In practice the highest derivative occurs linearly, i.e., these equations can be written as

\[ E(y)\dot{y} = f(y, t). \]  

(1.48)

We have to distinguish two special cases

- \[ \forall y : \det E(y) \neq 0 \]
- \[ \forall y : \det E(y) = 0. \]

In the first case, the differential equation can be easily transformed into explicit form. In this section, we will consider the second case. We will call this type of equations **differential algebraic equations** DAEs. This notation is motivated, when \( E \) has the particular form

\[ E(y) = \begin{pmatrix} M(y) & 0 \\ 0 & 0 \end{pmatrix} \]

with \( M(y) \) being a square, nonsingular matrix.

Then, the system can be written as a system of coupled differential and algebraic equations

\[ M(y)\dot{y}_1 = f_1(y_1, y_2) \]  
\[ 0 = f_2(y_1, y_2). \]  

(1.49a) 
(1.49b)

Examples for this kind of systems occur in

- mechanics, where the system describes a constrained dynamical system,
- electrical networks, where the algebraic equation describes a current balance (Kirchhoff’s law),
- chemical engineering, where the algebraic equation describes mass balances.

**Example 32** A mathematical pendulum can be described in Cartesian coordinates in stead of the standard angular coordinates. This gives the following equations

\[ \dot{p}_1 = v_1 \]  
\[ \dot{p}_2 = v_2 \]  
\[ \dot{v}_1 = -\lambda p_1 \]  
\[ \dot{v}_2 = -\lambda p_2 - g \]  
\[ 0 = p_1^2 + p_2^2 - 1. \]  

(1.50a) 
(1.50b) 
(1.50c) 
(1.50d) 
(1.50e)
Herein we assume the mass of the pendulum and its length being one. Comparing this system with Eq. (1.49) gives

\[
y_1 := \begin{pmatrix} p_1 \\ p_2 \\ v_1 \\ v_2 \end{pmatrix}
\]

and \( M = I_2 \).

Often DAEs occur in so-called Hessenberg-form. In this case the components of the independent variable \( y \) can be grouped in such a way, that Eq. (1.49) can be written as

\[
\dot{y}_1 = f_1(y_1, \ldots, y_{r-1}, y_r, t) \\
\dot{y}_2 = f_2(y_1, \ldots, y_{r-1}, t) \\
\dot{y}_3 = f_3(y_2, \ldots, y_{r-1}, t) \\
\vdots \\
\dot{y}_{r-1} = f_{r-1}(y_{r-2}, y_{r-1}, t) \\
0 = f_r(y_{r-1}, t)
\]

Note that the set of variables \( y_r \) does not occur differentiated and we have therefore a system of coupled differential and algebraic equations. We will see that by subsequent differentiation these so-called algebraic system variables can be eliminated and an explicit ODE system will be obtained.

To simplify things for this introduction we explain this process by considering a linear DAE in Hessenberg form:

\[
\dot{y}_1 = A_{1,1}y_1 + y_{r-1} + A_{1,r}y_r + b_1(t) \\
\dot{y}_2 = A_{2,1}y_1 + \ldots + A_{2,r-1}y_{r-1} + b_2(t) \\
\dot{y}_3 = A_{3,1}y_2 + \ldots + A_{3,r-1}y_{r-1} + b_3(t) \\
\vdots \\
\dot{y}_{r-1} = A_{r-1,1}y_{r-2} + A_{r-1,r-1}y_{r-1} + b_{r-1}(t) \\
0 = A_{r,r-1}y_{r-1} + b_r(t)
\]

By differentiating Eq. (1.52f) with respect to time we obtain

\[
0 = A_{r-1,r-1}\dot{y}_{r-1} + \dot{b}_r(t).
\]

Inserting in this equation Eq. (1.52e) gives then

\[
0 = A_{r-1,r-2}y_{r-2} + A_{r-1,r-1}y_{r-1} + b_{r-1}(t) + \dot{b}_r(t).
\]
This process of differentiating and inserting can be repeated until one reaches the first set of equations. One obtains an expression of the form

\[ 0 = A_{r,r-1}A_{r-1,r-2}y_{r-2} \cdots A_{2,1}A_{1,r}y_r + \sum_{i=2}^{r} b_i^{(i-1)}(t) + \ldots. \] (1.53)

If the leading matrix is non singular, i.e. if

\[ \det \left( A_{r,r-1}A_{r-1,r-2}y_{r-2} \cdots A_{2,1}A_{1,r} \right) \neq 0 \]

holds, then \( y_r \) can be expressed in terms of the remaining variables \( y_1, \ldots, y_{r-1} \) and the resulting system will be an explicit ODE.

**Definition 33** \( r \) is called the differentiation index of the DAE Eq. (1.52), if this system can be transformed by \( r - 1 \) differentiation steps into an explicit ODE.

If \( A_{r,r-1}A_{r-1,r-2}y_{r-2} \cdots A_{2,1}A_{1,r} \) is singular, there is no way to eliminate the algebraic variables \( y_r \) by further differentiations and the DAE is said to be a system of index infinity.

