i, u, h_{i,j}) + \alpha_{0,1} D_x Q(x_0, u) \mathbf{1}_{\{0\}}(t),\ t = k-1, ..., 0,
\end{array}
\tag{56}
$$

$$\begin{aligned}
&\delta_{k-1} = 0,\\
&\beta_{k-1} = p_k^{\mathrm{T}} F_{k-1}(\mathbf{x}_{k-1}, u),\\
&\gamma_{t,j_{\max}} = \delta_t\, \mathbf{1}_{[0,\varepsilon]^{(j_{\max})}}\left(e_{t,1}, ..., e_{t,j_{\max}}\right),\ t = k-1, ..., 0,\\
&\alpha_{t,j_{\max}} = \gamma_{t,j_{\max}} D_h E(\mathbf{x}_t, u, h_{t,j_{\max}}) + \beta_t\, \mathbf{1}_{[0,\varepsilon]^{(j_{\max})}}\left(e_{t,1}, ..., e_{t,j_{\max}}\right),\ t = k-1, ..., 0,\\
&\gamma_{t,s} = \alpha_{t,s+1} D_e r(e_{t,s}) h_{t,s} + \delta_t\, \mathbf{1}_{[0,\varepsilon]^{(s)}}\left(e_{t,1}, ..., e_{t,s}\right),\ s = j_{\max}-1, ..., 1;\ t = k-1, ..., 0,\\
&\alpha_{t,s} = \alpha_{t,s+1} r(e_{t,s}) + \gamma_{t,s} D_h E(\mathbf{x}_t, u, h_{t,s}),\ s = j_{\max}-1, ..., 1;\ t = k-1, ..., 0,\\
&\beta_t = p_t^{\mathrm{T}} F_t(\mathbf{x}_t, u) + \alpha_{t+1,1} R(e_{t+1}),\ t = k-2, ..., 0,\\
&\delta_t = \alpha_{t+1,1} D_e R(e_{t+1}),\quad t = k-2, ..., 0.
\end{aligned} \tag{57}$$

If j_t denotes, as before, the index corresponding to the accepted error and steplength, we note that:

$$\mathbf{1}_{[0,\varepsilon]^{(s)}}\left(e_{t,1}, ..., e_{t,s}\right) = 0,\ \text{for } s \neq j_t,$$

and this implies $\gamma_{t,s} = \alpha_{t,s} = 0$, for all t and all $s \geq j_t + 1$. For this reason, the number $j_{\max}$ can be changed by j_t in all the expressions of (57), and the characteristic functions won't appear. Then, (57) takes the form:

$$\begin{aligned}
&\delta_{k-1} = 0,\\
&\beta_{k-1} = p_k^{\mathrm{T}} F_{k-1}(\mathbf{x}_{k-1}, u),\\
&\gamma_{t,j_t} = \delta_t,\ t = k-1, ..., 0,\\
&\alpha_{t,j_t} = \gamma_{t,j_t} D_h E(\mathbf{x}_t, u, h_{t,j_t}) + \beta_t,\ t = k-1, ..., 0,\\
&\gamma_{t,s} = \alpha_{t,s+1} D_e r(e_{t,s}) h_{t,s},\ s = j_t-1, ..., 1;\ t = k-1, ..., 0,\\
&\alpha_{t,s} = \alpha_{t,s+1} r(e_{t,s}) + \gamma_{t,s} D_h E(\mathbf{x}_t, u, h_{t,s}),\ s = j_t-1, ..., 1;\ t = k-1, ..., 0,\\
&\beta_t = p_t^{\mathrm{T}} F_t(\mathbf{x}_t, u) + \alpha_{t+1,1} R(e_{t+1}),\ t = k-2, ..., 0,\\
&\delta_t = \alpha_{t+1,1} D_e R(e_{t+1}),\quad t = k-2, ..., 0.
\end{aligned} \tag{58}$$

