Time Solution of Differential Algebraic Equations (DAE) by Implicit Integration

Soumyabrata Talukder

October 3, 2020

1 Semi-explicit Index-1 DAE

To formulate the implicit integration scheme, we consider the semi-explicit index-1 DAE of the form:

$$\begin{aligned} \dot{x} &= f(x,y), \\ \vec{0} &= g(x,y), \end{aligned} \tag{1}$$

where $x \in \mathbb{R}^n$ and $y \in \mathbb{R}^m$ are the vectors of the state and algebraic variables respectively, and $f : \mathbb{R}^{m+n} \to \mathbb{R}^n$, $g : \mathbb{R}^{m+n} \to \mathbb{R}^m$ are (possibly) non-linear C^1 maps defining the vector-field and the algebraic constraints of a system respectively.

Eq. (1) is called semi-explicit since x and y are coupled in the maps f and g. The explicit form would be the special case of (1), where the following decompositions of f and g are feasible:

$$\begin{aligned} f(x,y) &= f_1(x) + f_2(y), \\ g(x,y) &= g_1(x) + g_2(y). \end{aligned} \tag{2}$$

On the other hand, the implicit form of DAE, the most general form, is given by:

$$f'(\dot{x}, x) = \vec{0}, \tag{3}$$

where $f' : \mathbb{R}^{2n} \to \mathbb{R}^n$ is a (possibly) nonlinear map. The term *index* corresponds to the *differentiation index*. Index-1 of (1) refers to the fact that

$\dot{y}$ can be determined using the first-order differentials of g as following:

$$\dot{y} = -g_y^{-1} g_x \dot{x}, \tag{4}$$

if g_y is non-singular. Here g_x and g_y are the jacobians of g w.r.t x and y respectively.

2 Existence of Unique Time Solution

The *uniqueness* and *existence* theorems together help to affirm the existence of time-solution of (1). According to the uniqueness theorem, non-singularity of g_y at a point $(\bar{x}, \bar{y})$ implies that a unique map $y = h(x)$ ($h : \mathbb{R}^n \to \mathbb{R}^m$) exists over an infinitesimally small neighborhood of $(\bar{x}, \bar{y})$. If such unique map exists, then a unique time solution of (1) exists in the same neighborhood, as per the existence theorem. Hence, a necessary condition of existence of a unique time solution of (1) is the non-singularity of g_y along the solution trajectory.

3 Formulation

The goal is to compute the time solution of (1) numerically, by computing the sequence of pairs (x_i,y_i) for the discrete time instants $i > 0$, given the value of (x_0, y_0).

For a given time-step $\Delta t > 0$, the state vector at the i^{th} and the $(i+1)^{th}$ discrete time instants are related by the following approximation:

$$x_{i+1} = x_i + \Delta t \tilde{f}(x_i, y_i, \Delta x, \Delta y), \tag{5}$$

where $\tilde{f}(x_i, y_i, \Delta x, \Delta y)$ denotes an estimate of $f(x_i, y_i)$ constant over the time-step Δt, and $\Delta x \in \mathbb{R}^n, \Delta y \in \mathbb{R}^m$ denotes the change in x, y through this time step. Given the values of x_i, y_i, one can define a function $F_i : \mathbb{R}^{m+n} \to \mathbb{R}^n$ as follows:

$$F_i(\Delta x, \Delta y) := \Delta x - \Delta t \tilde{f}(x_i, y_i, \Delta x, \Delta y), \tag{6}$$

which is simply the estimation error in (5) observed at the i^{th} discrete instant for the next time-step. For an ideal estimate $\tilde{f}$, we must have:

$$F_i(\Delta x, \Delta y) = \vec{0}. \tag{7}$$

Also, $\Delta x, \Delta y$ must be such that the algebraic constraints in (1) are satisfied at every time step, i.e. for all $i \geq 0$ we must have:

$$g(x_i + \Delta x, y_i + \Delta y) = \vec{0}. \tag{8}$$

which for the given the values of x_i, y_i, can be written more compactly as:

$$G_i(\Delta x, \Delta y) = \vec{0}, \tag{9}$$

that absorbs the constants x_i, y_i.

Finally, the problem is to find $\Delta x, \Delta y$ successively for every discrete time instant $i \geq 0$ and for the given quantities x_i, y_i and Δt, such that (7),(9) are satisfied up to a given accuracy, say $\epsilon > 0$. An efficient way to numerically compute the $m+n$ variables in $\Delta x, \Delta y$, solving the $m+n$ nonlinear equations in (7),(9), is to employ the popular Newton-Rhapson (NR) method.

