Approximate and Projected Natural Level Functions for Newton-type Iterations

Vom Promotionsausschuss der
Technischen Universität Hamburg-Harburg
zur Erlangung des akademischen Grades
Doktor der Naturwissenschaften (Dr. rer. nat.)

genehmigte Dissertation

von
Tim Steinhoff

aus der
Freien und Hansestadt Hamburg

2011

1. Gutachter: Prof. Dr. Wolfgang Mackens
2. Gutachter: Prof. Dr. Hubert Schwetlick

Tag der mündlichen Prüfung: 2. Februar 2011

Contents

List of Figures

List of Tables

List of Algorithms

List of Symbols

$\mathbb{R}$	set of real numbers
$\mathbb{R}^n$	(column) vector space of dimension n over $\mathbb{R}$ for positive integer n
$\mathbb{R}^{m\times n}$	matrix vector space of dimension $m \times n$ over $\mathbb{R}$ for positive integers m and n
$\dim(U)$	dimension of vector space U
$\operatorname{span}(u_1, \ldots, u_k)$	vector space spanned by the vectors $u_1, \ldots, u_k$
$u_{(i)}$	i-th component of vector u
a_{ij}	element of matrix A in the i-th row and j-th column
$\operatorname{img}(A)$	image of matrix A
$\ker(A)$	kernel of matrix A
$\operatorname{cond}_2(A)$	condition number of matrix A w.r.t. the Euclidean norm
$\det(A)$	determinant of matrix A
$\operatorname{rank}(A)$	rank of matrix A
A^{-1}	inverse of nonsingular matrix A
$A^\dagger$	Moore-Penrose pseudo-inverse of matrix A
A^-	(outer) generalized inverse of matrix A
A^T	transpose of matrix A
$\operatorname{diag}(u_1, \ldots, u_k)$	diagonal matrix with diagonal elements $d_{ii} = u_i$
I	identity matrix
p', p'', p'''	first, second, third derivative of p
$\operatorname{grad} f(x)$	gradient of f at x (a row vector)
$\mathcal{O}, o$	Landau symbols
$\delta x, \overline{\delta x}$	corrections in the context of approximate or quasi-Newton methods
$\Delta x, \overline{\Delta x}$	corrections in the context of Newton's method
$\angle(x, y)$	angle between the vectors x and y w.r.t. the dot product
$[\omega], [\Omega]$	computable estimates of nonlinearity bound or Lipschitz quantities
ω, Ω	nonlinearity bound or Lipschitz quantities
$\|\cdot\|$	arbitrary vector norm or corresponding matrix norm
$\|\cdot\|_2$	Euclidean vector norm or corresponding matrix norm
$\|\cdot\|_F$	Frobenius matrix norm

Chapter 1

Introduction

In this thesis we consider the problem to solve the nonlinear system of equations

$$F(x) = 0$$

via Newton type-iterations. We introduce a globalization approach to Newton's method via damping which is an enhancement of Deuflhard's natural level function concept, [10, 11]. The natural level function is defined at an iterate x_l by choosing $A = F'(x_l)^{-1}$ in

$$\tfrac{1}{2}\|AF(x)\|_2^2. \tag{1.1}$$

Step sizes are determined such that for the next iterate a decrease in the level function is achieved. The particular choice $A = F'(x_l)^{-1}$ from the set of nonsingular matrices A is motivated by the goal to avoid unnecessarily small step sizes. Such small step sizes are often observed in case that damping is controlled by the classical level function $\frac{1}{2}\|F(x)\|_2^2$.

A refinement of Deuflhard's analysis shows that the choice $A = F'(x_l)^{-1}$ can be ameliorated by introducing the projection onto the Newton correction Δx_l. We call the resulting level function

$$\tfrac{1}{2}\|P_{N_l}F'(x_l)^{-1}F(x)\|_2^2, \quad P_{N_l} := \frac{\Delta x_l \Delta x_l^T}{\Delta x_l^T \Delta x_l},$$

the *projected natural level function*. We consider the concept of the projected natural level function not only in the context of Newton's method. We also transfer it to a context where the Jacobian is not directly available but at least multiplications of the form $w^T \cdot F'(x)$ and $F'(x) \cdot d$ are computable. These products can be efficiently supplied by Automatic Differentiation techniques, [15]. By means of the resulting *approximate projected natural level function* we provide a damping strategy for approximate Newton methods. For an algorithmic realization we employ specific quasi-Newton rank-1 updates. Due to this choice we obtain in the context of quasi-Newton methods an alternative to Schlenkrich's globalization approach, [28]. His work is based on the classical level function $\frac{1}{2}\|F(x)\|_2^2$.

By means of specific affine invariant Lipschitz conditions on the Jacobian Deuflhard, [11], provides easy to handle polynomial models for the behavior of the natural level function in the direction of the Newton correction. If the respective Lipschitz constants are known step sizes are

determined according to these models. Proceeding in this way it is ensured that the next iterate is in the same path-connected component of the level set of the natural level function as the current iterate. In practice the Lipschitz constants are rarely known. They have to be estimated. For the computation of these estimates the affine invariance property of the Lipschitz conditions is exploited. We use similar techniques to determine step sizes in our algorithmic realizations of the approximate and projected natural level function concepts. It turns out that for the considered test problems our algorithms are as robust as a reference algorithm which is based on the natural level function. In general, the performance of our algorithm which is related to the projected natural level function is slightly better than the performance of the reference algorithm. The algorithm which is related to the approximate projected natural level function turns out to be superior in terms of run time for the considered problems of higher dimensions.

Inspired by the form of the estimates for the Lipschitz constants we introduce different measures to describe the nonlinearity of F. These measures, which we call *nonlinearity bounds*, are closer related to the estimates than the Lipschitz bounds are. By means of these bounds theory and praxis move closer together. We also make use of such bounds to prove a refined local convergence result in the context of Newton's method. Additionally, we provide a global convergence result for an approximate Newton's method where step sizes are controlled by a level function of the form (1.1) for a fixed nonsingular weight A.

The thesis is organized as follows.

In Chapter 2 we collect some well-known properties of Newton's method and the Newton correction to solve the problem $F(x) = 0$. Also, we provide a brief summary of properties of the natural level function and discuss the derivation of the polynomial model for the determination of step sizes.

In Chapter 3 we introduce the projected natural level function. We start with an in-depth-analysis of the influence of A on the set of step sizes which ensure descent for the respective level function. It turns out that for the natural level function a nonnegative 'disturbance'-term arises which narrows the above mentioned set. This term can be decreased in magnitude already by means of specific nonsingular weights A and vanishes entirely by introducing the projection onto the Newton correction, $A = P_{N_l} F'(x_l)^{-1}$.

In the course of our analysis of the projected natural level function we substitute Lipschitz conditions on the Jacobian by the above mentioned nonlinearity bounds and develop adaptations of the step size controls from [26, 11] and [5, 6]. Furthermore, we extend the concept of the projected natural level function to the context of least squares problems. Special emphasis is put on systems which emerge from a multiple shooting ansatz for boundary value problems and parameter estimation problems in ordinary differential equations.

