

CONVERGENCE RESULTS FOR A COORDINATE PROJECTION METHOD APPLIED TO MECHANICAL SYSTEMS WITH ALGEBRAIC CONSTRAINTS*

EDDA EICH†

Abstract. The equations of motion of mechanical multibody systems with algebraic constraints are of index 3 and therefore not directly solvable by standard ODE or DAE methods. Reducing the index by differentiating the constraints results in an ODE or reduced index DAE with invariants. The presence of discretization errors in the numerical solution leads to violations of the invariants and eventually yields a drift-off from the manifold given by the invariants. As a consequence one obtains physically meaningless solutions.

To overcome this difficulty a coordinate projection method is presented, which projects the discretized solution onto the invariant manifold. Convergence theorems for a combination of the BDF-method and projection are given. The techniques used in the proof allow insight into the way the errors are propagated. In particular, it can be shown that only those parts of the errors lying in the manifold will be propagated. This leads to solutions which not only satisfy the invariant but are more accurate.

Key words. differential-algebraic equations, constrained mechanical systems, conservation laws

AMS subject classification. 65L05

1. Introduction.

1.1. Equations of motion of mechanical systems and ODEs with invariants. Mechanical systems like robots or vehicles are often modeled as multibody systems consisting of n rigid bodies connected by joints or force elements. Constraints occur in the presence of joints connecting the bodies. These constraints lead to a reduction of the degrees of freedom of the system.

The equations of motions can be written in the following form¹:

$$(1.1) \quad M(p)\ddot{p} = F(p, \dot{p}) - G^T(p)\lambda,$$

$$(1.2) \quad 0 = g(p)$$

with

- p, \dot{p} the n_p position and velocity variables,
- $M(p)$ the positive definite mass matrix,
- $F(p, \dot{p})$ the applied forces,
- $G(p) := \frac{d}{dp}g(p) \in \mathbb{R}^{n_\lambda \times n_p}$ the constraint matrix,
- λ the n_λ Lagrange multipliers,
- $G^T(p)\lambda$ the constraint forces.

Together, (1.1) and (1.2) are a system of differential-algebraic equations of index 3 and therefore not directly solvable by standard ODE or DAE methods. The index can be reduced by differentiating the algebraic equations (1.2).

* Received by the editors March 11, 1992; accepted for publication (in revised form) September 25, 1992. This work has been supported by the Deutsche Forschungsgemeinschaft, Research Program "Dynamics of Multibody Systems."

† Institut für Mathematik der Universität Augsburg, Universitätsstrasse 2, D-8900 Augsburg, Germany. Current Address, Kiltrahinger Strasse 7, D 82205 Gilching, Germany.

¹ For ease of notation only this simplified case is presented. Nonholonomic constraints as well as forces f depending on Lagrange multipliers are neglected. Furthermore, we restrict ourselves to the autonomous case. However, all the analysis extends to more general cases.

One differentiation of the *constraints on position level* (1.2) leads to *constraints on velocity level*

$$(1.3) \quad 0 = G(p)\dot{p},$$

and one more differentiation leads to the *constraint equations on acceleration level*

$$(1.4) \quad 0 = G(p)\ddot{p} + \dot{p}^T G_p(p)\dot{p}, \quad G_p := \frac{d^2 g}{dp^2} = \frac{dG}{dp}.$$

Summarizing (1.1) and (1.4) results in the index-1 DAE

$$(1.5) \quad \begin{pmatrix} M & G^T \\ G & 0 \end{pmatrix} \begin{pmatrix} \ddot{p} \\ \lambda \end{pmatrix} = \begin{pmatrix} F(p, \dot{p}) \\ -\dot{p}^T G_p(p)\dot{p} \end{pmatrix}.$$

Under the above assumptions on M this system is uniquely solvable for $\begin{pmatrix} \ddot{p} \\ \lambda \end{pmatrix}$ if G has full rank. Explicitly solving this system for λ yields

$$(1.6) \quad \lambda = (GM^{-1}G^T)^{-1} [GM^{-1}F + \dot{p}^T G_p \dot{p}].$$

Substituting this result into (1.1) leads to the ODE system

$$(1.7) \quad M(p)\ddot{p} = \tilde{F}(p, \dot{p}),$$

with

$$\tilde{F}(p, \dot{p}, t) := F(p, \dot{p}) - G(p)^T (G(p)M(p)^{-1}G^T(p))^{-1} [G(p)M^{-1}(p)F(p, \dot{p}) + \dot{p}^T G_p(p)\dot{p}].$$

The exact solutions of the initial value problems of the four different formulations,

- the index-3 formulation (1.1), (1.2);
- the index-2 formulation (1.1), (1.3);
- the index-1 formulation (1.5); and
- the underlying ODE formulation (1.7)²

are the same, if the initial values $(p_0, \dot{p}_0, \lambda_0)^T$ are consistent, i.e., if they satisfy all constraints (1.2), (1.3), and (1.6).

In the presence of discretization errors, these formulations are no longer equivalent and have different solutions.

Let $\delta_w(y_n)$ denote the discretization operator of a linear multistep method for $\dot{y} = f(y)$

$$(1.8) \quad \delta_w(y_n) := \frac{\alpha_0}{h} y_n + \frac{1}{h} \sum_{i=1}^k \alpha_i w_{n-i} - \sum_{i=1}^k \beta_i f(w_{n-i}) = f(y_n).$$

The subindex w at δ denotes which values are taken at past timesteps.

Then the numerical solution of the discretized index-1 system (1.5), written as first-order system ($\dot{p} = v$)

$$\begin{aligned} \delta_p(p_n) &= v_n, \\ M(p_n)\delta_v(v_n) &= F(p_n, v_n) - G^T(p_n)\lambda_n, \\ 0 &= G(p_n)\delta_v(v_n) + v_n^T G_p(p_n)v_n, \end{aligned}$$

² Some authors define the underlying ODE including a differential equation for λ .

in general does not satisfy the constraints on position level and the constraints on velocity level:

$$g(p_n) \neq 0,$$

$$G(p_n)v_n \neq 0.$$

As a consequence of error propagation eventually a drift-off from these constraints can be observed. This leads to physically meaningless solutions as shown below.