Finally, the Lagrangian function derivative, with respect to u, gives the adjoint gradient formula:

$$\begin{aligned}
\nabla J_{h,k}(u) = &\sum_{i=0}^{k} D_u \Phi_i(x_i, u) - \sum_{i=0}^{k-1} p_{i+1}^{\mathrm{T}} h_i D_u F_i\left(\mathbf{x}_i, u\right) - p_0^{\mathrm{T}} D_u l(u) -\\
&-\alpha_{0,1} D_u Q(x_0, u) - \sum_{i=0}^{k-1} \sum_{j=1}^{j_i} \gamma_{i,j} D_u E(\mathbf{x}_i, u, h_{i,j}).
\end{aligned} \tag{59}$$

Remark 3. If the function Q does not depend on u, and the functions R and r have derivatives $D_u R, D_u r$ equal to zero then, $D_u Q = 0$, and $\gamma_{i,j} = 0$, for all i, j. Then, the formula for the gradient $\nabla J_{h,k}(u)$ of the model with automatic steplength selection is identical to the formula for the gradient $\nabla J_k(u)$ of the model with fixed steplengths. The only difference is that in the later case we know the steplengths *before* the integration and in the former case *after* the integration. The assumptions for Q, R and r are not so strongs as they look like. Frequently, piecewise constant functions R, r and a constant function Q are used. A constant Q means that a fixed step, say $h_{0,1} = 10^{-2}$, is always taken at the beginning,. Piecewise constant r or R means that the factors for step changing remain constant when the local error estimate lies in certain fixed intervals.

We collect all the above results in the following:

Theorem 3: Let be the continuous parameter estimation problem (1). Then, under "always succesful assumption", the problem (55) can be considered the discrete model we obtained if the integration is accomplished using an automatic steplength selection and a multi-step scheme. Furthermore, the gradient (adjoint form) of $J_{h,k}(u)$ is given by (59), where the multipliers are the solution of the adjoint system (56),(58). In addition, if Q does not depend on u and if R, r are piecewise constant functions, then $\nabla J_{h,k}(u)$ is equal to $\nabla J_k(u)$, for all $u \in \Re^m$.

Remark 4. It is also possible to increase the model with an usual order selection policy, i.e. adding order selection equations like:

$$\begin{aligned} Q_{i,s} &= s \sum_{j=0}^{j_{\max}} \mathbf{1}_{[0,\varepsilon]^{(j)}} \left(e^s_{i,1}, ..., e^s_{i,j} \right), \\ Q_{i+1} &= \min \left\{ \min_{Q_{i,s}>0} Q_{i,s},\ Q_{\max} - 1 \right\}, \end{aligned}$$

where the local error estimation $e^s_{i,j}$ at step i, would depend not only on the steplength h_{ij}, but also on the order s. But this is irrelevant, since the derivatives of $Q_{i,s}$ with respect to x and u are zero and the formula for the gradient remains unchanged.

2.5 Hessian for continuous and discrete models

If we assume that φ_i, f, g and l are continuous and twice continuously differentiable with respect to (x, u), then J, J_k and $J_{h,k}$ are also twice continuously differentiable and it is only a matter of (tedious) calculus the computation of the Hessian functions $\nabla^2 J(u), \nabla^2 J_k(u), \nabla^2 J_{h,k}(u)$, following the algorithms given in Lemma 1.