In Chapter 4 we adapt the concept of the projected natural level function to a situation where only an approximation of the Jacobian (at least implicitly) is available. We provide an approximate Newton correction which is a direction of descent for the corresponding approximate projected natural level function. We monitor the quality of the level function and the correction by means

of the angle between the correction and the transposed negative gradient of the level function and the angle between the correction and the Newton correction. The first of these is directly available. For the other we present a reliable estimate.

For our algorithmic realization we assume that an initial approximation of the Jacobian is explicitly given. We compute further approximations by means of quasi-Newton rank-1 updates. For this purpose we combine different types of updates. If necessary the *purifying updates* improve the quality of the approximation such that the above stated angles stay bounded by some predefined threshold. Furthermore, a second update provides a direction of descent for the approximate projected natural level function. Thus, we call it *descent update*. We show local superlinear convergence of a sequence of iterates which emerges from an iteration where the corrections are computed by a recursive application of the descent update.

In Chapter 5 we provide a global convergence result for a damped Newton-like iteration where corrections are determined using an approximation to the Jacobian. The step size control is based on a level function of type (1.1) where the nonsingular weight A is kept fixed during the whole iteration. To ensure that the approximate correction is a direction of descent for the considered level function we use techniques similar to the ones we already employed in the context of the approximate projected natural level function. Step sizes are determined via a polynomial model of the test function in the direction of the approximate Newton correction. This polynomial model provides a generalization of the affine covariant and affine contravariant models from [11]. Our convergence result requires sufficiently good approximations to the Jacobian. This is ensured by the application of generalized purifying updates.

In Chapter 6 we consider various numerical test problems to test out our algorithmic implementations of the concepts of the projected natural level function and the approximate projected natural level function. For comparison we also run the test problems for an implementation of the natural level function concept.

Additionally, we give an outlook on potential adaptations and enhancements of the concepts of the approximate and projected natural level function.

In the course of this work several Lipschitz quantities and nonlinearity bounds appear. Each of them is denoted by ω or Ω occasionally provided with accents and/or indices. The actual semantic of these characters should usually be clear from the context. In case of ambiguity, e.g., if we compare two quantities, we add the reference number as an index. For example, the Lipschitz constant ω from (2.5) would be cited as $\omega_{(2.5)}$.

It is always assumed that ω or Ω, respectively, is finite and of best possible choice.

Chapter 2

Newton's Method and the Natural Level Function

In order to solve the nonlinear system of equations

$$F(x) = 0$$

one may apply Newton's method

$$x_{l+1} = x_l + \Delta x_l, \quad \Delta x_l = -F'(x_l)^{-1} F(x_l). \tag{2.1}$$

In this chapter we motivate this choice by presenting amiable and well-known properties of Newton's method like local quadratic convergence. Also, we discuss advantageous properties of the Newton correction in the context of a globalization approach via damping,

$$x_{l+1} = x_l + \lambda_l \Delta x_l, \quad \lambda \in (0, 1], \tag{2.2}$$

like the close relationship of the correction to the Newton path.

Furthermore, we briefly discuss properties of the natural level function

$$\tfrac{1}{2} \| F'(x_l)^{-1} F(x) \|_2^2$$

and the basic approach from [11] how to determine step sizes by means of this level function. The approach will be adapted in the next chapter where we will introduce the *projected natural level function*. This level function is in close relationship to the natural level function. By providing properties of the natural level function we will be in a state to compare both functions.

2.1 Properties of the Newton Iteration and Correction

We consider the problem

$$F(x) = 0$$

for a general nonlinear function F which fulfills the following assumption.

Assumption 2.1 *$F : \mathcal{D} \to \mathbb{R}^n$ is continuously differentiable on $\mathcal{D} \subseteq \mathbb{R}^n$ with $\mathcal{D} \neq \emptyset$ open and convex.*

There exist several (semi-)local convergence results which guarantee under certain conditions on F and the initial guess x_0 quadratic convergence of the iterates provided by Newton's method—see e.g. the classical Newton-Kantorovich and Newton-Mysovskikh theorems in [27], their refinements in [12, 11] or the results from [16]. For reference purposes we state a respective result in Subsection 2.1.2. If a given initial guess is not contained in the local contraction domain of Newton's method it is a common globalization approach to employ a damped iteration like (2.2). In this section we collect several statements from the literature, [11, 26], to motivate this approach.

The iterates provided by an application of Newton's method feature particular invariance properties w.r.t. affine transformations in the range and domain space of F. We will give a short explanation of these properties as well.

2.1.1 Affine invariance properties

In the context of solving $F(x) = 0$ for general nonlinear F which fulfills Assumption 2.1 the Newton iterates have the following invariance properties:

2.1.1.1 Affine covariance

Let $A \in \mathbb{R}^{n\times n}$ be an arbitrary nonsingular matrix and consider the transformed system

$$G(x) := AF(x) = 0. \tag{2.3}$$

Certainly, $F(x_*) = 0 \Leftrightarrow G(x_*) = 0$. Applying Newton's method both to the original and the transformed system starting at the same initial guess $x_0 \in \mathcal{D}$ gives the same sequence of iterates $\{x_l\}$ since

$$G'(x_l)^{-1}G(x_l) = F'(x_l)^{-1}A^{-1}AF(x_l) = F'(x_l)^{-1}F(x_l) \quad \forall l.$$

According to [11] this property of the Newton iterates is called *affine covariance*. Furthermore, we call a problem related quantity *affine covariant* if it does not change under a transformation of the form (2.3). An analysis or an approach is said to be affine covariant if it solely deals with affine covariant quantities.

Obviously, affine covariance also holds for the iterates of the damped iteration if the step sizes λ_l are determined by an affine covariant approach.

Remark 2.2 For practical purposes *scaling invariance* of an algorithm is desirable. This means that a change of units, say, from *cm* to *m* or some other general componentwise scaling of variables should have no impact on the behavior of the algorithm. In an affine covariant setting such a scaling invariance is obtained if we consider relative quantities in the domain space instead of absolute ones, cf. [11, 26]. This may be achieved in the following way. Choose a $\hat{x} \in \mathcal{D}$ with no component equal to zero. Then, by transforming the original system via

$$G(y) := F(\mathrm{diag}(\hat{x}) \cdot y)$$

we obtain $\Delta y_l = \operatorname{diag}(\hat{x})^{-1}\Delta x_l$. Rescaling in the domain space implies rescaling of $\hat{x}$ but leaving Δy_l unchanged, hence scaling invariance is obtained. However, to prevent overflow for values close to zero absolute threshold values are necessary. This may destroy scaling invariance. We come back to scaling invariance in Subsection 3.4.3. □

2.1.1.2 Affine contravariance

A second invariance concept in the context of Newton's method applied to the problem $F(x) = 0$ for general nonlinear F is *affine contravariance*: If we apply the transformation $By = x$ to $F(x) = 0$, i.e.,

$$G(y) \coloneqq F(By) = 0 \tag{2.4}$$

and let $By_0 = x_0$, the Newton iterates are transformed in the same manner as the domain space since

$$BG'(y_l)^{-1}G(y_l) = BB^{-1}F'(x_l)^{-1}F(x_l).$$

Note that the classical level function $\frac{1}{2}\|F(x)\|_2^2$ is invariant under the above transformation. An analysis for a globalization approach of Newton's method via damping which takes affine contravariance into account is given in [11].