1.2. Example: The “flying” wheel set. A model of a wheel set running on a track [23] is used in order to demonstrate the drift-off from the constraints when only solving the index-1 system.

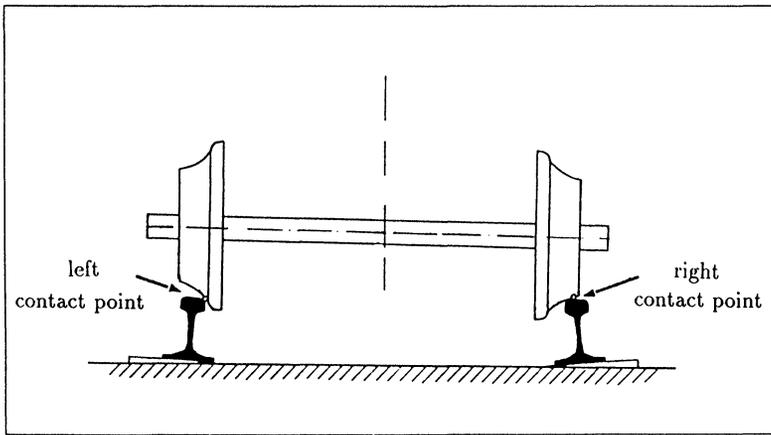


FIG. 1.1. *Wheel set and track, lateral cross section.*

The algebraic equations originate from the following.

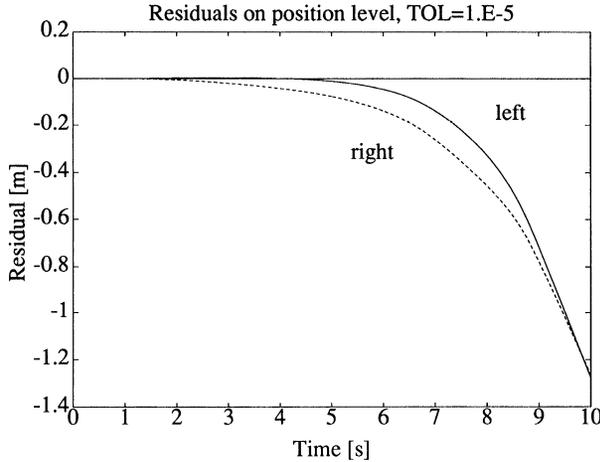
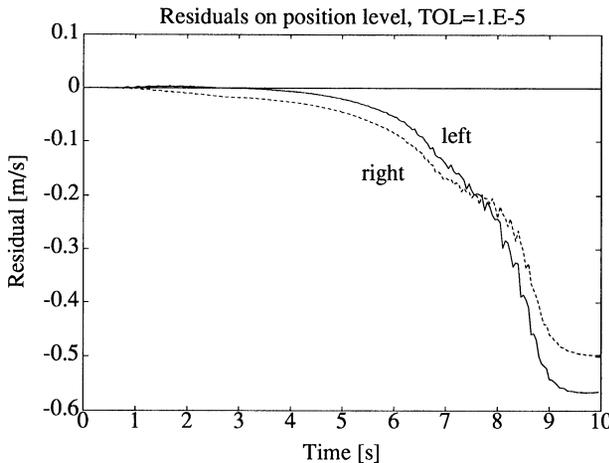
- The contact condition between wheel and track: the contact points between wheel and track must be the same points in space (1 equation for each wheel). This equation is of the form (1.2) and therefore an index 3 equation.
- The nonintersecting condition: the tangent planes on wheel and track in the contact points must coincide in order to avoid intersection between wheel and track (2 equations for each wheel). These equations are an index-1 system.

Both types of equations depend on the profile functions for wheel and track. For a detailed description and the equations; see [23]. Here we model a wheel set with two wheels and therefore have to deal with six algebraic equations.

The dynamic behaviour of the wheel set is given by five differential equations of second order and one equation of first order. The forces take into account gravity and centrifugal forces, creep forces in the contact points, forces on the car body, and constraint forces. Note that due to creepages the applied forces depend nonlinearly on λ . In the case considered here a dicone running on a circular track is considered.

The BDF integrator MKS-DAESOL [5] is used to integrate the index-1 system numerically. This code is designed to exploit the special multibody structure, and employs variable step, variable order error control strategies.

Integration over 10 seconds with an accuracy of $TOL=10^{-5}$ leads to residuals on position and velocity level shown in Figs. 1.2 and 1.3.

FIG. 1.2. *Residuals on position level.*FIG. 1.3. *Residuals on velocity level.*

In Fig. 1.4 some pictures from computer animation are given demonstrating that integrating the index-reduced system leads to a physically totally meaningless solution: after 10 seconds the wheel set is flying at a height of 1.27 m!

1.3. Overview. To overcome this difficulty, in §2 a coordinate projection method is presented. Applying this new method to the wheel set example leads to residuals which can be made arbitrarily small (within roundoff). In the figures above they therefore cannot be distinguished from the zero axis.

A convergence analysis for the BDF-method combined with this projection technique is presented in §3. Note that in the case of BDF-methods or, more generally, multistep methods, this proof requires other techniques than for one-step methods since the correction by projection enters into the error equation. The proof for one-step methods was given by Shampine in [22] in the framework of ODEs with invariants.

The techniques used for the proof here allow insight into the way the errors are propagated: In the linear case only those errors lying in the manifold given by the constraints are propagated. This leads to an improved accuracy of the solution, which is demonstrated by the pendulum example. Section 4 briefly gives some aspects of the numerical computation of the coordinate projection method.

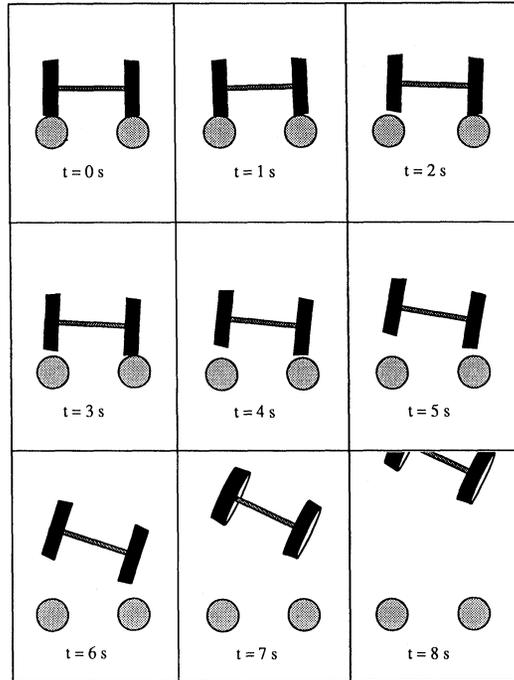


FIG. 1.4. Animation of wheel set without projection.

