Pergamon

Computers Math. Applic. Vol. 32, No. 12, pp. 15–28, 1996

0898-1221/96 $15.00 + 0.00

PII: S0898-1221(96)00204-0

MANPAK: A Set of Algorithms for Computations on Implicitly Defined Manifolds

W. C. RHEINBOLDT
Department of Mathematics and Statistics
University of Pittsburgh, Pittsburgh, PA 15260, U.S.A.

(Received July 1996; accepted August 1996)

Abstract—Mathematical models often involve differentiable manifolds that are implicitly defined as the solution sets of systems of nonlinear equations. The resulting computational tasks differ considerably from those arising for manifolds defined in parametric form. Here a collection of algorithms is presented for performing a range of essential tasks on general, implicitly specified submanifolds of a finite dimensional space. This includes algorithms for determining local parametrizations and their derivatives, and for evaluating quantities related to the curvature with sensitivity measures. The methods have been implemented as a FORTRAN 77 package, called MANPAK.

Keywords—Nonlinear equations, Manifolds, Implicit definition, Computational algorithms, FORTRAN package.

1. INTRODUCTION

Mathematical models of many, practically important scientific and technical problems involve differentiable manifolds that are implicitly defined as the solution sets of systems of nonlinear equations. For example, the computational study of equilibria typically leads to nonlinear systems of the form

$$F(z,\lambda) = 0, \qquad F : \mathbb{R}^m \times \mathbb{R}^d \mapsto \mathbb{R}^m, \tag{1.1}$$

where F is a sufficiently smooth mapping, $z \in \mathbb{R}^m$ a state variable, and $\lambda \in \mathbb{R}^d$ a parameter vector. Here, interest often centers on determining the behavior of the solutions under variation of λ. Under simple conditions (see Section 2 below), the zero set $M = \{(z,\lambda) \in \mathbb{R}^m \times \mathbb{R}^d : F(z,\lambda) = 0\}$ has the structure of a submanifold of dimension d of the product $\mathbb{R}^m \times \mathbb{R}^d$ of the state and parameter space. Then we are faced with a computational analysis of particular features of this manifold M, such as, for instance, of certain types of singular points on M or of the curvature behavior of M. Another example arises in connection with equality constrained dynamical systems that are modelled by differential-algebraic equations (DAEs). Such DAEs are known to be closely related to ordinary differential equations (ODEs) on implicitly defined differentiable manifolds.

For these, and similar problems, efficient numerical methods are required for computations on implicitly defined submanifolds of $\mathbb{R}^n$. These tasks differ considerably from those encountered, for example, in computer graphics, where curves or surfaces, i.e., submanifolds of $\mathbb{R}^3$, are considered

The work was supported in part by ONR-Grant N-00014-96-1-0235, and NSF-Grant CCR-9203488.

that are explicitly specified in a parametric form $M = \{x \in \mathbb{R}^3 : x = \varphi(y), y \in \mathbb{R}^d\}$ with $d = 1$ or $d = 2$, respectively. In fact, one of the basic computational problems arising in connection with any implicitly defined manifold is exactly the construction of such parametrizations and their derivatives which requires the solution of certain systems of nonlinear equations. Other related problems concern, for instance, the computation of the curvature of the manifolds or of the sensitivity of the solutions under specific changes.

This paper presents numerical methods for performing these and related tasks on general, implicitly specified submanifolds of a finite dimensional space. The methods have been collected in a FORTRAN 77 package, called MANPAK. All routines use reverse communication to avoid calls to subroutines for the evaluation of user-defined functions. The package is intended for applications to small or medium-sized problems, mainly because they involve many dense matrix computations. Some examples for use of the algorithms are noted here, and in addition, we refer to the companion paper [1] for applications to differential-algebraic equations.

2. BACKGROUND

For ease of reference, this section collects some basic definitions and theorems about submanifolds of $\mathbb{R}^n$. For details and proofs see, e.g., the texts [2,3].

Let $F : \mathbb{R}^n \mapsto \mathbb{R}^m$ be of class C^k, $k \geq 0$, on an open set $E \subset \mathbb{R}^n$; that is, assume that F is continuous and that all its partial derivatives of order at most k exist and are continuous on E. For $k \geq 1$, F is an *immersion* or *submersion* at a point $x \in E$ if its first derivative $DF(x) \in \mathcal{L}(\mathbb{R}^n, \mathbb{R}^m)$ is a one-to-one (linear) mapping or a mapping onto $\mathbb{R}^m$, respectively. We call F a submersion, or immersion on a subset S of E if it has that property at each point of S. These definitions obviously require that $n \leq m$ for F to be an immersion and $n \geq m$ for it to be a submersion. Clearly, if $n \geq m$ and $DF(x)$ has maximal rank m, then F is a submersion at x.

A nonempty subset $M \subset \mathbb{R}^n$ is a *submanifold* of $\mathbb{R}^n$ of dimension d and class C^k, if for every $x_c \in M$ there exists an open neighborhood $\mathcal{V}^n$ of x_c in $\mathbb{R}^n$ and a submersion $F : \mathcal{V}^n \mapsto \mathbb{R}^{n-r}$ of class C^k such that $M \cap \mathcal{V}^n = \{x \in \mathcal{V}^n : F(x) = 0\}$. In particular, if $F : E \mapsto \mathbb{R}^m$, $n - m = d > 0$, is of class C^k on an open set $E \subset \mathbb{R}^n$ and a submersion on $M := \{x \in E : F(x) = 0\}$, then M is a d-dimensional C^k submanifold of $\mathbb{R}^n$. Any nonempty, (relatively) open subset of a d-dimensional C^k-submanifold of $\mathbb{R}^n$ is itself a C^k submanifold of $\mathbb{R}^n$ of the same dimension.

For the analysis of submanifolds of $\mathbb{R}^n$ we need local parametrizations. Let M be any nonempty subset of $\mathbb{R}^n$. A local d-dimensional C^k *parametrization* of M is a pair $(\mathcal{V}^d, \varphi)$ consisting of a nonempty, open subset $\mathcal{V}^d$ of $\mathbb{R}^d$ and a C^k mapping $\varphi : \mathcal{V}^d \mapsto \mathbb{R}^n$ such that $\varphi(\mathcal{V}^d)$ is (relatively) open in M, φ is a homeomorphism of $\mathcal{V}^d$ onto its image $\varphi(\mathcal{V}^d)$, and φ is an immersion on $\mathcal{V}^d$. For any point x_c of M such that $x_c \in \varphi(\mathcal{V}^d)$ we call $(\mathcal{V}^d, \varphi)$ a local d-dimensional C^k parametrization of M near x_c. A nonempty subset $M \subset \mathbb{R}^n$ is a d-dimensional C^k submanifold of $\mathbb{R}^n$ if and only if for every $x_c \in M$ there exists a local d-dimensional C^k parametrization of M near x_c. If M is a d-dimensional C^k submanifold of $\mathbb{R}^n$, and $(\mathcal{V}^r, \varphi)$ a local r-dimensional C^k parametrization of M, then, necessarily, $r = d$.