2.1.2 Local quadratic convergence

As an evidence for local quadratic convergence of Newton's method we present the affine covariant Newton-Mysovskikh theorem from [11], however, with the slight modification of substituting the general vector norm $\|\cdot\|$ by the Euclidean norm. In Subsection 3.2.2 we will provide a refinement of this theorem. The techniques of the proof are similar to these used in [11] to prove the below given statements. Therefore, we omit a proof at this point.

Theorem 2.3 *Let F fulfill Assumption 2.1 and suppose that $F'(x)$ is invertible for each $x \in \mathcal{D}$. Assume that the following affine covariant Lipschitz condition holds:*

$$\|F'(z)^{-1}\big(F'(y) - F'(x)\big)(y - x)\|_2 \leqslant \omega\|y - x\|_2^2 \tag{2.5}$$

for collinear $x, y, z \in \mathcal{D}$. For the initial guess x_0 assume that

$$h_0 \coloneqq \omega\|\Delta x_0\|_2 < 2$$

where Δx_0 is the Newton correction at x_0.
Furthermore, suppose that for the closed ball $\bar{B}(x_0, \rho)$ with $\rho = \dfrac{\|\Delta x_0\|_2}{1 - \frac{1}{2}h_0}$ it holds that $\bar{B}(x_0, \rho) \subset \mathcal{D}$.

Then the sequence $\{x_l\}$ of ordinary Newton iterates defined via (2.1) remains in $\bar{B}(x_0, \rho)$ and converges to a solution $x_* \in \bar{B}(x_0, \rho)$ of $F(x) = 0$. Moreover,

$$\|x_{l+1} - x_l\|_2 \leqslant \tfrac{1}{2}\omega\|x_l - x_{l-1}\|_2^2, \tag{2.6}$$

$$\|x_l - x_*\|_2 \leqslant \frac{\|x_l - x_{l+1}\|_2}{1 - \frac{1}{2}\omega\|x_l - x_{l+1}\|_2}. \tag{2.7}$$

Proof. [11]

■

Corollary 2.4 *Under the assumptions of the above theorem there exist a $\kappa > 0$ and an index $\underline{l}$ such that for the Newton iterates it holds that*

$$\|x_{l+1} - x_*\|_2 \leqslant \kappa \|x_l - x_*\|_2^2 \quad \forall l \geqslant \underline{l},$$

i.e., the convergence is q-quadratic.

Proof. Let $e_l := x_l - x_*$. Since $\lim_{l\to\infty} \|\Delta x_l\|_2 = 0$ there is an index l_1 such that

$$0 < (1 - \tfrac{1}{2}\omega \|\Delta x_l\|_2)^{-1} \leqslant 2 \quad \forall l \geqslant l_1.$$

Hence, by means of (2.6) and (2.7) and for $l \geqslant l_1$,

$$\omega \|\Delta x_l\|_2 \geqslant 2 \frac{\|\Delta x_{l+1}\|_2}{\|\Delta x_l\|_2} \geqslant \frac{\|e_{l+1}\|_2}{\|\Delta x_l\|_2} \geqslant \frac{\|e_{l+1}\|_2}{\|e_{l+1}\|_2 + \|e_l\|_2} = \frac{\frac{\|e_{l+1}\|_2}{\|e_l\|_2}}{1 + \frac{\|e_{l+1}\|_2}{\|e_l\|_2}}$$

implying

$$\lim_{l\to\infty} \frac{\|e_{l+1}\|_2}{\|e_l\|_2} = 0.$$

Also,

$$\left| 1 - \frac{\|\Delta x_l\|_2}{\|e_l\|_2} \right| \leqslant \frac{\|e_{l+1}\|_2}{\|e_l\|_2}.$$

This means that there is an index $\underline{l} \geqslant l_1$ such that

$$\frac{\|\Delta x_l\|_2}{\|e_l\|_2} \leqslant 2 \quad \forall l \geqslant \underline{l}.$$

So finally we obtain

$$\|e_{l+1}\| \leqslant 2\|\Delta x_{l+1}\|_2 \leqslant \omega \|\Delta x_l\|_2^2 = \omega \frac{\|\Delta x_l\|_2^2}{\|e_l\|_2^2} \|e_l\|_2^2 \leqslant 4\omega \|e_l\|_2^2 =: \kappa \|e_l\|_2^2.$$

■

Starting far away from a solution the above stated conditions for the quadratic convergence of the full step Newton's method may not be guaranteed to hold. We motivate the application of the damped iteration in this case by providing further advantageous properties of the Newton correction.

2.1.3 Affine covariant trust region approach

The Newton correction Δx_l can be interpreted as the solution of the substitute *linear* problem

$$L(\Delta x) := F(x_l) + F'(x_l)\Delta x = 0.$$

As Theorem 2.3 shows, close to a solution x_* of the nonlinear problem $F(x) = 0$ successively solving the linearized problem L eventually produces a sequence of iterates which converges to x_*. The trust region approach of Levenberg-Marquardt type considers a linear model of F at some iterate x_l even if the iterate is far away from the solution. The idea is to restrict the next correction such that the iterate x_{l+1} is located in a neighborhood of x_l where the linearization of F is supposed to

be trusted to sufficiently model the behavior of F. This neighborhood is called the trust region. To determine the next correction the constrained quadratic minimization problem

$$\|F(x_l) + F'(x_l)\Delta x\|_2 = \min \qquad \text{s.t.} \quad \|\Delta x\|_2 \leqslant \delta_l$$

is considered. How to solve such a problem in a robust way is thoroughly described in [22].