Various other regularization and projection methods have been developed. *Regularization methods*, e.g., [7], [14], [16], and [17], reduce the index by introducing additional regularization parameters into the original system. The famous approach of Baumgarte [2], [20] can be interpreted as a regularization method as shown in [7], where also convergence results are presented: the solution of the perturbed system can be written as a power series depending on the regularization parameter.

In contrast to projection methods, regularization methods only achieve an asymptotic approximation to the manifold given by the invariants, whereas projection methods produce a solution lying in the manifold in every timestep.

Most *projection methods* can be subdivided into two classes, coordinate and derivative projection methods [6], depending on whether the coordinates itself or the derivative f is projected onto the manifold. The methods presented, e.g., in [8]–[13], [19], [21], [25], and [26], are *derivative projection* methods based on viewing the DAE as an ODE on a manifold and using local parametrizations of the manifold to construct the method. The method described here is a *coordinate projection method*. It avoids the unnecessary drift-off from the constraints by projecting the solution of the reduced index system back onto the manifold. Similar methods have been used in [1], [19], and [22] in the context of one-step methods.

2. A coordinate projection method. For the numerical integration of ODEs with invariants:

$$(2.1) \quad \dot{y} = f(y), \quad y(t_0) = y_0,$$

$$(2.2) \quad 0 = \phi(y) - \phi(y_0),$$

a coordinate projection method is presented. Equation (2.2) is called an (integral) invariant of the ODE (2.1) if and only if for all initial values y_0 (2.2) holds. Necessary and sufficient for ϕ to be an integral invariant is

$$(2.3) \quad H(y)f(y) := \frac{d\phi}{dy}f(y) = 0.$$

The equations of motion of multibody systems easily fit into this context: Taking the underlying ODE (1.7), the position and velocity constraints (1.2),(1.3) can be regarded as invariants of the solution

$$\phi(y) = \begin{pmatrix} g(p) \\ G(p)v \end{pmatrix}.$$

For consistent initial values we have

$$\phi(y_0) = \begin{pmatrix} g(p_0) \\ G(p_0)v_0 \end{pmatrix} = 0.$$

Therefore, we assume $\phi(y_0) = 0$ for ease of notation.

In this section we restrict ourselves to ODEs with invariants. However, in the implementation of MKS-DAESOL the underlying ODE is not computed explicitly since this requires the computation and decomposition of $GM^{-1}G^T$ for every function evaluation. Instead, the discretized version of the index-1 system (1.5) is solved iteratively. This has no influence on the error analysis presented in the sequel because semi-explicit index-1 DAEs behave under discretization like ODEs for most integration methods, especially for the BDF method [4]. The index-1 variable λ needs not to be recomputed after the projection has been carried out, because this would only affect the predictor values but not the corrector in the next step since the algebraic variable λ enters linearly.

Note, that principally this method can be used to solve DAEs of arbitrary high index by differentiating the equations until an index-1 system is obtained. The system is then stabilized by projection onto the higher-index equations.

Other examples for invariants are conservation laws, e.g., for energy or momentum.

The coordinate projection method mainly consists of two steps for each timestep.

1. The numerical solution \tilde{y}_n of the ODE (2.1) is computed by a discretization method:

$$(2.4) \quad \delta_y(\tilde{y}_n) = f(\tilde{y}_n).$$

This is done using projected values obtained at past timesteps.

2. The solution \tilde{y}_n is then projected orthogonally back onto the manifold given by the invariants, i.e., the projected solution is computed as the solution of

$$(2.5) \quad \|y_n - \tilde{y}_n\|_2 = \min_{y_n}$$

$$(2.6) \quad \phi(y_n) = 0.$$

These projected values are used to advance the solution.

Remark 2.1. In [22] the projection 2. was discussed for one-step methods. There this projection was viewed as an arbitrary perturbation of the solution, which may not be applicable to multistep methods. Here it will be shown that the projection

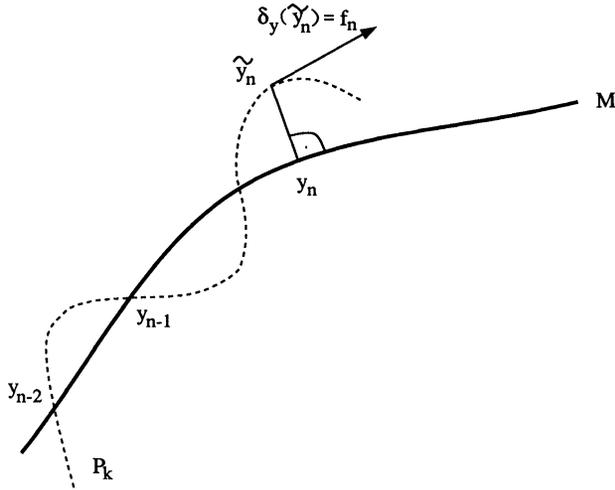


FIG. 2.1. Illustration of projection.

has a well defined influence on the discretization error of multistep methods so that convergence can be shown also for this case. Figure 2.1 illustrates this projection for the BDF method.

In case of linear invariants of the form

$$0 = Hy(t) + z(t)$$

the solution can be expressed using the Moore–Penrose pseudo-inverse H^+ of H . Let V be a matrix whose columns span the null-space of H , i.e., $HV = 0$. Then we get

$$(2.7) \quad y_n = -H^+ z_n + P\tilde{y}_n$$

with

$$P := VV^+ = V(V^T V)^{-1} V^T.$$

P is an orthogonal projector onto $\ker(H)$. $T := I - P = H^+ H = H^T (HH^T)^{-1} H$ is an orthogonal projector onto $\text{im}(H^T) = \ker(H)^\perp$.