For example, the model for the continuous gradient $\nabla J(u)$ (using sensitivity form) is given by:

$$\nabla J(u) = \sum_{i=1}^{s} \left[D_x \Phi_i(x(\tau_i), u) M(\tau_i) + D_u \Phi_i(x(\tau_i), u) \right],$$

where:

$$\begin{aligned} &\dot{x}(t) = f\left(x(t), u, t\right),\ t \in [0, T], \\ &x(0) = l(u), \\ &\dot{M}(t) = f_x(x(t), u, t) M(t) + f_u(x(t), u, t),\ t \in [0, T], \\ &M(0) = D_u l(u). \end{aligned}$$

A direct computation gives the sensitivity form of the continuous Hessian:

$$\nabla^2 J(u) = \sum_{i=1}^{s} \left[M^{\mathrm{T}}(\tau_i) D^2_{xx}\Phi_i(x(\tau_i),u)M(\tau_i) + D_x\Phi_i(x(\tau_i),u)\mathcal{M}(\tau_i) + \right.$$
$$\left. +2D^2_{ux}\Phi_i(x(\tau_i),u)M(\tau_i) + D^2_{uu}\Phi_i(x(\tau_i),u) \right],$$

where $M(t) = D_u x(t)$ is the $n\times m-$matrix, solution of the original sensitivity system, and $\mathcal{M}(t) = D_u M(t)$ is a $n \times m \times m-$tensor, solution of the following additional sensitivity system:

$$\begin{aligned} \frac{d\mathcal{M}(t)}{dt} = \quad & M^{\mathrm{T}}(t) f_{xx}(x(t),u,t)M(t) + f_x(x(t),u,t)\mathcal{M}(t) + \\ & +2f_{ux}(x(t),u,t)M(t) + f_{uu}(x(t),u,t),\ t\in[0,T], \\ \mathcal{M}(0) = \quad & D^2_{uu} l(u). \end{aligned}$$

As another example, we write the model for the discrete gradient $\nabla J_k(u)$ (using sensitivity form):

$$\nabla J_k(\bar{u}) = \sum_{i=0}^{k} \left[D_x\Phi_i(x_i,u)M_i + D_u\Phi_i(x_i,u) \right]$$

where:

$$\begin{aligned} & x_{i+1} = x_i + h_i F_i\left(\mathbf{x}_i,u\right),\ i=0,1,2,...,k-1, \\ & x_0 = l(u), \\ & M_{i+1} = M_i + h_i \left[\sum_{j=1}^{Q_i} D_{x_{i-j+1}} F_i(\bar{\mathbf{x}}_i,\bar{u}) M_{i-j+1} + D_u F_i(\bar{\mathbf{x}}_i,\bar{u}) \right],\ i=0,1,...,k-1, \\ & M_0 = D_u l(u), \qquad Q_i = \min\left\{ i+1, Q_{\max}-1 \right\}. \end{aligned}$$

Then, the Lagrangian is written as follows:

$$\begin{aligned} & \mathcal{L}(\mathbf{x},\mathbf{M},u) = \textstyle\sum_{i=0}^{k} \left[D_x\Phi_i(x_i,u)M_i + D_u\Phi_i(x_i,u)\right] + \\ & + \textstyle\sum_{i=0}^{k-1} \left[x_{i+1} - x_i - h_i F_i\left(\mathbf{x}_i,u\right)\right]^{\mathrm{T}} \mathcal{M}_{i+1} + \left[x_0 - l(u)\right]^{\mathrm{T}} \mathcal{M}_0 + \\ & + \textstyle\sum_{i=0}^{k-1} \pi_{i+1}^{\mathrm{T}} \left[M_{i+1} - M_i - h_i \left(\sum_{j=1}^{Q_i} D_{x_{i-j+1}} F_i(\bar{\mathbf{x}}_i,\bar{u}) M_{i-j+1} + D_u F_i(\bar{\mathbf{x}}_i,\bar{u}) \right) \right] + \\ & + \pi_0^{\mathrm{T}} \left[M_0 - D_u l(u) \right], \end{aligned}$$

where $\mathbf{x} = (x_0, ..., x_k)$ and $\mathbf{M} = (M_0, ..., M_k)$ enter as state variables and $\mathcal{M}_i, \pi_i$ are adjoint $n \times m-$matrices and $n-$vectors respectively.