The affine covariant reformulation from [11] of the above problem reads as follows

$$\|F'(x_l)^{-1}\big(F(x_l) + F'(x_l)\Delta x\big)\|_2 = \min \qquad \text{s.t.} \quad \|\Delta x\|_2 \leqslant \delta_l. \tag{2.8}$$

This affine covariant problem has a unique solution which we denote by $\widehat{\Delta x}$. Consider vectors $u_i \in \mathbb{R}^n$, $i = 1, \ldots, n-1$, with $u_i^T u_j = \delta_{ij}$ and $u_i^T \Delta x_l = 0$, $i = 1, \ldots, n-1$. Then, for certain $\alpha \in \mathbb{R}$ and $\beta_i \in \mathbb{R}$, $i = 1, \ldots, n-1$, a decomposition of $\widehat{\Delta x}$ of the form

$$\widehat{\Delta x} = \alpha \Delta x_l + \sum_{i=1}^{n-1} \beta_i u_i$$

exists. Hence,

$$\|F'(x_l)^{-1}\big(F(x_l) + F'(x_l)\widehat{\Delta x}\big)\|_2^2 = \|\widehat{\Delta x} - \Delta x_l\|_2^2 = |\alpha - 1| \cdot \|\Delta x_l\|_2^2 + \|\sum_{i=1}^{n-1} \beta_i u_i\|_2^2.$$

Since $\widehat{\Delta x}$ is the solution of the constrained minimization problem it holds that

$$\beta_i = 0, \quad i = 1, \ldots, n-1, \quad \text{and} \quad \alpha = \begin{cases} 1 & \text{if } \delta_l > \|\Delta x_l\|_2, \\ \delta_l & \text{otherwise,} \end{cases}$$

i.e., $\widehat{\Delta x} = \alpha \Delta x_l$ leading to a damped Newton iteration where the damping factor characterizes the radius of the trust region.

2.1.4 Descent properties

Let $x_* \in \mathcal{D}$ be a solution of $F(x) = 0$. A desirable criterion for determining step sizes λ_l in a damped Newton iteration would be

$$\|x_{l+1} - x_*\| < C \cdot \|x_l - x_*\|, \quad 0 \leqslant C < 1. \tag{2.9}$$

Unfortunately, such a monitor or an approximation of it, respectively, may only be at hand if we are already close to the solution—see Subsection 2.2.4 and the discussion in Chapter 2 of [11].

A common approach is to substitute the requirement (2.9) by a monotonicity criterion of the form

$$T(x_l + \lambda_l \Delta x_l) < T(x_l)$$

where $T : \mathcal{D} \to \mathbb{R}_+$ is a given test function. Here we consider *general level functions* defined via

$$T(x|A) := \tfrac{1}{2}\|AF(x)\|_2^2, \quad A \in \mathbb{R}^{n \times n}. \tag{2.10}$$

Remark 2.5 To the best of our knowledge general level functions of the above type are defined in the literature, see e.g. [10, 11, 26], solely for *nonsingular* A. We extend the classical definition in view of the *projected natural level function* which will be introduced in Chapter 3. For a singular A the property

$$T(x|A) = 0 \quad \Rightarrow \quad x = x_*$$

cannot be guaranteed in general. This is *not* necessarily a drawback, however, as it is seen from the discussion in Subsection 3.2.5. □

Regarding *first order* information the level functions from (2.10) turn out to be equally suited for determining step sizes in a damped Newton iteration:

For $x \in \mathcal{D}$ let

$$M_x := \{W \in \mathbb{R}^{n\times n} \mid WF(x) \neq 0\}. \tag{2.11}$$

Given a current iterate $x_l \in \mathcal{D}$ with $F(x_l) \neq 0$ the corresponding Δx_l is a direction of descent for *any* general level function where $A \in M_{x_l}$ since

$$\frac{d}{d\lambda} T(x_l + \lambda \Delta x_l|A)_{|\lambda=0} = -2T(x_l|A) < 0. \tag{2.12}$$

Furthermore, for affine linear F we obtain

$$T(x_l + \lambda \Delta x_l|A) \,/\, T(x_l|A) = (1-\lambda)^2. \tag{2.13}$$

This means that for affine linear F there is descent all the way down the Newton direction till the solution is reached for $\lambda = 1$, regardless which $A \in M_{x_l}$ is considered.

2.1.5 Relation to the Newton path

In terms of *level sets*

$$G(x|A) := \{z \in \mathcal{D} \mid T(z|A) \leqslant T(x|A)\} \tag{2.14}$$

and the path $\hat{x}_l : [0,2] \to \mathbb{R}^n$, $\hat{x}_l(\lambda) := x_l + \lambda \Delta x_l$, the relation (2.13) implies for affine linear F that

$$\hat{x}_l(\lambda) \in \overline{G}(x_l) \quad \forall \lambda \in [0,2], \tag{2.15}$$

where

$$\overline{G}(x) := \bigcap_{A \in \mathbb{R}^{n\times n}} G(x|A). \tag{2.16}$$

By definition, $\overline{G}(x)$ is affine covariant. Usually, (2.15) is not true for nonlinear F. However, under certain conditions on F and by means of $\overline{G}(x)$ there can be defined a path $\overline{x}_l$ as a generalization of $\hat{x}_l$ with the amiable property $T(\overline{x}_l(\lambda)|A) = (1-\lambda)^2 T(x_l|A)$ for all $A \in \mathbb{R}^{n\times n}$, hence fulfilling $\overline{x}_l(\lambda) \in \overline{G}(x_l)\ \forall \lambda \in [0,2]$. This is revealed by Theorem 2.7 below.

Remark 2.6 In [10, 11] $\overline{G}(x)$ is defined via

$$\overline{G}(x) := \bigcap_{\substack{A \in \mathbb{R}^{n\times n} \\ \text{nonsingular}}} G(x|A).$$

Since for all singular matrices $\tilde{A}$ there is a sequence of nonsingular matrices A_i such that $\lim_{i\to\infty}\{A_i\} = \tilde{A}$ and since $T(x|A)$ is continuous in A it is readily seen that

$$\bigcap_{\substack{A\in\mathbb{R}^{n\times n}\\ \text{nonsingular}}} G(x|A) = \bigcap_{A\in\mathbb{R}^{n\times n}} G(x|A).$$

We formally extend the definition of $\overline{G}(x)$ due to the fact that we consider general level functions of the type (2.10) where singular matrices are not excluded, cf. Remark 2.5. □

From [11] we obtain

Theorem 2.7 *Let F fulfill Assumption 2.1 and let $F'(x)$ be nonsingular for all $x \in \mathcal{D}$. For some nonsingular $\widehat{A} \in \mathbb{R}^{n\times n}$ and $x_0 \in \mathcal{D}$, let the path-connected component of $G(x_0|\widehat{A})$ in x_0 be compact and contained in $\mathcal{D}$. Then the path-connected component in x_0 of $\overline{G}(x_0)$ as defined in* (2.16) *is a topological path $\overline{x} : [0,2] \to \mathbb{R}^n$, the so-called Newton path, which satisfies*

$$\begin{aligned} F\big(\overline{x}(\lambda)\big) &= (1-\lambda)F(x_0),\\ T\big(\overline{x}(\lambda)|A\big) &= (1-\lambda)^2 T(x_0|A), \end{aligned} \tag{2.17}$$

$$\frac{d\overline{x}}{d\lambda} = -F'(\overline{x})^{-1}F(x_0), \qquad \begin{aligned} &\overline{x}(0) = x_0,\\ &\overline{x}(1) = x_* \text{ with } F(x_*) = 0, \end{aligned} \tag{2.18}$$

$$\frac{d\overline{x}}{d\lambda}_{|\lambda=0} = -F'(x_0)^{-1}F(x_0) \equiv \Delta x_0, \tag{2.19}$$

where Δx_0 is the ordinary Newton correction.