Instead of defining tangent spaces in general, we use here the following characterization: let M be a d-dimensional C^k submanifold of $\mathbb{R}^n$. For any point $x_c \in M$ we can choose, by definition, an open neighborhood $\mathcal{V}^n \subset \mathbb{R}^n$ of x_c and a submersion $F : \mathcal{V}^n \mapsto \mathbb{R}^{n-d}$ at x_c such that $M \cap \mathcal{V}^n = \{x \in \mathcal{V}^n : F(x) = 0\}$. Then, it can be shown that the d-dimensional linear subspace $S = \ker DF(x_c)$ of $\mathbb{R}^n$ is independent of the particular choice of the local submersion F; that is, S depends only on M and the particular point. This space S is the *tangent space* of M at x_c and is denoted by $T_{x_c}M$. The subset $TM = \cup_{x \in M}[\{x\} \times T_xM]$ of $\mathbb{R}^n \times \mathbb{R}^n$ is the *tangent bundle* of M.

Since $T_x\mathbb{R}^n = \mathbb{R}^n$ for every $x \in \mathbb{R}^n$, we have $T\mathbb{R}^n = \mathbb{R}^n \times \mathbb{R}^n$. Thus, the tangent bundle of a submanifold M of $\mathbb{R}^n$ appears as a subset of $T\mathbb{R}^n$. In general, TM is a submanifold of $T\mathbb{R}^n = \mathbb{R}^n \times \mathbb{R}^n$. More specifically, if M be a d-dimensional C^k submanifold of $\mathbb{R}^n$ with $k \geq 2$,

then, TM is a $2d$-dimensional C^{k-1} submanifold of $T\mathbb{R}^n = \mathbb{R}^n \times \mathbb{R}^n \cong \mathbb{R}^{2n}$. For $k \geq 2$ the local parametrizations of the C^{k-1} submanifold TM of $\mathbb{R}^{2n}$ can be constructed easily from local C^k parametrizations of M. Let M be a d-dimensional C^k submanifold of $\mathbb{R}^n$, $k \geq 2$, and $(x_c, v_c) \in TM$. Then for any local C^k parametrization $(\mathcal{V}^d, \varphi)$ of M near x_c, the pair $(\mathcal{V}^d \times \mathbb{R}^d, (\varphi, D\varphi))$ is a C^{k-1} local parametrization of TM near (x_c, v_c).

3. COMPUTATION OF LOCAL PARAMETRIZATIONS

This section presents algorithms for computing local parametrizations on submanifolds of $\mathbb{R}^n$ which are implicitly defined by local submersions. Some of the material was given earlier, in part, in [4–6]. In view of the local nature of the methods, there is no loss of generality to restrict attention to the case of a single submersion.

ASSUMPTION A. *With positive integers d, k, m, n such that $m = n - d$, $n > d$, let $F : E \mapsto \mathbb{R}^m$ be a C^k mapping on an open subset E of $\mathbb{R}^n$ and a submersion on $M = F^{-1}(0) = \{x \in E : F(x) = 0\}$, whence, M is a d-dimensional C^k submanifold of $\mathbb{R}^n$.*

The following result exhibits a method for the computation of a local parametrization on M.

THEOREM 3.1. *Under Assumption A, let $U \in \mathcal{L}(\mathbb{R}^d, \mathbb{R}^n)$ be a linear isomorphism from $\mathbb{R}^d$ onto a d-dimensional linear subspace $T \subset \mathbb{R}^n$. Denote by $U^* \in \mathcal{L}(\mathbb{R}^n, \mathbb{R}^d)$ the adjoint of U and by $J : \mathbb{R}^d \mapsto \mathbb{R}^m \times \mathbb{R}^d$ the canonical injection that maps $\mathbb{R}^d$ isomorphically onto $\{0\} \times \mathbb{R}^d$. Then the C^k mapping $H : E \mapsto \mathbb{R}^m \times \mathbb{R}^d$, defined by $H(x) = (F(x), U^*x)$ for $x \in E$, is a local diffeomorphism near a point $x_c \in M$ if and only if*

$$T_{x_c}M \cap T^{\perp} = \{0\}. \tag{3.1}$$

If (3.1) holds at x_c then there exists an open set $\mathcal{V}^d$ of $\mathbb{R}^d$ such that the pair $(\mathcal{V}^d, \varphi)$, defined with the mapping $\varphi = H^{-1} \circ J : \mathcal{V}^d \mapsto \mathbb{R}^n$, is a local C^k parametrization of M near x_c.

PROOF. By Assumption A, H is of class C^k on E. Evidently, $DH(x_c)h = 0$ for any $h \in \mathbb{R}^n$, requires that $DF(x_c)h = 0$ and $U^*h = 0$, whence, $h \in T_{x_c}M$, and because of

$$\langle h, Uy \rangle = \langle U^*h, y \rangle = 0, \qquad y \in \mathbb{R}^d, \tag{3.2}$$

that $h \in T^{\perp}$ which by (3.1) implies that $h = 0$. Conversely, if there exists a nonzero $h \in T_{x_c}M \cap T^{\perp}$, then $DF(x_c)h = 0$ and (3.2) requires that $U^*h = 0$ which together shows that $DH(x_c)h = 0$. Hence, there is some open neighborhood $\mathcal{V}^n$ of x_c in $\mathbb{R}^n$ such that H is a diffeomorphism from $\mathcal{V}^n$ onto the open set $H(\mathcal{V}^n)$ in $\mathbb{R}^n$. Evidently, the set $H(M \cap \mathcal{V}^n) = H(\mathcal{V}^n) \cap (\{0\} \times \mathbb{R}^d)$ is open in $\{0\} \times \mathbb{R}^d$ and $J^{-1}H(M \cap \mathcal{V}^n) = \mathcal{V}^d \subset \mathbb{R}^n$ is an open subset of $\mathbb{R}^d$. This shows that $\varphi = H^{-1} \circ J$ is a C^k mapping from $\mathcal{V}^d$ onto the open subset $M \cap \mathcal{V}^n$ of M. Both φ and its inverse $J^{-1} \circ H_{|M \cap \mathcal{V}^n}$ are continuous, and hence, φ is a homeomorphism of $\mathcal{V}^d$ onto $M \cap \mathcal{V}^n$. Since both H^{-1} and J are immersions, the same is true for φ. Thus, altogether, φ is a local d-dimensional C^k parametrization of M near x_c. ∎

Note that (3.1) is equivalent with $\ker DF(x_c) \cap T^{\perp} = \{0\}$ or $\operatorname{rge} DF(x_c)^{\top} \cap T = \{0\}$. At any point $x_c \in M$ an obvious choice for a subspace T satisfying (3.1) is, of course, $T = T_{x_c}M$. For ease of reference we introduce the following terminology.