Following the same routine, we compute derivatives with respect to each one of the state variables, equate them to zero and, after convenient rearrangements, we obtain the following adjoint system:

$$
\begin{aligned}
&\pi_k^{\mathrm{T}} = -D_x\Phi_k(x_k,u),\\
&\pi_i^{\mathrm{T}} = \pi_{i+1}^{\mathrm{T}} + \sum_{j=i}^{\tilde{Q}_i} h_j\pi_{j+1}^{\mathrm{T}} D_{x_i}F_j(\bar{\mathbf{x}}_j,u) - D_x\Phi_i(x_i,u),\ i=\overline{0,k-1},\\
&\mathcal{M}_k = -D_{xx}^2\Phi_k(x_k,u)M_k - D_{ux}^2\Phi_k(x_k,u),\\
&\mathcal{M}_i = \mathcal{M}_{i+1} + \sum_{j=i}^{k-1} h_j D_{x_i}F_j(\bar{\mathbf{x}}_j,u)\mathcal{M}_{j+1} +\\
&+\sum_{r=i}^{k-1}\left[\left(\sum_{j=r}^{\tilde{Q}_r}\pi_{j+1}^{\mathrm{T}} h_j D_{x_j x_i}^2 F_j(\bar{\mathbf{x}}_j,u)\right)M_r + \pi_{r+1}^{\mathrm{T}} h_r D_{ux_i}^2 F_r(\bar{\mathbf{x}}_r,u)\right] -\\
&-D_{xx}^2\Phi_i(x_i,u)M_i - D_{ux}^2\Phi_i(x_i,u),
\end{aligned}
$$

where $\tilde{Q}_i = \min\{i+Q_{\max}-1,k\}$.

The derivative of $\mathcal{L}$ with respect to u gives the following adjoint form for the discrete Hessian:

$$
\begin{aligned}
&\nabla^2 J_k(u) = \textstyle\sum_{i=0}^{k}\left[D_{xu}^2\Phi_i(x_i,u)M_i + D_{uu}^2\Phi_i(x_i,u)\right] -\\
&-\sum_{i=0}^{k-1} h_i D_u F_i^{\mathrm{T}}(\bar{\mathbf{x}}_i,u)\mathcal{M}_{i+1} - D_u l^{\mathrm{T}}(u)\mathcal{M}_0 - \pi_0^{\mathrm{T}} D_{uu} l(u) -\\
&-\sum_{r=0}^{k-1}\left[\left(\sum_{j=r}^{\tilde{Q}_r}\pi_{j+1}^{\mathrm{T}} h_j D_{x_j u}^2 F_j(\bar{\mathbf{x}}_j,u)\right)M_r + \pi_{r+1}^{\mathrm{T}} h_r D_{uu}^2 F(\bar{\mathbf{x}}_r,u)\right].
\end{aligned}
$$

3 Conclusions

In the present paper we stated and proved some fundamental results related with inverse problems in ODE modelling which we resume as follows:

1) We proposed some kind of "direct approach " for the solution of the inverse problem, substituting the ordinary differential equation by a difference scheme. This is a general approach and can be used as well for other dynamical model,

2) We strongly recommended to use exact formulas for the gradients of the discrete approximation schemes instead of using a finite difference approach. For this reason we gave also a general method for their calculation,

3) We developed complete proofs of the gradient formulas for the continuous inverse problem and for several of its discrete approximations, using explicit and implicit multistep schemes.

4) As far as we know, the model and results developed for the case of integration by automatic steplength control are completely new. We also included some formulas for the Hessian matrices.

References

[1] Bard, J., Nonlinear Parameter Estimation, Academic Press Inc., 1974.

[2] Birgin, E.G., Y.G. Evtushenko, Automatic differentiation and spectral projected gradient methods for optimal control problems. Optimization Methods and Software 10 (1998), pp. 125-146.

[3] Bock, H.G. Recent advances in parameter identification techniques for ODE.In Modern methods of Optimization, Proceedings of the Summer School, Bayreuth, 1990, W. Krabs and J. Zowe Eds. Springer Verlag, New York, 1991.