Proof. [11]

■

As it is seen from the above theorem, one step of the damped Newton iteration provides an approximation of first order to the Newton path, the 'path of virtue', which leads the way from x_0 to a solution x_*. We quote [11]:

> Even 'far away' from the solution point x_*, the Newton direction is an outstanding direction, only its length may be 'too large' for highly nonlinear problems.

2.2 The Natural Level Function

If one decides to exert a damped Newton iteration, i.e.,

$$x_{l+1} = x_l + \lambda_l \Delta x_l, \quad \Delta x_l = -F'(x_l)^{-1}F(x_l), \quad \lambda \in (0,1], \tag{2.20}$$

it still remains to determine the step sizes λ_l. As it is seen from the previous section all level functions $T(x|A)$ with $A \in M_{x_l}$ are basically suitable for this purpose.

In this section we will discuss the choice $A = F'(x_l)^{-1}$ at x_l, the *natural level function*, which we abbreviate *NLF*. We will provide a brief summary of statements from the literature, [10, 26, 11, 6], about properties of the NLF and the basic idea how to determine step sizes by means of the NLF.

Remark 2.8 The concept of the natural level function is *not* accompanied by a global convergence result since cycles in the iterates may occur—see the example in [2]. In [6] a specific step size restriction is presented such that 2-cycles can be excluded. However, m-cycles for $m > 2$ still may occur, [11]. In [6] it is also shown that under certain conditions global convergence can be guaranteed if an additional intermediate iteration is executed, i.e.,

$$x_{l+1,0} = x_l + \lambda_l \Delta x_l \tag{2.21a}$$

is followed by

$$\begin{aligned} x_{l+1,i+1} &= x_{l+1,i} + \tilde{\Delta} x_{l,i} \\ \tilde{\Delta} x_{l,i} &= -F'(x_l)^{-1}\big(F(x_{l+1,i}) - (1-\lambda_l)F(x_l)\big), \quad i = 0, \dots . \end{aligned} \tag{2.21b}$$

The additional steps simply perform a back projection onto the Newton path. However, the numerical tests from [6] does not indicate that this extended scheme is to be preferred to the basic one without back projection. In fact, the numerical results from [10] and [26] show that algorithms based on the concept of the natural level function without back projection perform very well in practice which gives indeed a justification for applying this concept. □

2.2.1 Polynomial model to determine step sizes

Ideally, the step size λ_l in the damped iteration (2.20) should ensure a decrease in the NLF for all $\lambda \in (0, \lambda_l]$, i.e.,

$$T\big(x_l + \lambda \Delta x_l | F'(x_l)^{-1}\big) < T\big(x_l | F'(x_l)^{-1}\big).$$

Additionally, it should fulfill

$$T\big(x_l + \lambda_l \Delta x_l | F'(x_l)^{-1}\big) \leqslant T\big(x_l + \lambda \Delta x_l | F'(x_l)^{-1}\big)$$

for all $\lambda \in (0, 1]$ with $x_l + \lambda \Delta x_l \in \mathcal{D}$. However, in general these demands are not satisfiable in an efficient way. Thus, an easy to handle approximation is required. The idea is to consider a polynomial model $p_l(\lambda)$ of the change of the NLF, i.e.,

$$\frac{T\big(x_l + \lambda \Delta x_l | F'(x_l)^{-1}\big)}{T\big(x_l | F'(x_l)^{-1}\big)} \leqslant p_l(\lambda)$$

and to determine a step size by means of this model: Let $\lambda \in (0, 1]$ such that $x_l + \lambda \Delta x_l \in \mathcal{D}$. With

$$\chi_l(\lambda) := F'(x_l)^{-1}\big(F(x_l + \lambda \Delta x_l) - F(x_l) - \lambda F'(x_l)\Delta x_l\big) \tag{2.22}$$

we obtain the identity

$$F'(x_l)^{-1} F(x_l + \lambda \Delta x_l) = (1-\lambda) F'(x_l)^{-1} F(x_l) + \chi_l(\lambda).$$

Assume that the affine covariant Lipschitz condition

$$\|F'(x)^{-1}\big(F'(y) - F'(x)\big)(y-x)\|_2 \leqslant \omega \|y-x\|_2^2 \quad \forall x, y \in \mathcal{D} \tag{2.23}$$

holds. Since

$$F(x_l + \lambda\Delta x_l) = F(x_l) + \int_0^\lambda F'(x_l + s\Delta x_l)\Delta x_l ds$$

a short calculation shows that

$$\|\chi(\lambda)\|_2 \leqslant \tfrac{1}{2}\omega\|\Delta x_l\|_2^2\lambda^2. \tag{2.24}$$

Thus,

$$\|F'(x_l)^{-1}F(x_l + \lambda\Delta x_l)\|_2 \leqslant (1 - \lambda + \tfrac{1}{2}\omega\|\Delta x_l\|_2\lambda^2) \cdot \|F'(x_l)^{-1}F(x_l)\|_2$$

and

$$\frac{T\big(x_l + \lambda\Delta x_l|F'(x_l)^{-1}\big)}{T\big(x_l|F'(x_l)^{-1}\big)} \leqslant p_l(\lambda)$$

with

$$p_l(\lambda) := (1 - \lambda + \tfrac{1}{2}\omega\|\Delta x_l\|_2\lambda^2)^2. \tag{2.25}$$

This polynomial is strictly convex on $[0, 1]$ and has a unique minimizer $\overline{\lambda}_l$ in $[0, 1]$ given via

$$\overline{\lambda}_l = \min\left(1, \frac{1}{\omega\|\Delta x_l\|_2}\right). \tag{2.26}$$

So in terms of this polynomial model the optimal choice for λ_l is

$$\lambda_l = \overline{\lambda}_l.$$

The step size $\overline{\lambda}_l$ depends on the Lipschitz constant from (2.23) which is in general computational not available, however, efficiently computable estimates exists, see [26, 11] for details.

Remark 2.9 Assume that the step size strategy $\lambda_l = \overline{\lambda}_l$ is applied for the iteration (2.20) and that the sequence of iterates $\{x_l\}$ is well defined. If it holds for an index $\underline{l}$ that

$$\omega_{(2.23)}\|\Delta x_{\underline{l}}\|_2 \leqslant 1 \qquad \text{and} \qquad \omega_{(2.5)}\|\Delta x_{\underline{l}}\|_2 < 2$$

then the first inequality ensures that $\lambda_{\underline{l}} = 1$, whereas by Theorem 2.3 the second inequality implies that $\lambda_l = 1$, $l > \underline{l}$. Hence, under the stated conditions eventually quadratic convergence of the iterates to a solution x_* of $F(x) = 0$ is obtained by this step size strategy.