DEFINITION 3.3. *Under Assumption A a d-dimensional linear subspace $T \subset \mathbb{R}^n$ is a coordinate subspace of M at $x_c \in M$ if (3.1) holds, otherwise x_c is a foldpoint of M with respect to T. In the case $T = T_{x_c}M$, we speak of the tangential coordinate space of M at the point x_c.*

Theorem 3.1 readily becomes a computational procedure for local parametrizations by the introduction of bases. On $\mathbb{R}^n$ and $\mathbb{R}^d$ the canonical bases $e_1^n, \ldots, e_n^n$ and $e_1^d, \ldots, e_n^d$, respectively, will be used and we assume that the vectors $u_1, \ldots, u_d \in \mathbb{R}^n$ form an orthonormal basis of the given coordinate subspace T of M at x_c. Then the matrix representation of the mapping U is

the $n \times d$ matrix with the vectors $u_1, \ldots, u_d$ as columns. We denote this matrix again by U. It is advantageous to shift the open set $\mathcal{V}^d$ such that $\varphi(0) = x_c$. Then, in component form, the nonlinear mapping H assumes the form

$$H : \mathbb{R}^n \mapsto \mathbb{R}^n, \qquad H(x) = \begin{pmatrix} F(x) \\ U^\top (x - x_c) \end{pmatrix}, \quad \forall x \in E \subset \mathbb{R}^n, \tag{3.3}$$

where $F(x)$ is the column vector consisting of the m components of F evaluated at x. By definition of φ we have

$$H(\varphi(y)) = Jy, \qquad \forall y \in \mathcal{V}^d. \tag{3.4}$$

Thus, the evaluation of $x = \varphi(y)$ for given $y \in \mathcal{V}^d$ requires the solution of the nonlinear system of equations

$$H(x) \equiv \begin{pmatrix} F(x) \\ U^\top (x - x_c) \end{pmatrix} = \begin{pmatrix} 0 \\ y \end{pmatrix}. \tag{3.5}$$

Since (3.1) is assumed to hold at $x_c \in M$, the Jacobian

$$DH(x) = \begin{pmatrix} DF(x) \\ U^\top \end{pmatrix} \tag{3.6}$$

is nonsingular in an open neighborhood of $x = x_c$.

Experience has shown, that for the solution of the nonlinear system (3.5), a chord Newton method works well in practice. It is advantageous to start with $x^0 = x_c + Uy$, which allows the process to be applied in the y-independent form

$$x^{j+1} = G\left(x^j\right), \quad j = 0, 1, 2, \ldots, \qquad G(x) = x - A^{-1} \begin{pmatrix} F(x) \\ 0 \end{pmatrix}, \quad A = DH(x_c). \tag{3.7}$$

We sketch briefly its convergence properties. For given $\epsilon > 0$ such that $\|A^{-1}\|\epsilon \leq 1/2$ there exists, by Assumption A, a $\delta > 0$ such that the closed ball $\bar{B} \equiv \bar{B}(x_c \delta)$ is contained in E, and that $\|DF(x) - DF(x_c)\|_2 \leq \epsilon$, for all $x \in \bar{B}$. Then

$$\|DG(x)\|_2 \leq \left\|A^{-1}\right\|_2 \|DF(x_c) - DF(x)\|_2 \leq \frac{1}{2}, \qquad \forall x \in \bar{B}$$

shows that G is contractive on $\bar{B}$. Moreover, for $x \in \bar{B}$ it follows from

$$\begin{aligned} \|G(x) - G(x_c)\|_2 &\leq \left\|A^{-1}\right\|_2 \left\| \int_0^1 DF\left(x_c + s(x - x_c)\right)(x - x_c)\, ds \right\|_2 \\ &\leq 2\epsilon \left\|A^{-1}\right\|_2 \|x - x_c\|_2 \leq \|x - x_c\|_2 \end{aligned}$$

that G maps $\bar{B}$ into itself. Recall from the proof of Theorem 3.1 that H is a diffeomorphism from an open neighborhood $\mathcal{V}^n$ of x_c onto its image. Let δ be sufficiently small that $\bar{B} \subset \mathcal{V}^n$. Then it follows from the contraction theorem that for any $y \in \mathcal{V}^d$, the process (3.6), started from $x^0 = x_c + Uy$, converges to the unique fixed point $x^* \in \bar{B}$ of G. Clearly, $F(x^*) = 0$, and from

$$U^\top (x - x_c) = U^\top \left(x^0 - x_c\right) + \sum_{j=0}^{\infty} U^\top \left(x^{j+1} - x^j\right) = y,$$

we obtain that $H(x^*) = 0$, and therefore, in view of $x^* \in \mathcal{V}^n$, that $x^* = \varphi(y)$.

This shows that, for any local vector y near the origin of $\mathbb{R}^d$, the following algorithm produces the point $x = \varphi(y)$ in the local parametrization $(\mathcal{V}^d, \varphi)$ near x_c defined by Theorem 3.1.

GPHI Input: Center point $x_c \in M$ of the local parametrization, local vector $y \in \mathbb{R}^d$, basis matrix U, Jacobian $DF(x_C)$, tolerances;

$x := x_c + Uy$;

Compute the LU factorization of $A = \begin{pmatrix} DF(x_c) \\ U^\top \end{pmatrix}$;

While: iterates do not meet tolerances

Return for the evaluation of $F(x)$;

$q := \begin{pmatrix} F(x) \\ 0 \end{pmatrix}$;

Solve $Aw = q$ for $w \in \mathbb{R}^n$;

$x := x - w$;

End While

Output: $\varphi(y) := x$.

For the sake of clarity, our reverse communication paradigm was here, only indicated by the statement "Return for the evaluation of $F(x)$." The MANPAK implementation of GPHI uses a different step-order for handling the repeated returns to the calling program. Of course, a full Newton method or some other iterative process can be applied as well. However, experience has shown that the faster convergence of Newton's method at the expense of several evaluations and factorizations of DH does not improve the overall performance.

So far, it was assumed that T is a given coordinate subspace of M at x_c and that an orthonormal basis of T is available. For the construction of T we use a simple reformulation of (3.1): let $Z = T^\perp$ be the orthogonal complement of T (under the canonical inner product of $\mathbb{R}^n$), then (3.1) implies that $Z \oplus T_{x_c}M = \mathbb{R}^n$; that is, that Z is also a complementary subspace of $T_{x_c}M$. Thus, in order to ensure the validity of (3.1), it is advantageous to construct T by choosing a complementary linear subspace Z of $T_{x_c}M$ and to determine T as the orthogonal complement of Z.

As before, we use the canonical bases on $\mathbb{R}^d$ and $\mathbb{R}^n$ and assume, for the moment, that a basis $z_1, \ldots, z_m$ of Z is available. Then, T is the nullspace of the $n \times m$ matrix formed with these vectors as its columns. We denote this matrix again by Z. Our task is now to compute an orthogonal basis of the nullspace of the transposed matrix $Z^\top$ for which obviously rank $Z^\top = m$. There are several approaches for this; probably the simplest one is based on the LQ-factorization (with row pivoting)

$$Z^\top = P^\top \begin{pmatrix} L & 0 \end{pmatrix} Q^\top, \qquad Q = (Q_1, Q_2)\,. \tag{3.8}$$

Here P is an $m \times m$ permutation matrix, L an $m \times m$ nonsingular lower triangular matrix, and Q an $n \times n$ orthogonal matrix partitioned such that Q_1 and Q_2 are $n \times m$ and $n \times d$ matrices, respectively. Then, clearly, the d columns of Q_2 form the desired orthonormal basis of T. This justifies the following MANPAK algorithm.