[4] Enright, W.H. Second derivate multi-step methods for stiff ordinary differential equations. *Numerical Analysis,* Vol. 2, No. 2, 1976, pp. 351-362.

[5] Evtushenko, Y., Computation of exact gradients in distributed dynamic systems. Optimization Methods and Software 9 (1998), pp. 45-76.

[6] Gear, W.C. Numerical Initial Value Problems in O.D.E.Prantice-Hall. New Jersey, 1971.

[7] Gómez Fernández J.A.Curso de Cálculo Variacional y Control Optimal.(Notas para un libro de texto).*Ediciones Universidad de la Habana.*Tomo 1, 1995.

[8] Gómez, J.A., Marrero, A. ODEPARES: A nonlinear ODE parameter estimation toolbox for MATLAB. Research Report of ICIMAF, 1999 (In preparation).

[9] Hairer, E., G. Wanner and S.P. Norcett, Solving Ordinary Differential Equations I, II, Springer Verlag, New York, 1996.

[10] Henrici, P. Discrete variable methods in Ordinary Differential Equations, John Wiley and Sons, New York, 1961.

[11] Kolmogorov A.N.,Fomin S.V. Elementos de la Teoría de Funciones y del Análisis Funcional. Tercera Edición. *Editorial Mir.*Moscú, 1978.

[12] Hintze, J.L., Software SOLO, version 2.0, BMPD Statistical Software 1440, Los Angeles, Copyright 1998.

[13] Jacquez, J., P. Greif. Numerical parameter identifiability and estimability: Integrating identifiability, estimability and optimal sampling design, *Mathematical Biosciences,* 77, 1985, pp. 201-227.

[14] Marrero A.; J.A. Gómez. Estimación de parámetros en modelos dinámicos no lineales con empleo de métodos numéricos. *Investigación Operacional*, Vol. XI , No. 2 ,1990, pp. 101-118.

[15] Marrero A.; J.A. Gómez. Problemas de estimación de parámetros en modelos dinámicos no lineales. *Ciencias Matemáticas*, Vol. 15, No. 1, 1994, pp. 41-54.

[16] Marrero A; J.A. Gómez, Experiencias numéricas con métodos multipaso-multiderivadas para el problema de estimación de parámetros en modelos dinámicos. *Investigación Operacional,* Vol. 18, No. 1-3 ,1997, pp. 3-10.

[17] Marrero, A., J.A. Gómez. An Approach for the Numerical Solution of Parameter Estimation Problems in Nonlinear ODE Models. Proceedings CIMAF'99. Ciudad de la Habana. 1999.

[18] MINITAB for Windows, Release 11.2, MINITAB Inc. Support 814-231-2682, Copyright 1996.

[19] Mitsui, Taketomo. The initial value adjusting method for problems of the least square type of ODE, Publ. RIMS, Kyoto University, 16, 1980, pp. 785-810.

[20] Nowak, U., P. Deuflhard. Towards parameter identification for large chemical reaction systems. In Numerical Treatment of inverse problems in Differential and Integral Equations. Proceedings of an International Workshop, Heidelberg 1982, P. Deuflhard and E. Hairer Eds., Birkhauser, Boston, 1983.

[21] Polak, E., Computational Methods in Optimization: A unified approach. Academic Press, New York (1971).

[22] Schloder, J., G. Bock. Identification of rate constants in bistable chemical reactions. In Modern methods of Optimization, Proceedings of the Summer School, Bayreuth, 1990, W. Krabs and J. Zowe Eds. Springer Verlag, New York, 1991.

[23] Shampine, L.F., M.K. Gordon, Computer Solution of Ordinary Differential Equations. The Initial Value Problem. W.H. Freeman and Company, San Francisco, 1991.

[24] Stoer J., R.Burlisch. Introduction to Numerical Analysis. Second Edition. Springer Verlag, 1993.