□

The step size strategy $\lambda_l = \overline{\lambda}_l$ can also be interpreted in terms of the Newton path: Let $\overline{x}_l$ be the Newton path at x_l, i.e., $\overline{x}_l(0) = x_l$. For sufficiently smooth F it follows from Lemma 7 in [6] that

$$\overline{x}_l(\lambda) - x_l = \lambda\Delta x_l - \chi_l(\lambda) + O(\lambda^3). \tag{2.27}$$

Since (2.24) holds we obtain for $\lambda \in (0, \lambda_l]$ and by means of the triangular inequality the relations

$$1 - \frac{1}{2} \leqslant \frac{\|\overline{x}_l(\lambda) - x_l\|_2}{\lambda\|\Delta x_l\|_2} + \mathcal{O}(\lambda^2) \qquad \text{and} \qquad \frac{\|\overline{x}_l(\lambda) - x_l\|_2}{\lambda\|\Delta x_l\|_2} \leqslant 1 + \frac{1}{2} + \mathcal{O}(\lambda^2).$$

Neglecting the term of second order this means that up to λ_l the change of the Newton path is essentially represented by $\lambda\Delta x_l$. So it is very likely that the next iterate $x_{l+1} = x_l + \lambda_l\Delta x_l$ does not stray too far from the Newton path $\overline{x}_l(\lambda)$.

2.2.2 Relation to Steepest Descent method

The gradient of the general level function $T(x|A)$ is given via

$$\operatorname{grad} T(x|A) = \big(AF(x)\big)^T AF'(x). \tag{2.28}$$

At x_l and for $A = F'(x_l)^{-1}$ we obtain

$$-\operatorname{grad} T(x_l|F'(x_l)^{-1}) = -\big(F'(x_l)^{-1}F(x_l)\big)^T = \Delta x_l^T.$$

Hence, for the natural level function the Newton correction equals the (transposed) negative gradient. So determining step sizes by means of the natural level function this procedure may be interpreted as a modified Steepest Descent method, utilizing a sequence of level functions and its associated gradients—see [10].

2.2.3 Local dewarping

Considering a local model of the level function $T(x|A)$ at $x_l \in \mathcal{D}$, i.e.,

$$T_l^L(x|A) := \tfrac{1}{2}\|AF(x_l) + AF'(x_l)(x - x_l)\|_2^2.$$

The corresponding level sets

$$C_l(A) := \{x \in \mathbb{R}^n \mid T_l^L(x|A) = T_l^L(x_l|A)\} \tag{2.29}$$

are ellipsoids where the length of the half-axis are given as the inverses of the square roots of the eigenvalues of $\big(AF'(x_l)\big)^T AF'(x_l)$. It may be the case that the Newton correction is nearly orthogonal to the corresponding transposed gradient leading to small step sizes. For $A = F'(x_l)^{-1}$ the local model turns out to be

$$T_l^L\big(x|F'(x_l)^{-1}\big) = \tfrac{1}{2}\|x - (x_l + \Delta x_l)\|_2^2,$$

i.e., $C_l(F'(x_l)^{-1})$ describes a *sphere* where the Newton correction equals the transposed negative gradient pointing to the midpoint of the sphere.

2.2.4 Asymptotic error measurement

Let F be twice continuously differentiable. Then, by means of a Taylor expansion at a solution $x_* \in \mathcal{D}$ we obtain for $A \in \mathbb{R}^{n\times n}$

$$\begin{aligned} T(x|A) = {} & \tfrac{1}{2}\|x - x_*\|_2^2 \\ & - (x - x_*)^T\big(I - AF'(x_*)\big)(x - x_*) + \tfrac{1}{2}\|\big(I - AF'(x_*)\big)(x - x_*)\|_2^2 \\ & + o(\|x - x_*\|_2^2). \end{aligned} \tag{2.30}$$

Hence, for a converging sequence of iterates $\{x_l\}$ to x_* we have

$$T(x_{l+1}|F'(x_l)^{-1}) = \tfrac{1}{2}\|x_{l+1} - x_*\|_2^2 + o(\|x_{l+1} - x_*\|_2^2). \tag{2.31}$$

Furthermore, if $T(x_{l+1}|F'(x_l)^{-1}) < T(x_l|F'(x_l)^{-1})$ and $\|x_{l+1} - x_*\|_2 \in o(\|x_l - x_*\|_2)$ then

$$\|x_{l+1} - x_*\|_2 \leqslant \|x_l - x_*\|_2 + o(\|x_l - x_*\|_2).$$

Thus, asymptotically a reduction in the NLF implies a reduction in the error. Therefore, it is justifiable to call a globalization approach based on the NLF error-oriented.

Note that the relation $\|x_{l+1} - x_*\|_2 \in o(\|x_l - x_*\|_2)$ is true for $l \geqslant \underline{l}$ with some $\underline{l} \in \mathbb{N}$ if the damped iteration turns into an ordinary full step iteration and if the conditions of Theorem 2.3 are fulfilled—see Corollary 2.4.

2.2.5 Comparison to other choices of A in $T(x|A)$

The above described relation to Steepest Descent and the dewarping-property of the NLF suggest that the range of *valid step sizes* is increased compared to other choices of A. By the term *valid step size* we refer to some $\lambda \in (0, 1]$ such that with the Newton correction Δx_l at x_l and for given A it holds that

$$x_l + \lambda \Delta x_l \in \mathcal{D}$$

and (2.32)

$$T(x_l + s\Delta x_l|A) < T(x_l|A) \quad \forall s \in (0, \lambda].$$

From [11] we obtain the following result.

Theorem 2.10 *Let F fulfill Assumption 2.1 and let $F'(x)$ be nonsingular for all $x \in \mathcal{D}$. For a given current iterate $x_l \in \mathcal{D}$ with $F(x_l) \neq 0$ and for the level set $G(x_l|A)$ defined according to* (2.14) *let the closure of the path-connected component of $G(x_l|A)$ in x_l for some nonsingular $A \in \mathbb{R}^{n\times n}$ be a subset of $\mathcal{D}$. Assume that the affine covariant Lipschitz condition* (2.23) *holds. Let Δx_l be the Newton correction at x_l and define*

$$h_l := \omega \|\Delta x_l\|_2, \quad \overline{h}_l := h_l \operatorname{cond}_2\big(AF'(x_l)\big).$$

Then, one obtains for $\lambda \in [0, \min(1, 2/\overline{h}_l)]$:

$$\|AF(x_l + \lambda \Delta x_l)\|_2 \leqslant p_l(\lambda|A)\|AF(x_l)\|_2 \tag{2.33}$$

where

$$p_l(\lambda|A) := 1 - \lambda + \tfrac{1}{2}\overline{h}_l\lambda^2.$$

The optimal choice of damping factor in terms of this local estimate is

$$\overline{\lambda}_l(A) := \min(1, 1/\overline{h}_l).$$

Proof. [11] ∎

This result is a generalization of the model considerations from Subsection 2.2.1 since for $p_l(\lambda)$ from (2.25) and $\overline{\lambda_l}$ from (2.26) we obtain

$$p_l(\lambda) = p_l^2\big(\lambda|F'(x_l)^{-1}\big) \qquad \text{and} \qquad \overline{\lambda}_l = \overline{\lambda}_l\big(F'(x_l)^{-1}\big).$$