COBAS Input: $m \times n$ matrix $Z^\top$ of rank m;

Compute the LQ-factorization (3.8) of $Z^\top$ with row-pivoting;

For $j = 1, \ldots, d$ **Do:** $u_j := Q_2^\top e_j^d$;

Output: $U := (u_1, \ldots, u_d)$.

Other algorithms for the computation of nullspace-bases of $m \times n$ matrices are given, for example, in [7,8].

Obviously, when the tangential coordinate system is used at x_c, then COBAS can be applied with the Jacobian matrix $DF(x_c)$ as the matrix $Z^\top$. In that case, GPHI simplifies considerably if the LQ-factorization (3.8) of $DF(x_c)$ is applied for the solution of the corrector equation $Aw = q$ in GPHI. In fact, this equation has the block-components $DF(x)w = F(x)$ and $U^\top w = 0$, which with (3.8) and $U = Q_2$, can be rewritten as $LU^\top w = PF(x)$ and $U^\top w = 0$. Thus, in this case, the algorithm can be modified as follows.

TPHI Input: Center point $x_c \in M$ of the local parametrization, local vector $y \in \mathbb{R}^d$, the LQ factorization (3.8) of $DF(x_c)$, tolerances;

$x := x_c + Q_2^\top y$;

While: iterates do not meet tolerances

Return for the evaluation of $F(x)$;
Solve $Lv = PF(x)$ for $v \in \mathbb{R}^m$;
$x := x - Q_1^\top v$;
End While
Output: $\varphi(y) := x$.

Thus, for each iteration step we need to solve now only an $m \times m$, rather than an $n \times n$ lower-triangular. The convergence behavior, of course, remains the same.

In COBAS the transposed basis matrix $Z^\top$ of some complementary space of $T_{x_c}M$, was assumed to be available. In order to compute such a matrix it is natural to start with the matrix representation $DF(x_c)$ of the Jacobian of F at x_c for which the rows form a basis of $(T_{x_c}M)^\perp$. We construct bases of complementary spaces of $T_{x_c}M$ by replacing suitable columns vectors of $DF(x_c)$.

In applications, it is frequently important to work with coordinate spaces T that contain a specific canonical basis vector, say, e_ℓ^n of $\mathbb{R}^n$. Often the reason for this is that the independent variable x_ℓ represented by this vector is of a special nature, as for instance, when x_ℓ corresponds to the time variable in some nonautonomous DAE. Evidently, in order to ensure that $e_\ell^n \in T$ we have to replace the ℓ^{th} column of $DF(x_c)$ by a zero column. This leads to the following MANPAK algorithm.

GNBAS Input: The Jacobian matrix $DF(x_c)$, index $\ell \in \{1, \ldots, n\}$;
Form $Z^\top$ by zeroing column ℓ of $DF(x_c)$;
Use COBAS to compute the orthonormal basis U of of ker W;
OUTPUT The basis matrix U.

Of course, this algorithm requires that the constructed matrix $Z^\top$ still has maximal rank m. In order to verify this, while computing the basis of the nullspace, we may replace the LQ-factorization of $Z^\top$ in COBAS by the singular value decomposition (SVD) and then apply scaling and standard rank-tests (see, e.g., [9]). Such a version of the algorithm was not included in MANPAK since it appears to be rarely needed.

In certain applications, it is desirable to work with coordinate spaces spanned by d suitably chosen canonical basis vectors $e_{i_1}^n, \ldots, e_{i_d}^n$ of $\mathbb{R}^n$, (see, e.g., [10,11]). In this case, the space $Z^\top$ should be spanned by $n - d$ canonical basis vectors of $\mathbb{R}^n$. These vectors can be obtained from the QR-factorization of the Jacobian $DF(x_c)$ with column pivoting. The resulting permutation selects m column-vectors of the Jacobian such that the $m \times m$ matrix formed by these columns is nonsingular. The indices of these selected columns correspond to the desired m canonical basis vectors of $\mathbb{R}^n$ spanning the space Z. This leads to the following algorithm.

GCBAS Input: The Jacobian matrix $DF(x_c)$
Compute the QR-factorization $DF(x_c)P = Q(R, S)$;
$k_i := 0$, $\forall i = 1, \ldots, n$;
For $j = 1, \ldots, m$ **Do:** **If** $e_i^m = Pe_j^m$ **Then** $k_i = 1$;
$\ell = 1$;
For $i = 1, \ldots, n$ **Do:** **If** $k_i = 0$ **Then** $z_\ell = e_i^n$; $\ell := \ell + 1$;
Output $Z := (z_1, \ldots, z_m)$.

4. SIMPLICIAL APPROXIMATIONS AND MOVING FRAMES

Continuation methods are probably the oldest methods for the computational analysis of an implicitly defined manifold M, although this fact is often not noticed. They apply to problems of the form (1.1) with a parameter space of dimension $d = 1$, in which case the solution manifold M is one-dimensional. All continuation methods begin from a given point $x^0 \in M$ and produce a sequence of points x^j, $j = 0, 1, 2, \ldots$, on M. In general, the step from x^j to x^{j+1} corresponds to an implementation of a local parametrization of M. More specifically, a one-dimensional

coordinate subspace $T = \text{span}\{u\}$ at x^k is chosen, a predicted point w is selected, and with the corresponding local coordinate $y = w^\top u$ as input, a local parametrization algorithm, such as GPHI or TPHI, is applied to generate the next point x^{j+1} on M. The various continuation methods differ in the choice of:

(i) the coordinate direction u;
(ii) the construction of the predicted point w; and
(iii) the form of the iterative process used in the local parametrization algorithm.

In the case of a multidimensional, implicitly defined manifold M, it is obviously difficult to achieve a good assessment of the features of M solely from information along some paths. This led to the development of several methods for the approximation of subsets of such a manifold (see [12] for a recent survey). In particular, [4] presented a first method for computing a simplicial approximation Σ of a neighborhood of a given point $x^0 \in M$ consisting of a grid of points $x^k \in M$, $k = 0, 1, \ldots, k$, and their connectivity pattern. This method was globalized in [13] to allow for the computation of a simplicial approximation covering a specified subset of M. The algorithms in [4,13], were restricted to the case $d = 2$. Recently, in [12], an extension to the case of any $d \geq 2$ was developed, and some applications and other related algorithms were discussed.

The methods in the cited articles are similar to a continuation method. At a point x^j on the current "frontier" of the already computed part of Σ a tangential coordinate space of M is chosen, and by means of a tangential local parametrization of M at x^j, the already computed neighboring simplices of Σ incident with x^j are mapped onto the tangent space $T_{x^j}M$. In addition, a reference triangulation patch of $\mathbb{R}^d$ is introduced on $T_{x^j}M$ and matched to the open facets of the already existing part of Σ. This produces a set of points in the unfilled gap of the triangulation at x^j which are then used to complete the triangulation of the neighborhood of the point on T_xM. While for $d = 2$ this is a relatively simple task, for $d > 2$ a Delauney triangulation process was needed in [12]. The nodes of the completed local mesh around x^j are then mapped onto M using the local parametrization algorithm TPHI and the resulting points and their incidence relations are added to Σ.