4 Numerical Solution

Using NR method, we can solve (7),(9) for $\Delta x, \Delta y$ by solving the following iteratively:

$$\begin{bmatrix} \Delta x^{j+1} \\ \Delta y^{j+1} \end{bmatrix} = \begin{bmatrix} \Delta x^{j} \\ \Delta y^{j} \end{bmatrix} - \begin{bmatrix} F^j_{i\Delta x} & F^j_{i\Delta y} \\ G^j_{i\Delta x} & G^j_{i\Delta y} \end{bmatrix}^{-1} \begin{bmatrix} F^j_i \\ G^j_i \end{bmatrix}, \tag{10}$$

where $j \geq 0$ denotes the iteration number; $F^j_{i\Delta x}$,$F^j_{i\Delta y}$ (resp. $G^j_{i\Delta x}$,$G^j_{i\Delta y}$) denotes the jacobians of F_i (resp. G_i) w.r.t $\Delta x, \Delta y$ respectively, evaluated at $\Delta x = \Delta x^j, \Delta y = \Delta y^j$; and F^j_i, G^j_i respectively, denotes the evaluations of functions F_i, G_i at $\Delta x = \Delta x^j$, $\Delta y = \Delta y^j$.

The convergence of the NR method is sensitive to the distance of the initial guess (i.e. $\Delta x^0, \Delta y^0$ in our case) from the true solution. If the choice of Δt is small, the corresponding true values of Δx, Δy are also expected to be small. Hence, the initialization $\Delta x^0 = \vec{0}, \Delta y^0 = \vec{0}$ is usually a good choice in case (1) is dynamically stable. For each discrete instant $i \geq 0$, the iterative method defined in (10) is continued until the norm $\left\| \begin{bmatrix} F^j_i \\ G^j_i \end{bmatrix} \right\|$ is lower than ϵ.

The jacobians $G^j_{i\Delta x}$,$G^j_{i\Delta y}$ are simply given by:

$$\begin{aligned} G^j_{i\Delta x} &= g_x(x_i + \Delta x^j, y_i + \Delta y^j) \\ G^j_{i\Delta y} &= g_y(x_i + \Delta x^j, y_i + \Delta y^j). \end{aligned} \tag{11}$$

Note that the jacobians $F^j_{i\Delta x}$,$F^j_{i\Delta y}$ depend on the choice of the vector-field estimate $\tilde{f}$. Next we derive these jacobians for the two popular implicit methods: (i) Trapezoidal and (ii) Backward Euler.

4.1 Trapezoidal Method

In trapezoidal method, $\tilde{f}$ is estimated as follows:

$$\tilde{f}(x_i, y_i, \Delta x, \Delta y) = 0.5(f(x_i, y_i) + f(x_i + \Delta x, y_i + \Delta y)), \tag{12}$$

and accordingly, (6) can be rewritten as:

$$F_i(\Delta x, \Delta y) = \Delta x - 0.5\Delta t(f(x_i, y_i) + f(x_i + \Delta x, y_i + \Delta y)). \tag{13}$$

So, one can compute the jacobians $F^j_{i\Delta x}$ and $F^j_{i\Delta y}$ from (17) as following:

$$\begin{aligned} F^j_{i\Delta x} &= \mathbb{I}_n - 0.5.\Delta t.f_x(x_i + \Delta x^j, y_i + \Delta y^j), \\ F^j_{i\Delta y} &= -0.5.\Delta t.f_y(x_i + \Delta x^j, y_i + \Delta y^j), \end{aligned} \tag{14}$$

where $\mathbb{I}_n$ denotes the $n \times n$ identity matrix.

4.2 Backward Euler Method

In backward Euler method, $\tilde{f}$ is estimated as follows:

$$\tilde{f}(x_i, y_i, \Delta x, \Delta y) = f(x_i + \Delta x, y_i + \Delta y), \tag{15}$$

and accordingly, (6) can be rewritten as:

$$F_i(\Delta x, \Delta y) = \Delta x - \Delta t.f(x_i + \Delta x, y_i + \Delta y). \tag{16}$$

So, one can compute the jacobians $F^j_{i\Delta x}$ and $F^j_{i\Delta y}$ from (17) as following:

$$\begin{aligned} F^j_{i\Delta x} &= \mathbb{I}_n - \Delta t.f_x(x_i + \Delta x^j, y_i + \Delta y^j), \\ F^j_{i\Delta y} &= -\Delta t.f_y(x_i + \Delta x^j, y_i + \Delta y^j). \end{aligned} \tag{17}$$