Remark 2.11 In [11] the above statement is given in terms of an arbitrary vector norm $\|\cdot\|$. We restrict ourselves to the Euclidean norm since we investigate general level functions of the form (2.10). □

By means of the above result we have

$$\lambda \in \big(0, \min(1, 2/\overline{h}_l)\big) \quad \Rightarrow \quad p_l(\lambda|A) < 1$$

which in turn implies descent, i.e.,

$$\frac{T(x_l + \lambda \Delta x_l|A)}{T(x_l|A)} < 1.$$

Since $\overline{h}_l$ is smallest for the choice $A = F'(x_l)^{-1}$ the range of λ such that $p_l(\lambda|A) < 1$ is maximized for the NLF. However, for a given A the polynomial model $p_l(\lambda|A)$ may vastly overestimate the relative change of $T(x|A)$ and therefore may only guarantee descent for a small subset of the set of all valid step sizes. Hence, it is not excluded that there are choices for A such that for the associated level function the set of all valid step sizes is a superset to the set of all valid step sizes w.r.t. the NLF. We will see in the next chapter that such choices indeed exist.

Chapter 3

The Projected Natural Level Function

In this chapter we will introduce the *projected natural level function* (PNLF). This level function emerges from an in-depth-analysis of the influence of A in $T(x|A)$ on the range of step sizes which provide descent for the respective level function. This analysis will be given in Section 3.1. We will show that compared to the natural level function (NLF) the PNLF provides descent for a wider range of step sizes. The PNLF is given via

$$\tfrac{1}{2}\|P_{N_l}F'(x_l)^{-1}F(x)\|_2, \quad P_{N_l} = \frac{\Delta x_l \Delta x_l^T}{\Delta x_l^T \Delta x_l},$$

where Δx_l is the Newton correction at x_l. In contrast to other level functions the PNLF is based on a weight A that is singular. As we will see from the discussions in Section 3.2 this does not turn out to be a drawback.

In Section 3.3 we will show that the idea of a projected natural level function can be transported to the context of least squares problems as well. We will provide an analysis of the generalized PNLF and define refinements of this generalization in case that the least squares problem is related to a multiple shooting approach to solve boundary value problems or parameter estimation problems in ordinary differential equations.

The main objective of this chapter is to provide an affine covariant globalization approach of Newton's method via damping where the step sizes are determined by means of the PNLF. Therefore, we will adapt and extend in Section 3.4 existing step size strategies from [26, 11, 5, 6] to fit into the context of the PNLF. In the course of this a new method to provide a predictor step size will be derived. This predictor directly exploits the underlying concept of a projected level function.

Apart from one minor exception, see Paragraph 3.4.1.3 for details, for the complete analysis in this chapter it is not mandatory to rely on Lipschitz conditions on the Jacobian to describe the nonlinearity of F. Instead we will use conditions of the form

$$2\|F'(x)^{-1}\big(F(y) - F(x) - F'(x)(y-x)\big)\|_2 \leqslant \omega\|y-x\|_2^2 \quad \forall x, y \in \tilde{\mathcal{D}} \subseteq \mathcal{D}$$

and, taking the projection into account, of the form

$$2\|P_N(x)F'(x)^{-1}\big(F(y)-F(x)-F'(x)(y-x)\big)\|_2 \leqslant \omega\|y-x\|_2^2 \quad \forall x,y\in\tilde{\mathcal{D}}\subseteq\mathcal{D}.$$

Here $P_N(x)$ is the orthogonal projector onto the Newton correction at x or the identity matrix if $F(x)=0$. We call these affine covariant conditions *nonlinearity bounds*. As we will see these bounds are more suitable to describe the nonlinearity of F than accordingly defined affine covariant Lipschitz conditions on the Jacobian. Also we will show that they are in closer relationship to the numerical available estimates of the nonlinearity of F which are employed in the step sizes controls from Section 3.4.

3.1 In-depth-analysis of the Influence of A in $T(x|A)$

The polynomial

$$p_l(\lambda|A) = 1-\lambda+\tfrac{1}{2}\overline{h}_l\lambda^2$$

from Theorem 2.10 nicely reflects that one has to take information beyond first order into account to justify a particular choice of A. This is a direct consequence of the descent properties (2.12) and (2.13).

To provide an in-depth-analysis of the influence of A, and in particular of $A=F'(x_l)^{-1}$, our first goal is to find a quantity which provides information about the nonlinearity of F independently of A. For the upcoming analysis we will drop the iteration index and assume that $x\in\mathcal{D}$, $F(x)\neq 0$ and

$$\lambda\in\Lambda := \{\lambda\in(0,1] \mid x+\lambda\Delta x\in\mathcal{D}\} \tag{3.1}$$

holds. Note that since $\mathcal{D}$ is convex it holds that $\lambda\in\Lambda \Rightarrow s\in\Lambda$ for all s with $0<s<\lambda$.

First, suppose F to be affine linear. Then, the Jacobian is constant, i.e., $F'(x)\equiv J\in\mathbb{R}^{n\times n}$ and it holds that

$$AF(x+\lambda\Delta x)-AF(x) = AJ\cdot\lambda\Delta x.$$

For nonlinear F we have to introduce a correction to the above stated equation. We choose this quantity to be related to the *domain space* of F. As it turns out such a quantity is given by $\chi(\lambda)$ from (2.22). We obtain

$$AF(x+\lambda\Delta x)-AF(x) = AF'(x)\cdot\big(\lambda\Delta x+\chi(\lambda)\big)$$

with (3.2)

$$\chi(\lambda) = F'(x)^{-1}\big(F(x+\lambda\Delta x)-F(x)-\lambda F'(x)\Delta x\big).$$

It is readily seen that $\chi(\lambda)$ is the unique quantity that fulfills the above identity for *any* $A\in\mathbb{R}^{n\times n}$. First, we will use this identity to develop a refinement of the result from Theorem 2.10. For this, we write

$$AF(x+\lambda\Delta x) = (1-\lambda)AF(x)+AF'(x)\chi(\lambda).$$

We introduce the affine covariant *nonlinearity bound*

$$2\|F'(x)^{-1}\big(F(y)-F(x)-F'(x)(y-x)\big)\|_2 \leqslant \omega\|y-x\|_2^2 \quad \forall x,y\in\mathcal{D}. \tag{3.3}$$

Then, applying the Euclidean norm, utilizing the triangular inequality and submultiplicativity of the Euclidean norm we obtain

$$\begin{aligned}\|AF(x+\lambda\Delta x)\|_2 &\leqslant (1-\lambda)\|AF\|_2 + \|AF'(x)\|_2\chi(\lambda)\|_2\\ &\leqslant (1-\lambda)\|AF\|_2 + \tfrac{1}{2}\|AF'(x)\|_2\omega\|\Delta x\|_2^2\lambda^2\\ &\leqslant (1-\lambda+\tfrac{1}{2}\operatorname{cond}_2\big(AF'(x)\big)\omega\|\Delta x\|_2\lambda^2)\|AF\|_2\end{aligned} \tag{3.4}$$

for nonsingular A. Certainly, this result also advocates the choice $A = F'(x)^{-1}$. However, the range of valid step sizes guaranteed by this model is probably increased compared to the one provided by Theorem 2.10 since $\omega_{(3.3)} \leqslant \omega_{(2.23)}$ as the following proposition shows.