During the process it is important to align the orientations of the computed simplices. In the setting of the simplicial approximation algorithms this can be accomplished by setting the orientation of any newly computed simplex equal to that of one of its neighbors. This was the approach chosen in [12] where also provisions were made for the resolution of any conflicts.

This situation is a special case of the more general problem of matching the basis orientations of the local parametrization at any neighboring points. In differential geometry this corresponds to the concept of a moving frame; that is, of a mapping that associates with each point x of a d-dimensional manifold M an ordered basis, $\{u_1, \ldots, u_d\}$ of T_xM such that the mappings $u_i : M \mapsto TM$, $i = 1, \ldots, d$, form d vector fields on M. A manifold is parallelizable if such a moving frame exists on all of M.

In computational problems it is often important to generate a moving frame on an open neighborhood of a given point of M. Such an algorithm was developed in [4] and applied in the mentioned simplicial approximation method for two-dimensional manifolds.

Under Assumption A, suppose that orthonormal bases of $T_{x_1}M$ and $T_{x_2}M$ are needed at neighboring points x_1, $x_2 \in M$. The MANPAK algorithm COBAS uses the LQ-factorization of the Jacobians $DF(x_1)$ and $DF(x_2)$ to determine the desired $m \times n$ basis matrices U_1 and U_2, respectively. But, as noted in [14] the QR-factorization algorithm (and hence, the LQ-factorization algorithm) need not produce matrices that depend continuously on the elements of the given matrix. In other words, the computed basis U_2 at x_2 need not converge to the basis U_1 at x_1 when x_2 tends to x_1. This observation extends to other algorithms for computing the nullspace of $DF(x)$ for neighboring x. In fact, it relates directly to the well-known loss of continuity under changes of the matrix elements in the computation of eigenvectors associated with a multiple eigenvalue.

Let M_0 be an open subset of M, and suppose that $T \subset \mathbb{R}^d$ is a coordinate subspace at each point of M_0. Assume that U_0 is an orthonormal basis matrix of T and that at any $x \in M_0$ an orthonormal basis matrix $U(x)$ of T_xM is chosen. Let $V(x) = A(x)\Sigma B(x)^\top$ be the singular value decomposition of the $d \times d$ matrix $V(x) = U(x)^\top U_0$, and form the product $Q(x) = A(x)B(x)^\top$ of the matrices of left and right singular vectors. Then, it was proved in [4] that the mapping

$$x \in M_0 \longrightarrow U(x)Q(x) \in \mathcal{L}\left(\mathbb{R}^d, \mathbb{R}^n\right)$$

is of class C^{k-1} on M_0 and defines an orthonormal moving frame on M_0.

This leads to the following MANPAK algorithm first given in [4]:

MOVFR Input: Orthonormal tangent basis matrices U_0 and U_x;
For $V = U_x^T U_0$ compute the SVD $V = A\Sigma B^T$;
$Q = AB^T$;
$\hat{U}_x = U_x Q$;
Output: Rotated basis matrix $\hat{U}_x$.

The use of the SVD for the $d \times d$ matrix V of MOVFR may, of course, become costly if the dimension d of the manifold is larger. Thus, other algorithms for matching the orientation of computed bases are of interest. A simple approach is based on the use of the well-known greedy algorithm (see [15]) of combinatorial computing which can be applied to two orthonormal bases U_1 and U_2 at some neighboring points x_1, $x_2 \in M$ in the following form.

ORIENT Input: Orthonormal basis matrices U_1, U_2;
Compute the matrix $V = U_1^\top U_2 = (v_{ij},\ i,j \in \mathcal{J})$, $\mathcal{J} = \{1,\dots,d\}$;
$\omega(i) = 0$, $\forall i \in \mathcal{J}$;
For $i = 1,\dots,d$ **DO:**
Find the smallest $k = k_i \in \mathcal{J}$ such that $|v_{ik}| = \max\{|v_{ij}|,\ j \in \mathcal{J}\}$
and $\omega(k) = 0$;
If no such index k_i exists **Then** reorientation failed;
$\omega(k_i) = 1$;
If $v_{ik} < 0$ **Then** change the sign of all elements in the k_i^{th} column of U_2;
End If;
With the permutation $P \in \mathcal{L}(\mathbb{R}^d)$, $Pe_i^d = e_{k_i}^d$, $i \in \mathcal{J}$ form $\hat{U}_2 := PU_2$;
Output: Rotated basis matrix $\hat{U}_2$.

It can be shown that the algorithm may fail in cases when MOVFR is successful. However, this happens very rarely. Moreover, the problem can be resolved by incorporating in ORIENT a standard backtracking approach (loc.cit.). This modification has been implemented but was not included in MANPAK.

5. SENSITIVITY COMPUTATION

As noted in the introduction, many physical systems lead to mathematical models in the form (1.1) of parameter dependent equations. Then it is often assumed, especially in the engineering literature, that the solutions $x = (z, \lambda)$ of (1.1) can be written in the form $(z(\lambda), \lambda)$. Of course, under Assumption A, this requires that at $x \in M$ the parameter subspace Λ is a local coordinate space of M, which means that $\ker DF(x) \cap \Lambda^\top = \{0\}$ or, equivalently, that the partial derivative $D_Z F(x) \equiv DF(x)_{|Z}$ of F with respect to the state space is nonsingular. If this holds then, traditionally, the derivative

$$Dz(\lambda) = -D_Z F\left(z(\lambda), \lambda\right)^{-1} F\left(z(\lambda), \lambda\right) \tag{5.1}$$

is defined as the sensitivity measure of the particular solution under variation of the parameters (see, e.g., [16]). As noted, (5.1) is applicable only at solutions $x \in M$ which are not foldpoints of

M with respect to Λ. In many applications this is an undesirable restriction since exactly these foldpoints are of special interest. In fact, it is one of the fundamental observations of bifurcation theory that these are the points where the character of the solutions of (1.1) may undergo major changes. This led to the development of a more general sensitivity theory in [17] which applies generally on M.

Let $T \subset \mathbb{R}^n$ be a local coordinate space of M at a given point $x^0 \in M$ and denote by $(\mathcal{V}^d, \varphi)$ the induced local parametrization of M at x_c. Then $D\varphi(0)y$ represents the change of the solution x_c in the local coordinate direction $y \in T$. Let the columns of the $n \times m$ matrix V_m and of the $n \times d$ matrix V_d, form orthonormal bases of T and $T^\perp$, respectively. Then $V_m V_m^\top D\varphi(0)y$ is the orthogonal projection of $D\varphi(0)y$ onto $T^\perp$. It characterizes, in essence, the change due to the nonlinear nature of M; in fact, if M were flat; that is, if $M \subset T$, then this vector would be zero. In line with this, in [17], the linear mapping

$$\Sigma \in \mathcal{L}\left(\mathbb{R}^d, \mathbb{R}^n\right), \qquad \Sigma = V_m^\top D\varphi(0) \tag{5.2}$$

is defined as the sensitivity map of M at x_c with respect to the local coordinate space T.