Remark 2.2. Gear [12] has shown that for linear invariants of the form

$$\phi(y) = Hy + b$$

one-step methods of the form

$$y_n = y_{n-1} + h\Phi(y_{n-1}, h)$$

with $H\Phi(y, h) = 0$ keep the invariant. A similar result holds for consistent linear multistep methods of the form (1.8) as can be seen from

$$Hy_n = \frac{1}{\alpha_0} H \left(- \sum_{i=1}^k \alpha_i y_{n-i} + h \sum_{i=0}^k \beta_i f(y_{n-i}) \right) = - \sum_{i=1}^k \frac{\alpha_i}{\alpha_0} Hy_{n-i} = \underbrace{\sum_{i=1}^k \frac{\alpha_i}{\alpha_0}}_{=-1} b = -b.$$

Herein the k starting values must satisfy the invariant. This result no longer holds for linear *time variant* invariants like

$$Hy(t) + z(t) = 0$$

with $z \neq \text{const.}$, which are considered here.

In the general nonlinear case a solution y_n of the projection step 2. must satisfy the Euler equations

$$(2.8) \quad y_n = \tilde{y}_n - H^T(y_n)\eta_n,$$

$$(2.9) \quad \phi(y_n) = 0,$$

with $H := \frac{d\phi}{dy}$. This relation will be used in the sequel.

3. Convergence results. In this section convergence results for the coordinate projection method in the framework of BDF methods are given. The main result is that the projection does not disturb the convergence properties of the BDF method. This is proven by relating the error equations for the projected method to those of the original method and using the known stability results for these methods (see, e.g., [11]).

In the first part of this section we give a convergence proof for linear constant coefficient ODEs with linear time variant invariants. This is done in order to get further insight into the nature of errors and their propagation. It will be shown

- that only those parts of the discretization errors are propagated, which lie in the manifold given by the invariants;
- that the local error of the projected method is smaller than the local error of the original method.

In the second part a convergence proof for nonlinear systems will be given.

3.1. Convergence results for linear systems. Let us consider the linear constant coefficient system

$$(3.1) \quad \dot{y}(t) = Ay(t) + u(t),$$

$$(3.2) \quad 0 = Hy(t) + z(t).$$

THEOREM 3.1. *Let H be of full rank. If the starting values satisfy*

$$e_i = \mathcal{O}(h^k) \\ Pe_i = e_i, \quad i = 0, \dots, k - 1,$$

$e_i = y_i - y(ti)$ denoting the global error, then the projected k -step BDF method ($1 \leq k \leq 6$) converges with the same order as the corresponding original method, i.e.,

$$e_n = \mathcal{O}(h^k) \quad \text{for } nh \leq \text{const.}$$

if h is sufficiently small.

Proof. Let ρ be the BDF discretization operator based on projected values y_{n-i} from past timesteps (using the same notation as for δ):

$$\rho_y(\tilde{y}_n) = \alpha_0 \tilde{y}_n + \sum_{i=1}^k \alpha_i y_{n-i}.$$

Let \tilde{y}_n be the solution of the discretization scheme applied to (3.1),

$$\frac{1}{h}\rho_y(\tilde{y}_n) = A\tilde{y}_n + u(t_n).$$

Let y_n be obtained from projecting this solution onto the manifold given by (3.2), thus by (2.7),

$$y_n = -H^+z(t_n) + P\tilde{y}_n.$$

Let τ_n be the local discretization error

$$(3.3) \quad \tau_n := \dot{y}(t_n) - \frac{1}{h} \sum_{i=0}^k \alpha_i y(t_{n-i})$$

and e_n the global error of the projected solution $e_n := y_n - y(t_n)$. With these definitions we get

$$\begin{aligned} e_n &= y_n - y(t_n) \\ &= P(\tilde{y}_n - y(t_n)) \\ &= P(\alpha_0 I - hA)^{-1} \left[-\sum_{i=1}^k \alpha_i e_{n-i} + h\tau_n \right] \\ &= P \left[\left(-\frac{\alpha_1}{\alpha_0} + \mathcal{O}(h) \right) e_{n-1} + \left(-\frac{\alpha_2}{\alpha_0} + \mathcal{O}(h) \right) e_{n-2} + \dots + \left(-\frac{\alpha_k}{\alpha_0} + \mathcal{O}(h) \right) e_{n-k} \right. \\ &\quad \left. + \left(\frac{h}{\alpha_0} + \mathcal{O}(h^2) \right) \tau_n \right] \end{aligned}$$

if h is sufficiently small.

Rewriting this in the form of a one-step method, we obtain

$$\begin{aligned} (3.4) \quad \begin{pmatrix} e_n \\ e_{n-1} \\ \vdots \\ e_{n-k+1} \end{pmatrix} &= \left[\begin{pmatrix} -\frac{\alpha_1}{\alpha_0} P & -\frac{\alpha_2}{\alpha_0} P & \dots & -\frac{\alpha_k}{\alpha_0} P \\ I & 0 & \dots & 0 \\ 0 & \ddots & & \\ & & I & 0 \end{pmatrix} + \mathcal{O}(h) \right] \begin{pmatrix} e_{n-1} \\ e_{n-2} \\ \vdots \\ e_{n-k} \end{pmatrix} \\ &\quad + \begin{pmatrix} \left(\frac{h}{\alpha_0} + \mathcal{O}(h^2) \right) P\tau_n \\ 0 \\ \vdots \\ 0 \end{pmatrix} \\ &= \left[\begin{pmatrix} -\frac{\alpha_1}{\alpha_0} I & -\frac{\alpha_2}{\alpha_0} I & \dots & -\frac{\alpha_k}{\alpha_0} I \\ I & 0 & \dots & 0 \\ 0 & \ddots & & \\ & & I & 0 \end{pmatrix} + \mathcal{O}(h) \right] \begin{pmatrix} Pe_{n-1} \\ Pe_{n-2} \\ \vdots \\ Pe_{n-k} \end{pmatrix} \\ &\quad + \begin{pmatrix} \left(\frac{h}{\alpha_0} + \mathcal{O}(h^2) \right) P\tau_n \\ 0 \\ \vdots \\ 0 \end{pmatrix}. \end{aligned}$$

Since

$$\begin{aligned}
 e_{n-i} &= y_{n-i} - y(t_{n-i}) \\
 &= -H^+ z_{n-i} + P y_{n-i} - (-H^+ z_{n-i} + P y(t_{n-i})) \\
 (3.5) \quad &= P(y_{n-i} - y(t_{n-i})) \\
 &= P e_{n-i},
 \end{aligned}$$

the same difference equation as for the BDF-method applied to ODEs is obtained and hence follows convergence. \square

We now can conclude the following from the proof.