An important consequence of this transformation process is, that the time dependent input function \( b_r \) will be also differentiated \( r - 1 \) times. By interpreting this function as a perturbation to a system without this input, one intuitively realizes the fact, that perturbations influence the solution with the size of their derivatives up to order \( r - 1 \).

This statement can be formulated as a generalization of Theorem 5:

**Lemma 34** Consider the DAE

\[ y = f(y, z) \] (1.54)
\[ 0 = g(y) \] (1.55)

with the initial value \( y_0, z_0 \) and assume that it can be written in Hessenberg form and has index \( r \), furthermore let \( \hat{y}, \hat{z} \) be the solution of the perturbed DAE

\[ y = f(y, z) + \delta_1(t) \] (1.56a)
\[ 0 = g(y) + \delta_2(t) \] (1.56b)

with the same initial values. Then the influence of the perturbation is bounded by

\[ \| \hat{x}(t) - x(t) \| \leq C \left( \| \hat{x}(t_0) - x(t_0) \| + \max_{t_0 \leq \tau \leq t} \| \int_{t_0}^{\tau} \delta(\tau)d\tau \| + \right. \\
+ \max_{t_0 \leq \tau \leq t} \| \delta(\tau) \| + \cdots + \max_{t_0 \leq \tau \leq t} \| \delta^{(r-1)}(\tau) \| \)

with \( x = (y, z) \) and \( \delta = (\delta_1, \delta_2) \).
Often perturbations are caused by measurement errors, which might be small in magnitude, but with a large frequency (noise!). When computing the solution numerically such high frequent disturbances are already introduced by the round-off error and by local errors due to discretization. Thus, one expects numerical problems, when solving high index DAEs.

We leave it as an example, to show, that the pendulum equation can be formulated as a Hessenberg system and that this system has index-3. One can show generally, that all practically relevant mechanical systems have index-3. For most applications in electrical network simulation the index is one, sometimes two.

The index concept is also important to determine the degree of freedom one has, when choosing initial systems for a DAE.

This can be seen already from the following simple examples

**Example 35**

\[
\begin{align*}
\dot{y}_2 &= y_1 \\
0 &= y_1 - b_2(t)
\end{align*}
\]

with \( b_2 \) being a sufficiently smooth function. The system has index-1 and the solution of this problem is given by

\[
y_1(t) = b_2(t) \quad \text{and} \quad y_2(t) = y_2(t_0) + \int_{t_0}^t b_2(s)ds.
\]

(1.57)

If \( y_1(t_0) \neq b_2(t_0) \), there exists no smooth solution.

This example shows what one might have expected. The freedom in selecting initial conditions is limited by the constraint. Clearly, the initial conditions must meet the constraints.

The second example shows is a simple index-2 DAE

**Example 36**

\[
\begin{align*}
\dot{y}_2 &= y_1 - b_1(t) \\
0 &= y_2 - b_2(t)
\end{align*}
\]

with \( b_1 \) and \( b_2 \) being sufficiently smooth functions. The solution of this problem is given by

\[
y_2(t) = b_2(t) \quad \text{and} \quad y_1(t) = \dot{b}_2(t) + b_1(t).
\]

(1.58)

In that case not only the constraint must be met by the initial condition but also its derivative reduces the freedom in the choice of initial conditions.

For index-2 DAEs the algebraic equation (constraint) and its time derivative restrict the choice of the initial conditions.
For a system with index \( r \) initial conditions which satisfy the constraint and its \( r - 1 \) derivatives are called consistent initial conditions.

Finding consistent initial conditions for practical problems is often extremely hard.

From the differentiation process to determine the index of the problem it becomes evident, that not only the initial conditions but also the entire solution of the differential-algebraic equation systems has to satisfy the constraint and the first \( r - 1 \) derivatives.

**Example 37** Consider the system (1.56) and assume, that the matrix \( \frac{\partial g}{\partial y} \frac{\partial f}{\partial z} \) is nonsingular along the solution. Then, the system has index 2.

The solution of this problem and in particular the initial conditions are elements of the set

\[
M_1 \cap M_2 := \{(y, z) : g(y) = 0\} \cap \left\{ (y, z) : \frac{\partial g}{\partial y}(y) \frac{\partial f}{\partial z}(y, z) = 0 \right\}.
\]

In differential geometry the two sets \( M_1 \) and \( M_2 \) are called differentiable manifolds if \( f \) and \( g \) are sufficiently smooth functions. \( M_1 \) is evident from the problem, while \( M_2 \) normally has to be constructed. Therefore \( M_2 \) is often also called the hidden manifold of the problem. For DAEs of index \( r \) the solution is restricted to the intersection of \( r \) manifolds.

### 1.6.2 Numerical methods for DAEs

For solving DAEs only implicit methods are applicable. Methods which are designed for stiff ODEs can often also applied to DAEs. In particular BDF methods and implicit Runge–Kutta methods of the Radau type are used in practice.

Codes, used in practice are DASSL (BDF-method) and RADAU5 (impl. Runge–Kutta method).

We will not develop here the full theory of discretization for DAEs with stability and convergence considerations. The fundamental techniques can already be understood by considering the implicit Euler method.