In [17], it was shown that when the natural parameter space Λ is a local coordinate space at $x_c \in M$, then the new definition (5.2) reduces exactly to (5.1). More generally, consider besides the local coordinate space T at x_c also the tangential coordinate space $T_{x_c}M$. In analogy to V_m and V_d, let the columns of the $n \times m$ matrix U_m and of the $n \times d$ matrix U_d, form orthonormal bases of $T_{x_c}M$ and $T_{x_c}M^\perp$, respectively. Then the following relations were proved in [17]

$$\Sigma = V_m^\top U_d \left(V_d^\top U_d\right)^{-1}, \tag{5.3}$$

$$\|\Sigma\|_2 = \frac{\operatorname{dist}\left(T, T_{x_c}M\right)}{\left[1 - \operatorname{dist}\left(T, T_{x_c}M\right)\right]^{1/2}}. \tag{5.4}$$

Here, as usual, the distance between any two equidimensional linear subspaces S_1 and S_2 of $\mathbb{R}^n$ is defined by $\operatorname{dist}(S_1, S_2) = \|P_1 - P_2\|_2$ where P_1, P_2 are the orthogonal projections onto S_1 and S_2, respectively.

In MANPAK two algorithms implement (5.3) and (5.4), respectively. In view of the noted engineering applications, both algorithms assume that the natural parameter space Λ is used as the local coordinate space, and more specifically, that Λ is spanned by the canonical basis vectors $e_{i_j}^n$, $j = 1, \ldots, d$, specified by a given index set $\mathcal{J} \subset \{1, \ldots, n\}$.

SENMAP Input: Index set $\mathcal{J} = \{i_1, \ldots, i_d\}$, orthonormal basis matrix U_d of $T_{x_c}M$;
Form the $d \times d$ matrix A with the columns $U_d^\top e_{i_j}^n$, $i_j \in \mathcal{J}$, $j = 1, \ldots, d$;
Compute the LU-factorization of A;
If A is numerically singular **then Output:** "Undefined sensitivity."
Determine the indices $k_j \notin \mathcal{J}$, $1 \le k_j \le n$, $j = 1, \ldots, m = n - d$;
For $j = 1, \ldots, m$ **Do:** Solve $Aw_j = U_d^\top e_{k_j}^n$;
Output: $\Sigma := (w_1, \ldots, w_m)^\top$.

SENNRM Input: Index set $\mathcal{J} = \{i_1, \ldots, i_d\}$, orthonormal basis matrix U_d of $T_{x_c}M$;
Form the $d \times d$ matrix A with the rows $(e_{i_j}^n)^\top U_d$, $j = 1, \ldots, d$;;
Compute the smallest singular value σ_d of A;
If $\sigma_d = 0$ **Then** $\zeta = \infty$ **Else** $\zeta = \sqrt{\frac{1}{\sigma_d^2} - 1}$;
Output: $\|\Sigma\|_2 := \zeta$.

The corresponding algorithms for the more general case of an arbitrary local coordinate space T were not included in MANPAK due to their infrequent applicability.

6. DERIVATIVES OF LOCAL PARAMETRIZATIONS

Let M be a d-dimensional C^k submanifold of $\mathbb{R}^n$ with $k \ge 2$ and $(\mathcal{V}^d, \varphi)$ a local C^k parametrization of M near $x_c \in M$. Then, as noted in Section 2, for any $v_c \in \mathbb{R}^n$ such that $(x_c, v_c) \in TM$,

the pair $(\mathcal{V}^d \times \mathbb{R}^d, (\varphi, D\varphi))$ is a C^{k-1} local parametrization of the tangent bundle near (x_c, v_c). This means that for the evaluation of the corresponding local parametrization of TM at that point we need an algorithm for computing the derivative $D\varphi$ of φ.

Under Assumption A, let T be a coordinate subspace of M at x_c. As before, suppose that on $\mathbb{R}^n$ and $\mathbb{R}^d$ the canonical bases are chosen, and that the columns of the $n \times m$ matrix U form an orthonormal basis of the given coordinate subspace T at x_c. Then by Theorem 3.1 the mapping (3.3) is a local diffeomorphism near $x_c \in M$ and the local parametrization $(\mathcal{V}^d, \varphi)$ induced by T satisfies (3.4). Thus, by differentiation of (3.4), it follows that

$$DH(\varphi(y))\, D\varphi(y)v = Jv, \qquad \forall y \in \mathcal{V}^d, \quad v \in \mathbb{R}^d. \tag{6.1}$$

Since the Jacobian (3.5) of H is nonsingular at x_c, this shows that at any $x = \varphi(y)$, $y \in \mathcal{V}^d$, the derivative $D\varphi(y)$ can be computed as follows.

DGPHI Input: Local coordinate basis U at x_c, Jacobian $DF(x)$ at a neighboring point $x = \varphi(y)$;

Compute the LU-factorization of $A := \begin{pmatrix} DF(x) \\ U^\top \end{pmatrix}$;

For $j = 1, \ldots, d$ **Do:** Solve $Az_j = e^n_{d+j}$;

Output: $D\varphi(y) := (z_1, \ldots, z_d)$.

In line with our general format, the algorithm assumes that the Jacobian $DF(x)$ has already been evaluated by the calling program.

In the case of a tangential coordinate space, we obtained the modified algorithm TPHI based on the LQ-factorization of $DF(x_c)$. This reduced the cost of the GPHI algorithm. For the evaluation of the Jacobian of φ the use of the LQ-factorization is not as advantageous. Suppose again that we want to compute $D\varphi(y)$ at $x = \varphi(y)$. Moreover, let U_x and U_{x_c} be $n \times d$ matrices with orthogonal columns that form bases of T_xM and $T_{x_c}M$, respectively. With the matrix representation of $DH(x)$ used in DGPHI, the j^{th} column z_j of $D\varphi(y)$ satisfies $DF(x)z_j = 0$ and $U_{x_c}z_j = e^d_j$. Hence, we have $z_j = U_x y_j$ for some $y_j \in \mathbb{R}^d$ and $U^\top_{x_c} U_x y_j = e^d_j$ which implies that

$$D\varphi(y) = U_x \left[U^\top_{x_c} U_x\right]^{-1}. \tag{6.2}$$

In analogy to DGPHI this gives the following algorithm.

DTPHI Input: Tangent basis U_{x_c} at x_c, a neighboring point $x \in M$;

Use COBAS to evaluate U_x at x;

$A := U^\top_{x_c} U_x$;

For $j = 1, \ldots, d$ **Do:** Solve $Az_j = e^d_j$; $z_j := U_x z_j$;

Output: $D\varphi(y) := (z_1, \ldots, z_d)$.

Since the computation of the basis matrix U_x requires the application of COBAS, the use of DTPHI is, in general, more costly than that of DGPHI.