- Only the error lying in the manifold enters the error equation; see (3.5);
- Only the projected part of the local error is propagated; see (3.4).

P is an orthogonal projector and therefore $\|P\| \leq 1$ and the errors of the projected method are actually reduced. This can also be seen from the example in §3.3.

3.2. Convergence results for nonlinear systems. Next, consider the autonomous nonlinear system

$$\begin{aligned}
 \dot{y} &= f(y), \\
 0 &= \phi(y).
 \end{aligned}$$

In this section it will be shown that the errors of the projected BDF methods of order k are $\mathcal{O}(h^k)$ as in the nonprojected case.

THEOREM 3.2. *Let ϕ, f be sufficiently differentiable, at least twice. Assume that $(HH^T)^{-1}$ exists and is bounded for all $t \in [t_0, t_e]$ in a neighbourhood of the solution and that the starting values satisfy*

$$\begin{aligned}
 e_i &= \mathcal{O}(h^k) \\
 P e_i &= e_i, \quad i = 0, \dots, k - 1.
 \end{aligned}$$

Then the projected k -step BDF method ($1 \leq k \leq 6$) has the same order of convergence as the corresponding unprojected method, i.e.,

$$e_n = \mathcal{O}(h^k) \quad \text{for } nh \leq \text{const.}$$

if h is sufficiently small.

Proof. The proof is based on relating the error equation for the projected method to the one for the original BDF method.

The techniques used in the proof are similar to those used, e.g., in [3] and [18] to prove convergence for semi-explicit index-2 systems. However, in contrast to the stabilization method proposed by Gear in [12],

$$\begin{aligned}
 \dot{y} &= f(y) + H^T \mu, \\
 0 &= \phi(y),
 \end{aligned}$$

this coordinate projection method cannot be written as a (discretized) index-2 system, such that these convergence results cannot be applied directly.

Let τ_n be the local discretization error

$$(3.6) \quad \tau_n := \dot{y}(t) - \frac{1}{h} \sum_{i=0}^k \alpha_i y(t_{n-i}) = f(y(t_n)) - \frac{1}{h} \rho_y(t)(y(t_n)).$$

Using the Euler equations (2.8) and (2.9) we can rewrite

$$\begin{aligned} \frac{1}{h}\rho_y(\tilde{y}_n) &= f(\tilde{y}_n), \\ \|y_n - \tilde{y}_n\|_2 &= \min_{y_n}, \\ \phi(y_n) &= 0 \end{aligned}$$

by introducing the Lagrange multipliers η_n

$$\begin{aligned} \frac{1}{h}\rho_y(y_n + H^T(y_n)\eta_n) &= f(y_n + H^T(y_n)\eta_n), \\ \phi(y_n) &= 0. \end{aligned}$$

From this and from (3.6) we get by linearization

$$(3.7) \quad \frac{1}{h}\rho_e(e_n) = f_y(y(t_n))(e_n + H^T(y(t_n))\eta_n) - \frac{\alpha_0}{h}H^T(y(t_n))\eta_n + \tau_n + R_1(e_n, \eta_n)$$

$$(3.8) \quad 0 = H(y(t_n))e_n + R_2(e_n),$$

where R_1, R_2 summarize higher order terms in e_n, η_n :

$$\begin{aligned} R_{1,n}(e_n, \eta_n) &= \frac{1}{2}(e_n + H^T(y(t_n))\eta_n)^T \tilde{f}_{yy}(e_n + H^T(y(t_n))\eta_n) \\ &\quad + f_y(y(t_n))(e_n^T \tilde{H}_y^T \eta_n) - \frac{\alpha_0}{h}e_n^T \tilde{H}_y^T \eta_n, \\ R_{2,n}(e_n, \eta_n) &= \frac{1}{2}e_n^T \tilde{H}_y e_n. \end{aligned}$$

The tilde indicates that the term is not evaluated on the exact solution but in a neighbourhood of it, maybe at different locations for each component.

Now, we rewrite this as a linear system³ for $(e_n^T \ \eta_n^T)^T$:

$$(3.9) \quad \begin{pmatrix} \alpha_0 I - hf_y & (\alpha_0 I - hf_y)H^T \\ H & 0 \end{pmatrix} \begin{pmatrix} e_n \\ \eta_n \end{pmatrix} = \begin{pmatrix} -\sum_{i=1}^k \alpha_i e_{n-i} + hR_{1,n} + h\tau_n \\ -R_{2,n} \end{pmatrix}.$$

Solving this for e_n results in

$$(3.10) \quad \begin{aligned} e_n &= (I - H_n^T(H_n H_n^T)^{-1}H_n)(\alpha_0 I - hf_y)^{-1} \left(-\sum_{i=1}^k \alpha_i e_{n-i} + hR_{1,n} + h\tau_n \right) \\ &\quad - H_n^T(H_n H_n^T)^{-1}R_{2,n}, \end{aligned}$$

with $H_n := H(y(t_n))$. Premultiplying this by the projector $P_n := (I - H_n^T(H_n H_n^T)^{-1}H_n)$ yields

$$(3.11) \quad P_n e_n = P_n(\alpha_0 I - hf_y)^{-1} \left(-\sum_{i=1}^k \alpha_i e_{n-i} + hR_{1,n} + h\tau_n \right).$$

³ Note that $R_{i,n}$ still depend on e_n, η_n .

For h sufficiently small,

$$(\alpha_0 I - hf_y)^{-1} = \frac{1}{\alpha_0} \left(I + \frac{h}{\alpha_0} A \right), \quad A = \mathcal{O}(1)$$

holds where A summarizes the remainder from Neumann series.