We consider first an index-1 system

\[
\begin{align*}
\dot{y} &= f(y, z) \\
0 &= g(y, z)
\end{align*}
\]

with \( \det \left( \frac{\partial g}{\partial z} \right) \neq 0 \) (Index-1 condition).

A discretization with the implicit Euler method has the form

\[
\begin{align*}
y_{n+1} &= y_n + hf(y_{n+1}, z_{n+1}) \\
0 &= g(y_{n+1}, z_{n+1})
\end{align*}
\]
This discrete system can be seen as a system of nonlinear equations for $y_{n+1}$ and $z_{n+1}$ of the form

$$ G(y_{n+1}, z_{n+1}) := \begin{pmatrix} y_{n+1} - y_n - hf(y_{n+1}, z_{n+1}) \\ g(y_{n+1}, z_{n+1}) \end{pmatrix} = 0. $$

Applying Newton’s method to this system requires that the Newton iteration matrix

$$ G'(y_{n+1}, z_{n+1}) = \begin{pmatrix} I - hf'_y(y_{n+1}, z_{n+1}) & -hf'_z(y_{n+1}, z_{n+1}) \\ g'_y(y_{n+1}, z_{n+1}) & g'_z(y_{n+1}, z_{n+1}) \end{pmatrix} $$

is non singular, which is evidently the case for $h$ small enough as long as the index-1 condition is satisfied.

We leave the implementation details of a fixed step size implicit Euler discretization as a MATLAB homework.

Formulate for the same problem a discretization with the explicit Euler method and show, why this results in a method which does not work.

The situation in the index-2 case is very similar. Here we consider again the problem (1.56) with the index-2 condition $\det(g'_y f'_z) \neq 0$.

The discretized form of this equation reads

$$ y_{n+1} = y_n + hf(y_{n+1}, z_{n+1}) $$

$$ 0 = g(y_{n+1}). $$

The corresponding Jacobian for the classical Newton iteration has the following form

$$ G'(y_{n+1}, z_{n+1}) = \begin{pmatrix} I - hf'_y(y_{n+1}, z_{n+1}) & -hf'_z(y_{n+1}, z_{n+1}) \\ g'_y(y_{n+1}) & 0 \end{pmatrix}. $$

This matrix becomes singular, when $h = 0$, so one might expect problems when $h$ tends to zero. By scaling the algebraic variables with the step size $h$ these problems can be avoided:

We write Newton’s method in the scaled form as

$$ \begin{pmatrix} I - hf'_y(y_{n+1}, z_{n+1}) & -hf'_z(y_{n+1}, z_{n+1}) \\ g'_y(y_{n+1}) & 0 \end{pmatrix} \begin{pmatrix} \Delta y_{n+1} \\ \Delta z_{n+1} \end{pmatrix} = -G(y_{n+1}, z_{n+1}) \quad (1.63) $$

(The iteration counters were omitted for simplicity).

Now, the iteration matrix is non singular for sufficiently small $h$ as long as the index-2 condition is fulfilled.

Solving index-3 systems with the implicit Euler method is based on the same technique, though even the equations and not only the variables have to be scaled.

On the other hand solving index-3 systems cause serious problems, when trying to solve them with variable step size and error estimation techniques. The error
propagation models which we discussed in the preceding sections are no longer valid in the higher index-3 case.
In practice index-3 problems (like those arising in mechanics) are reduced by one differentiation step to index-2 problems. The differentiated constraint has in mechanics an interpretation as a constraint on velocity variables and can often be directly obtained from mechanical considerations without performing the differentiation analytically. On the other hand one looses by differentiation an integration constant, which leads to so-called drift-off effects, i.e. the numerical solution leaves the solution manifolds of the original problem. This is often covered by special re-projection techniques.
Bibliography


Index

A-stable, 10
absolute stability, 10
Adams–Moulton methods, 17
autonomous, 4

backward difference formulas, 17
backward difference operator, 16
BDF, 17

Cauchy problems, 1
characteristic polynomial, 13
collocation, 22, 23
   point, 22
   polynomial, 23
consistent, 6
difference equation, 11
   homogene, 11
error constant, 19
evolution of the differential equation, 4
Explicit Euler Method, 5
flow of the differential equation, 4
forward shift operator, 16
fundamental solutions, 13

Gauß
   integration method, 24
   points, 24
generating functions, 17
global error, 6
global error increment, 6, 19
Gronwall’s Lemma, 3

Heun’s method, 5
Implicit Euler Method, 5
implicit Runge–Kutta method, 23
increment function, 5
initial value problem, 1
Lax-Richtmyer principle, 9
Legendre polynomials, 24
linear multistep method, 15
linear test equation, 10
Lipschitz continuous, 2
Milne’s device, 20
multistep method, 15
one step method, 5
order of consistency, 6, 19
region of absolute stability, 10
Stability
   of an initial value problem, 3
   of one step methods, 6
   zero stable, 7
stability
   absolute, 10
   stability bound, 10
state variable, 1
step size, 5
Trapezoidal Rule, 5
zero stable, 7