Suppose now that $k > 2$ in Assumption A. Then φ is at least twice differentiable at any $y \in \mathcal{V}^d$ and by differentiation of (6.1), we obtain

$$DH(\varphi(y))\, D^2\varphi(y)(v_1, v_2) = -D^2H(\varphi(y))(D\varphi(y)v_1, D\varphi(y)v_1), \qquad \forall v_1, v_1 \in \mathbb{R}^d \tag{6.3}$$

which, because of the nonsingularity of $DH(\varphi(y))$, defines $D^2\varphi(y)(v_1, v_2)$ uniquely. In line with our reverse communication paradigm, the following MANPAK algorithm of $D^2\varphi$ assumes that the vector $w = D^2F(x)(u_1, u_2)$ has already been computed for given $x = \varphi(y)$ and $u_i = D\varphi(y)v_i$, $i = 1, 2$.

D2GPHI Input: Tangent basis U_{x_c} at x_c, Jacobian $DF(x)$ at a neighboring point $x \in M$, the vector $w = D^2F(x)(u_1, u_2)$ with $u_i = D\varphi(y)v_i$, $i = 1, 2$;

Compute the LU-factorization of $A := \begin{pmatrix} DF(x) \\ U^\top \end{pmatrix}$;

Solve $Az = \begin{pmatrix} -w \\ 0 \end{pmatrix}$ for z;

Output: $D^2\varphi(y)(v_1, v_2) := z$.

7. CURVATURE AND THE SECOND FUNDAMENTAL TENSOR

A principal application of the algorithms for the computation of the derivatives of a local parametrization arises in connection with the solution of differential algebraic equations. This is discussed in the mentioned companion paper [1], and will not be further addressed here.

Another interesting use of the algorithm DGPHI occurs in certain constrained minimization methods. We indicate here, only briefly, the approach suggested in [18]. Under Assumption A with $k \geq 2$, suppose that $g : E \in \mathbb{R}^n \mapsto \mathbb{R}^1$ is a C^r functional $r \geq 2$, on some open set E and that $E \cap M$ is not empty. Consider the problem of computing a local minimizer $x^* \in M$ of g on M, and let $x \in M$ be a point on M that represents our current approximation of x^*. We introduce a local parametrization $(\mathcal{V}^d, \varphi)$ of M at x. Then the local representation $h = g \circ \varphi$ of g near x is at least of class C^2, and locally, the minimization of g on M is equivalent with the unconstrained minimization of h. This suggests the application of a trust region step to h in order to obtain a new approximation of x^*. For this, we approximate h by the quadratic functional

$$h_q(y) = h(0) + Dh(0)y + \frac{1}{2} D^2h(0)(y, y), \qquad y \in \mathbb{R}^d. \tag{7.1}$$

Here we have

$$Dh(0)y = Dg(x)D\varphi(0)y, \qquad D^2h(0)(y, y) = Dg(x)D^2\varphi(0)(y, y) + D^2g(x)\left(D\varphi(0)y, D\varphi(0)y\right),$$

which can be evaluated by means of the MANPAK routines DGPHI and D2GPHI. For details about the trust region step we refer, for example, to [19]. Clearly, in practice, the computation of the local Hessian $Dh(0)$ can be replaced by some update scheme.

The second derivative $D^2\varphi$ of a local parametrization has an important connection with the second fundamental tensor of the manifold M. This tensor is a concept of Riemannian geometry; that is, it requires a metric on the manifold M. We use here the metric induced by the canonical (Euclidean) inner product of $\mathbb{R}^n$. For a definition of this symmetric, vector valued tensor

$$V_x : T_xM \times T_xM \mapsto T_xM^\perp, \qquad x \in M, \tag{7.2}$$

in a setting similar to that used here, we refer to [20]. In lieu of a definition we cite only the following characterization of V_x proved in [21].

THEOREM 7.1. *Under Assumption A, let $x \in M$, Z any complement of T_xM, and Q the orthogonal projection onto $(T_xM)^\perp$. Then the component of the second fundamental tensor V_x of M at x in the tangential directions u^1, $u^2 \in T_xM$ is*

$$V_x\left(u^1, u^2\right) = -Q\left[DF(x)_{|Z}\right]^{-1} D^2F(x)\left(u^1, u^2\right). \tag{7.3}$$

Let T be a local coordinate space at $x \in M$ and $(\mathcal{V}^d, \varphi)$ the induced local parametrization of M. Moreover, set $Z = T^\perp$, and as before, let the columns of the $n \times d$ matrix U form an orthonormal basis of T. Note that at $x = \varphi(0)$ the equation (6.3) defining $D^2\varphi(0)$ requires that $DF(x)D^2\varphi(0)(v_1, v_2) = -D^2F(x)(u^1, u^2)$ and $U^\top D^2\varphi(0)(v_1, v_2) = 0$ with $u_i = D\varphi(0)v_i$, $i = 1, 2$. Evidently, this is equivalent with

$$\left[DF(x)_{|Z}\right] D^2\varphi(0)\left(v_1, v_2\right) = -D^2F(x)\left(u^1, u^2\right),$$

whence, it follows from (7.3) that

$$V_x(u^1,u^2) = QD^2\varphi(0)(v_1,v_2), \qquad u_i = D\varphi(0)v_i, \quad i=1,2.$$

Note that $D^2\varphi(0)(v_1,v_2) \in T^\perp$, and thus, $V_x(u^1,u^2) = D^2\varphi(0)(v_1,v_2)$ for the tangential coordinate space $T = T_xM$. Hence, this tensor component can be obtained by application of the MANPAK algorithm D2GPHI with a tangential coordinate space. However, if COBAS was used with $Z^\top = DF(x_c)$ to compute the tangent basis, we may proceed as in DTPHI and simplify the process as follows.

TSFT Input: The LQ-factorization $DF(x) = P^\top(L\ 0)Q^\top$,
the vector $w = D^2F(x)(u_1,u_2)$ with $u_i = D\varphi(y)v_i$, $i=1,2$;
Solve $Lz = P^\top w$ for $z \in \mathbb{R}^m$;
$z := Q_1 z$;
Output: $V_{x_c}(v_1,v_2) := z$.

The second fundamental tensor characterizes curvature properties of the manifold and is closely connected with the Riemann curvature tensor R of M. Accordingly, it is not surprising that it has numerous applications. For example, in [20] the tensor has been applied for computating bifurcation directions at certain foldpoints on an implicitly defined manifold. Then, in [21] it was used in an algorithm for solving the Euler-Lagrange equations arising in the modeling of constrained dynamical systems. In both cases, early forms of the MANPAK algorithms D2GPHI and TSFT were applied.

Note that it suffices to have a method for computing the diagonal components $V_x(u,u)$ since, by the bilinearity and symmetry of the tensor, we have

$$V_x(u^1,u^2) = 2V_x(u,u) - \frac{1}{2}\left(V_x(u^1,u^1) + V_x(u^2,u^2)\right),$$

for any u^1, $u^2 \in T_{x_c}M$ and $u = (1/2)(u^1+u^2)$. In [20], a geometrically based algorithm was given for approximating the diagonal component $V_x(u,u)$ for any $u \in T_xM$.