Inserting this into (3.11) gives

$$P_n \left(\sum_{i=0}^k \alpha_i e_{n-i} \right) = P_n(hR_{1,n} + h\tau_n) + \frac{h}{\alpha_0} P_n A \left(- \sum_{i=1}^k \alpha_i e_{n-i} + hR_{1,n} + h\tau_n \right),$$

and we get

$$(3.12) \quad \sum_{i=0}^k \alpha_i P_{n-i} e_{n-i} = h \sum_{i=1}^k \alpha_i \left(\frac{P_{n-i} - P_n}{h} - \frac{1}{\alpha_0} P_n A \right) e_{n-i} + h P_n \left(I + \frac{h}{\alpha_0} A \right) (R_{1,n} + \tau_n).$$

From the assumptions it follows that H, H^+ are differentiable and therefore P is, too. This leads to

$$\left\| \frac{P_n - P_{n-i}}{h} \right\| \leq iK$$

and we get from (3.12)

$$(Pe)_n = \sum_{i=1}^k \left(-\frac{\alpha_i}{\alpha_0} + \mathcal{O}(h) \right) (Pe)_{n-i} + \frac{h}{\alpha_0} P_n \left(I + \frac{h}{\alpha_0} A \right) (R_{1,n} + \tau_n),$$

with $(Pe)_i := P_i e_i$. The matrix on the right-hand side is the same as in the case of the unprojected BDF method except for the $\mathcal{O}(h)$ term. In [15] and [24] it has been shown that this perturbation does not destroy stability. Therefore, we get

$$\|P_n e_n\| = \mathcal{O}(\bar{R}_{1,n}) + \mathcal{O}(\bar{\tau}_n) + \mathcal{O}(\|e_0\|),$$

with

$$\bar{R}_{i,n} = \max_{j \leq n} \|R_{i,j}\|, \quad i = 1, 2, \quad \bar{\tau}_n = \max_{j \leq n} \|\tau_n\|.$$

Premultiplying (3.10) by $H_n^T (H_n H_n^T)^{-1} H_n$ results in

$$T_n e_n = -H_n^T (H_n H_n^T)^{-1} R_{2,n}$$

with $T_n = H_n^T (H_n H_n^T)^{-1} H_n = I - P_n$ and we get

$$\|T_n e_n\| = \mathcal{O}(\|R_{2,n}\|).$$

From $e_n = T_n e_n + P_n e_n$ follows

$$(3.13) \quad \|e_n\| = \mathcal{O}(\bar{R}_{1,n}) + \mathcal{O}(\bar{R}_{2,n}) + \mathcal{O}(\bar{\tau}_n) + \mathcal{O}(\|e_0\|).$$

This is an implicit equation for e_n , because $R_{i,n}, i = 1, 2$ are still dependent on e_n . Since $R_{1,n}, R_{2,n}$ depend, besides on e_n , also on η_n , we first have to estimate η_n before we can estimate e_n from the implicit system (3.13) with the help of a fixed-point argument.

Solving (3.9) for η_n results in

$$\eta_n = (H_n H_n^T)^{-1} H_n (\alpha_0 I + h f_y)^{-1} \left(- \sum_{i=0}^k \alpha_i e_{n-i} + h \tau_n + h R_{1,n} + h f_y e_n \right).$$

Together with (3.13) we get

$$(3.14) \quad \|\eta_n\| = \mathcal{O}(\bar{R}_{1,n}) + \mathcal{O}(\bar{R}_{2,n}) + \mathcal{O}(\bar{\tau}_n) + \mathcal{O}(\|e_0\|).$$

Now, we have to show that R_1, R_2 are small.

For this reason we use a fixed-point argument which has been used in [18] and [4] for a similar purpose. Assume that for $\bar{R}_{1,n}, \bar{R}_{2,n}$ we have the inequalities

$$\bar{R}_{i,n} \leq \varepsilon_i, \quad i = 1, 2$$

and that for the errors in the starting values we have $\|e_0\| = \mathcal{O}(h^k)$. Then we must show that ε_i are small.

From the definition of R_i we get using (3.13),(3.14)

$$\begin{aligned} \bar{R}_{1,n} &\leq K_1 \left((\|e_n\| + \|H_n^T \eta_n\|)^2 + \frac{1}{h} \|e_n\| \|\eta_n\| \right) \\ &\leq K_2 \left((h^k + \varepsilon_1 + \varepsilon_2)^2 + \frac{1}{h} (h^k + \varepsilon_1 + \varepsilon_2)^2 \right) \\ \bar{R}_{2,n} &\leq K_3 (h^k + \varepsilon_1 + \varepsilon_2)^2. \end{aligned}$$

$\varepsilon_1, \varepsilon_2$ can be determined as solutions of

$$\begin{pmatrix} \varepsilon_1 \\ \varepsilon_2 \end{pmatrix} = \underbrace{\begin{pmatrix} K_4 ((h^k + \varepsilon_1 + \varepsilon_2)^2 + \frac{1}{h} (h^k + \varepsilon_1 + \varepsilon_2)^2) \\ K_5 (h^k + \varepsilon_1 + \varepsilon_2)^2 \end{pmatrix}}_{F(\varepsilon)}$$

by functional iteration $\varepsilon^{(i+1)} = F(\varepsilon^{(i)})$ ($F : U_{h^k}(0) \rightarrow U_{h^k}(0)$) and a starting value $\varepsilon_i^{(0)} = 0$. F is for small h a contractive mapping with the contraction coefficient $\|F'\| \leq r = \mathcal{O}(h^{k-1})$, if $k \geq 2$. Thus, from the contractive mapping theorem we get

$$\|\varepsilon\| = \|\varepsilon - \varepsilon^{(0)}\| \leq \frac{1}{1-r} \|\varepsilon^{(1)} - \varepsilon^{(0)}\| \leq \frac{1}{1-r} K_6 \|h^{2k-1}\|.$$

For $k = 1$ the result can be obtained by scaling the variables as in [18]. □

Remark 3.1. Errors resulting from not exactly solving the nonlinear equations (e.g., because of truncation errors or not iterating until convergence), can also be subsummed in R_1, R_2 . Assuming that these errors are $\mathcal{O}(h^{k+1})$, we obtain, with the same fixed-point argument as above,

$$\varepsilon_i = \mathcal{O}(h^{k+1}),$$

and the method converges.

Remark 3.2. The corresponding results for variable order and stepsize can be obtained, because in no part of the proof has the constancy of stepsize and order been used. One only has to assume that the nonprojected method is stable.

3.3. Example. In order to demonstrate the error reduction the example of a simple pendulum is considered, because its solution is known in the turning points. The equations of motion are

$$\begin{aligned}
 (3.15) \quad & m\ddot{x} = \lambda x, \\
 (3.16) \quad & m\ddot{y} = -mg + \lambda y, \\
 (3.17) \quad & 0 = x^2 + y^2 - 1, \\
 (3.18) \quad & 0 = x\dot{x} + y\dot{y}, \\
 (3.19) \quad & 0 = x\ddot{x} + y\ddot{y} + \dot{x}^2 + \dot{y}^2.
 \end{aligned}$$

The gravitational acceleration g is chosen as $g = 13.7503716373294544$ in order to get a period of 2 seconds. The starting values are

$$x(0) = 1, \quad y(0) = 0, \quad \dot{x}(0) = 0, \quad \dot{y}(0) = 0.$$

The results obtained with MKS-DAESOL by integrating the index-1 system (3.15), (3.16), (3.19) over 100 seconds, i.e., 50 periods, and projecting back after each timestep in order to satisfy (3.17), (3.18) are summarized in the following table.

TABLE 3.1

		absolute error					RES	ERR
TOL	Proj.	Δx	Δy	$\Delta \dot{x}$	$\Delta \dot{y}$	$\Delta \lambda$		
10^{-5}	no	$5.6 \cdot 10^{-2}$	$4.1 \cdot 10^{-1}$	$1.3 \cdot 10^0$	$2.9 \cdot 10^0$	$1.4 \cdot 10^1$	$6.1 \cdot 10^{-2}$	$1.3 \cdot 10^{-1}$
10^{-5}	yes	$2.4 \cdot 10^{-8}$	$2.2 \cdot 10^{-4}$	$3.6 \cdot 10^{-5}$	$1.6 \cdot 10^{-1}$	$2.4 \cdot 10^{-2}$	$7.2 \cdot 10^{-10}$	$5.4 \cdot 10^{-3}$
10^{-6}	no	$9.7 \cdot 10^{-5}$	$2.1 \cdot 10^{-5}$	$2.9 \cdot 10^{-5}$	$8.7 \cdot 10^{-2}$	$7.9 \cdot 10^{-3}$	$1.9 \cdot 10^{-4}$	$2.9 \cdot 10^{-3}$
10^{-6}	yes	$9.7 \cdot 10^{-9}$	$1.4 \cdot 10^{-4}$	$3.9 \cdot 10^{-6}$	$2.8 \cdot 10^{-2}$	$1.1 \cdot 10^{-3}$	$5.9 \cdot 10^{-11}$	$9.3 \cdot 10^{-4}$
10^{-7}	no	$2.4 \cdot 10^{-4}$	$3.7 \cdot 10^{-4}$	$3.9 \cdot 10^{-6}$	$8.3 \cdot 10^{-3}$	$5.0 \cdot 10^{-3}$	$4.9 \cdot 10^{-4}$	$2.8 \cdot 10^{-4}$
10^{-7}	yes	$9.1 \cdot 10^{-12}$	$4.5 \cdot 10^{-6}$	$2.9 \cdot 10^{-9}$	$5.7 \cdot 10^{-4}$	$6.2 \cdot 10^{-5}$	$2.1 \cdot 10^{-12}$	$1.9 \cdot 10^{-5}$
10^{-8}	no	$1.5 \cdot 10^{-4}$	$2.0 \cdot 10^{-4}$	$5.4 \cdot 10^{-6}$	$1.4 \cdot 10^{-2}$	$3.0 \cdot 10^{-3}$	$2.9 \cdot 10^{-4}$	$4.7 \cdot 10^{-4}$
10^{-8}	yes	$1.3 \cdot 10^{-13}$	$5.3 \cdot 10^{-7}$	$7.2 \cdot 10^{-11}$	$1.1 \cdot 10^{-4}$	$7.3 \cdot 10^{-6}$	$2.1 \cdot 10^{-14}$	$3.5 \cdot 10^{-6}$
10^{-9}	no	$1.7 \cdot 10^{-5}$	$2.3 \cdot 10^{-5}$	$3.3 \cdot 10^{-7}$	$1.6 \cdot 10^{-3}$	$3.1 \cdot 10^{-4}$	$3.3 \cdot 10^{-5}$	$5.2 \cdot 10^{-5}$
10^{-9}	yes	$5.0 \cdot 10^{-15}$	$9.8 \cdot 10^{-8}$	$1.4 \cdot 10^{-12}$	$1.6 \cdot 10^{-5}$	$1.3 \cdot 10^{-6}$	$3.3 \cdot 10^{-16}$	$5.4 \cdot 10^{-7}$

The error ERR is computed in the same norm as used for the error computation in the integration routine, where each component is scaled by the value of the corresponding variable. $RES = x^2 + y^2 - 1$ denotes the residual on position level. From this table we can conclude that the projection method reduces the errors significantly.

4. Aspects of numerical computation.

4.1. Error estimation. The reduced error of the projection method allows the error formulas to be changed.

In the case of linear systems (3.1),(3.2) it follows from the definition of the local discretization error τ_n that the local error \tilde{d}_n of the unprojected method satisfies

$$\tilde{d}_n = (\alpha_0 I - hA)^{-1} h\tau_n.$$

The local error d_n of the projected method is given as

$$d_n = P(\alpha_0 I - hA)^{-1} h\tau_n = P\tilde{d}_n.$$

Thus the local error of the projected method can be obtained by projection of the local error of the original method.

4.2. Computation of projection step. Only a brief summary of the computation of the projection step is given. Details can be found in [5] and will be published in a forthcoming report.