Let π be any path on M and x_ℓ, x_c, $x_r \in M$ three consecutive points along π that form a nondegenerate triangle. Then the curvature of the circumscribing circle of the triangle is given by Heron's formula

$$\kappa = \frac{4}{abc}\sqrt{s(s-a)(s-b)(s-c)}, \qquad s = \frac{a+b+c}{2}, \tag{7.4}$$

where $a = \|x_\ell - x_c\|_2$, $b = \|x_r - x_c\|_2$, $c = \|x_c - x_\ell\|_2$. When the outside points x_ℓ, x_r tend to the middle point x_c, then the circumscribing circle tends to the osculating circle of the path π at x_c, and the limit of κ is the curvature of π at x_c. It was shown in [20] that the value (7.4) of κ approximates $\|V_{x_c}(u,u)\|_2$ for the tangent vector $u \in T_{x_c}M$ of π at x_c. Moreover, the unit vector in the direction of $V_{x_c}(u,u)$ is approximated by the principal normal vector of the path π at x_c.

For the computation, it is useful to rewrite (7.4) as

$$\hat{\kappa} = \frac{1}{c}\left[\frac{1}{a_c} + \frac{1}{b_c}\right]\sqrt{1-\delta^2}\sin\alpha, \qquad \delta = a_c - b_c, \quad \alpha = \arccos\gamma, \quad \gamma = \frac{1}{a_c+b_c}.$$

This leads to the following MANPAK algorithm.

CURVT Input: Consecutive points x_ℓ, x_c, x_r along a path π on M,
tangent vector $v \in T_{x_c}M$ of π at x_c, machine precision ϵ
Evaluate $a := \|x_\ell - x_c\|_2$, $b := \|x_r - x_c\|_2$, $c := \|x_\ell - x_r\|_2$;
$u := v/\|v\|_2$;
$p := (1/a)(\ell - x_c) + (1/b)(x_r - x_c)$;

$p := p - (p^\top u)u$;
$a_c := a/c$; $b_c := b/c$;
$\delta := a_c - b_c$; $\gamma := 1/(a_c + b_c)$;
If $1 - \gamma < \epsilon$ **Then**
$\kappa := 0$;
Else
$\alpha :=$ arc cosγ;
$\kappa := (1/c)(1/ac + 1/b_c)\sqrt{1 - \delta^2}\sin\alpha$;
End If;
Output: p, κ; $V_{x_c}(u, u) := \kappa p$.

When $1 - \gamma$ falls below the machine precision ϵ then, in floating point arithmetic, α will be zero, and accordingly, we set $\kappa = 0$. The approximation p of the principal normal of the path at x_c is generated by a simple Gram-Schmidt orthonormalization step.

For some numerical examples involving CURVT we refer to [20].

REFERENCES

1. W.C. Rheinboldt, Solving algebraically explicit DAEs with the MANPAK-Manifold-Algorithms, Inst. for Comp. Math. and Appl., University of Pittsburgh, Tech. Rept. TR-ICMA-96-199, July 1996, *Computers Math. Applic.*, (submitted).
2. M. Spivak, *A Comprehensive Introduction to Differential Geometry*, Five Volumes, Publish or Perish, Inc., Berkeley, CA, (1979).
3. R. Abraham, J.E. Marsden and T. Ratiu, *Manifolds, Tensor Analysis, and Applications*, Second edition, Springer-Verlag, New York, (1988).
4. W.C. Rheinboldt, On the computation of multi-dimensional solution manifolds of parametrized equations, *Num. Math.* **53**, 165–181, (1988).
5. W.C. Rheinboldt, *Numerical Analysis of Parametrized Nonlinear Equations*, J. Wiley and Sons, New York, (1985).
6. W.C. Rheinboldt, Differential-geometric computational methods for nonlinear problems, In *World Congress of Nonlinear Analysts 1992*, (Edited by V. Lakshmikantham), pp. 1017–1037, W. de Gruyter, Berlin, Germany, (1996).
7. M.W. Berry, M.T. Heath, I. Kaneko, M. Law, R.J. Plemmons and R.C. Ward, An algorithm to compute a sparse basis of the null space, *Numer. Math.* **47**, 483–504, (1985).
8. T.F. Coleman and A. Pothen, *The Null-space Problem II: Algorithms*, Techn. Rep. TR 86-747, Cornell University, Dept. of Computer Science, Ithaca, New York, (April 1986).
9. G.H. Golub and C.F. Van Loan, *Matrix Computations*, Second edition, Johns Hopkins University Press, Baltimore, MD, (1989).
10. R.A. Wehage and E.J. Haug, Generalized coordinate partioning for dimension reduction in analysis of constrained dynamic systems, *J. of Mech. Design* **104**, 247–255, (1982).
11. F. Potra and W.C. Rheinboldt, Differential-geometric techniques for solving differential algebraic equations, In *Real-Time Integration Methods for Mechanical Systems Simulation*, (Edited by R. Deyo and E. Haug), pp. 155–192, Springer-Verlag, New York, (1991).
12. M.L. Brodzik, *Numerical Approximation of Manifolds and Applications*, Ph.D. Thesis, Applied Mathematics, University of Pittsburgh, Pittsburgh, PA, (1996).
13. M.L. Brodzik and W.C. Rheinboldt, On the computation of simplicial approximations of implicitly defined two-dimensional manifolds, *Computers Math. Applic.* **28** (9), 9–21, (1994).
14. T.F. Coleman and D.C. Sorensen, A note on the computation of an orthonormal basis for the nullspace of a matrix, *Math. Progr.* **29**, 234–242, (1984).
15. R.G. Parker and R.L. Rardin, *Discrete Optimization*, Acad. Press, Inc., Boston, MA, (1988).
16. R. Tomovich and M. Vukobratovic, *General Sensitivity Theory*, American Elsevier, New York, (1972).
17. W.C. Rheinboldt, On the sensitivity of parametrized equations, *SIAM J. Num. Anal.* **30**, 305–320, (1993).
18. W.C. Rheinboldt, Geometric notes on optimization with equality constraints, *Appl. Math. Lett.* **9** (3), 83–87, (1996).
19. J.J. More and D.C. Sorensen, Computing a trust region step, *SIAM J. Scient. Stat. Comp.* **4**, 553–560, (1983).
20. P.J. Rabier and W.C. Rheinboldt, On a computational method for the second fundamental tensor and its application to bifurcation problems, *Num. Math.* **57**, 681–694, (1990).
21. P.J. Rabier and W.C. Rheinboldt, On the numerical solution of the Euler Lagrange equations, *SIAM J. Num. Anal.* **32**, 318–329, (1995).
22. P.J. Rabier and W.C. Rheinboldt, A geometric treatment of implicit differential algebraic equations, *J. on Differential Equations* **109**, 110–146, (1994).

23. W.C. Rheinboldt, Differential-algebraic systems as differential—Equations on manifolds, *Math. of Comp.* **43**, 473–482, (1984).
24. W.C. Rheinboldt, On a constructive moving frame algorithm and the triangulation of equilibrium manifolds, In *Bifurcation: Analysis Algorithms Applications*, (Edited by T. Kuepper, R. Seydel and H. Troger), pp. 256–267, Birkhauser Verlag, Basel, Switzerland, (1987).