The numerical solution of the projection step (2) can be computed iteratively by a Gauss–Newton method which avoids the explicit computation of η_n . It can be shown that this method converges with a rate $r = \mathcal{O}(h^k)$. Thus, by reducing the stepsize convergence can always be obtained. The effort due to decomposition and solution of the corresponding linearized system can be significantly reduced by relaxing the orthogonality requirement of the projection slightly. Further gains in effectiveness can be obtained by taking into consideration the special multibody structure in both steps, the discretization of the index-1 system (1.5) and the Gauss–Newton iteration.

Acknowledgments. The author thanks the Deutsche Forschungsgemeinschaft for its support, and is indebted to Georg Bock (University of Heidelberg) and Claus Führer (DLR Oberpfaffenhofen) for many fruitful discussions and helpful comments.

REFERENCES

- [1] T. ALISHENAS, *Zur numerischen Behandlung, Stabilisierung durch Projektion und Modellierung mechanischer Systeme mit Nebenbedingungen und Invarianten*, Ph.D. thesis, Department of Numerical Analysis and Computer Science, Royal Institute of Technology, Stockholm, Sweden, 1992.
- [2] J. BAUMGARTE, *Stabilization of constraints and integrals of motion in dynamical systems*, Comp. Methods Appl. Mech., 1 (1972), pp. 1–16.
- [3] K. BRENNAN, *Stability and convergence of backward difference approximations for higher index differential-algebraic systems with applications in trajectory control*, Ph.D. thesis, University of California, Los Angeles, CA, 1983.
- [4] K. BRENNAN, S. CAMPBELL, AND L. PETZOLD, *The Numerical Solution of Initial Value Problems in Ordinary Differential-Algebraic Equations*, North Holland, Amsterdam, 1989.
- [5] E. EICH, *Projizierende Mehrschrittverfahren zur numerischen Lösung der Bewegungsgleichungen technischer Mehrkörpersysteme mit Zwangsbedingungen und Unstetigkeiten*, Ph.D. thesis, Institut für Mathematik, Universität Augsburg, 1991; also: VDI-Fortschrittsberichte, Reihe 18, No. 109, VDI-Verlag, Düsseldorf, Germany, 1992.
- [6] E. EICH, C. FÜHRER, B. LEIMKÜHLER, AND S. REICH, *Stabilization and Projection Methods for Multibody Dynamics*, Tech. Rep. 279, Helsinki University of Technology, Helsinki, Finland, 1990.
- [7] E. EICH AND M. HANKE, *Regularization Methods for Constrained Mechanical Systems*, Tech. Report 91-8, Humboldt-Universität zu Berlin, Germany, 1991.
- [8] C. FÜHRER, *Differential-algebraische Gleichungssysteme in mechanischen Mehrkörpersystemen*, Ph.D. thesis, Mathematisches Institut, Technische Universität München, Munich, Germany, 1988.
- [9] C. FÜHRER AND B. LEIMKÜHLER, *A new class of generalized inverses for the solution of discretized Euler–Lagrange equations*, in NATO Advanced Research Workshop on Real-Time Integration Methods for Mechanical System Simulation, E. Haug and R. Deyo, eds., Springer-Verlag, Heidelberg, 1990.
- [10] ———, *Numerical solution of differential-algebraic equations for constrained mechanical motion*, Numer. Math., 59 (1991), pp. 55–69.
- [11] C. GEAR, *Numerical Initial Value Problems in Ordinary Differential Equations*, Prentice-Hall, Englewood Cliffs, NJ, 1971.
- [12] ———, *Maintaining solution invariants in the numerical solution of ODEs*, SIAM J. Sci. Statist. Comput., 7 (1986), pp. 734–743.
- [13] C. GEAR, G. GUPTA, AND B. LEIMKÜHLER, *Automatic integration of the Euler–Lagrange equations with constraints*, J. Comput. Appl. Math., 12 & 13 (1985), pp. 77–90.
- [14] M. KNORRENSCHILD, *Differential/algebraic equations as stiff ordinary differential equations*, SIAM J. Numer. Anal., 29 (1992), pp. 1694–1715.
- [15] H. KREISS, *Über implizite Differenzenmethoden für partielle Differentialgleichungen*, Numer. Math., 5 (1963), pp. 24–47.

- [16] P. LÖTSTEDT, *A Numerical Method for the Simulation of Mechanical Systems with Unilateral Constraints*, Tech. Report, The Royal Institute of Technology, Stockholm, Sweden, 1979.
- [17] ———, *On the Relation Between Singular Perturbation Problems and Differential-Algebraic Equations*, Tech. Report, Uppsala University, Uppsala, Sweden, 1985.
- [18] P. LÖTSTEDT AND L. PETZOLD, *Numerical solution of nonlinear differential equations with algebraic constraints I. Convergence results for BDF*, *Math. Comp.*, 46 (1986), pp. 491–516.
- [19] C. LUBICH, *Extrapolation Methods for Constrained Multibody Systems*, Tech. Reports, Universität Innsbruck, Institut für Mathematik und Geometrie, A-6020 Innsbruck, Austria, 1990.
- [20] G. OSTERMEYER, *Baumgarte stabilization for differential-algebraic equations*, in *NATO Advanced Research Workshop on Real-Time Integration Methods for Mechanical System Simulation*, R. Deyo and E. Haug, eds., Springer-Verlag, Heidelberg, Germany 1990.
- [21] W. RHEINOLDT, *Differential-algebraic systems as differential equations on manifolds*, *Math. Comp.*, 43 (1984), pp. 473–482.
- [22] L. SHAMPINE, *Conservation laws and the numerical solution of ODEs*, *Comput. Math. Appls.*, Part B., 12 (1986), pp. 1287–1296.
- [23] B. SIMEON, C. FÜHRER, AND P. RENTROP, *Differential-algebraic equations in vehicle system dynamics*, *Surv. Math. Ind.*, 1 (1991), pp. 1–37.
- [24] G. STRANG, *Accurate partial difference methods II—nonlinear problems*, *Numer. Math.*, 6 (1964), pp. 37–46.
- [25] J. YEN, *Numerical Methods for Constrained Equations of Motion in Mechanical System Dynamics*, Ph.D. thesis, The University of Iowa, Iowa City, IA, 1990.
- [26] ———, *Numerical solution of constrained equations of motion in mechanical system dynamics as solution of ODEs on manifolds*, *SIAM J. Numer. Anal.*, 30 (1993), pp. 553–568.