Proposition 3.1 *Let F fulfill Assumption 2.1 and assume that the Lipschitz condition* (2.23) *holds. Then, the nonlinearity bound from* (3.3) *is well defined and one has*

$$\omega_{(3.3)} \leqslant \omega_{(2.23)}.$$

Proof. Let x, $y \in \mathcal{D}$. Since the Lipschitz condition (2.23) is assumed to hold, $F'(x)$ is nonsingular. Then,

$$\begin{aligned}&2\|F'(x)^{-1}\big(F(y)-F(x)-F'(x)(y-x)\big)\|_2\\ &\quad= 2\left\|\int_0^1 F'(x)^{-1}\Big(F\big(x+s(y-x)\big)-F(x)-F'(x)\Big)(y-x)ds\right\|_2\\ &\quad\leqslant 2\int_0^1 \left\|F'(x)^{-1}\Big(F\big(x+s(y-x)\big)-F(x)-F'(x)\Big)(y-x)\right\|_2 ds\\ &\quad\leqslant 2\int_0^1 s\omega_{(2.23)}\|y-x\|_2^2 ds = \omega_{(2.23)}\|y-x\|_2^2.\end{aligned} \tag{3.5}$$

By definition $\omega_{(3.3)}$ is of best possible choice, i.e.,

$$\omega_{(3.3)} = 2\cdot\sup_{x,y\in\mathcal{D},\, x\neq y}\|F'(x)^{-1}\big(F(y)-F(x)-F'(x)(y-x)\big)\|_2 \,/\, \|y-x\|_2^2.$$

Hence, because of (3.5) any statement of the form $\omega_{(3.3)} > \omega_{(2.23)}$ would lead to a contradiction. Therefore, $\omega_{(3.3)} \leqslant \omega_{(2.23)}$.

■

As a direct consequence we obtain

Corollary 3.2 *Under the assumptions of Proposition 3.1 and for $x_l \in \mathcal{D}$ with $F(x_l) \neq 0$ let the optimal damping factors*

$$\overline{\lambda}_{l,(2.23)} := \overline{\lambda}_l(F'(x_l)^{-1}) = \min\big(1, 1/\omega_{(2.23)}\|\Delta x_l\|_2\big) \qquad \textit{and} \qquad \overline{\lambda}_{l,(3.3)} := \min\big(1, 1/\omega_{(3.3)}\|\Delta x_l\|_2\big)$$

be given. Then,

$$\overline{\lambda}_{l,(3.3)} \geqslant \overline{\lambda}_{l,(2.23)}.$$

Remark 3.3 A nonlinearity bound of the form

$$\|F(y)-F(x)-F'(x)(y-x)\| \leqslant \gamma\|y-x\|^2 \quad \forall x, y \in \mathcal{D}$$

is already mentioned in [27] as a substitute for classical Lipschitz conditions in the context of a local convergence analysis of Newton's method. However, such a bound lacks affine covariance and is therefore not suited to serve as a substitute for the affine covariant Lipschitz condition (2.23). This is corrected by our choice (3.3). □

For our further analysis we reconsider (3.2). In addition to the drop of the iteration index, we introduce the abbreviations $F := F(x)$ and $J := F'(x)$. First, we will provide a result regarding the relative change of $T(\cdot|A)$ along the Newton direction without introducing estimates.

Proposition 3.4 *With the orthogonal projection onto the Newton direction*

$$P_N := \frac{\Delta x \Delta x^T}{\Delta x^T \Delta x} \tag{3.6}$$

let the quantities $\mu(\lambda)$ and $\chi_\perp(\lambda)$ be defined via the decomposition

$$\begin{aligned} \chi(\lambda) &= P_N \chi(\lambda) + (I - P_N)\,\chi(\lambda) \\ &=: \mu(\lambda) J^{-1} F + \chi_\perp(\lambda). \end{aligned} \tag{3.7}$$

Then, for $A \in M_x$, with M_x defined in (2.11), the ratio

$$T(x + \lambda \Delta x|A)\,/\,T(x|A) \tag{3.8}$$

can be written as

$$\begin{aligned} \frac{T(x+\lambda\Delta x|A)}{T(x|A)} &= \big(1 - \lambda + \mu(\lambda)\big)^2 + \frac{\|AJ\chi_\perp(\lambda)\|_2^2}{\|AF\|_2^2} \\ &\quad + 2\big(1 - \lambda + \mu(\lambda)\big)\frac{(AF)^T}{\|AF\|_2} \cdot \frac{AJ\chi_\perp(\lambda)}{\|AF\|_2} \quad \forall \lambda \in \Lambda \end{aligned} \tag{3.9}$$

where Λ is given as in (3.1).

Proof. By the definitions of Δx, $\mu(\lambda)$ and $\chi_\perp(\lambda)$ we obtain

$$\begin{aligned} AF(x + \lambda\Delta x) &= AF + AJ(\lambda\Delta x) + AJ\chi(\lambda) \\ &= (1-\lambda)AF + AJ\chi(\lambda) \\ &= \big(1 - \lambda + \mu(\lambda)\big)AF + AJ\chi_\perp(\lambda). \end{aligned}$$

Hence, for the numerator of (3.8) we have

$$\begin{aligned} T(x + \lambda\Delta x|A) &= \tfrac{1}{2}\|\big(1 - \lambda + \mu(\lambda)\big)AF + AJ\chi_\perp(\lambda)\|_2^2 \\ &= \tfrac{1}{2}\big(1 - \lambda + \mu(\lambda)\big)^2\|AF\|_2^2 \\ &\quad + \big(1 - \lambda + \mu(\lambda)\big)(AF)^T AJ\chi_\perp(\lambda) \\ &\quad + \tfrac{1}{2}\|AJ\chi_\perp(\lambda)\|_2^2. \end{aligned}$$

Dividing by $T(x|A) = \frac{1}{2}\|AF(x)\|_2^2$ and rearranging yields (3.9).

■

We discover that there is some part, namely

$$\big(1 - \lambda + \mu(\lambda)\big)^2, \tag{3.10}$$

which cannot be touched by the scaling provided by A. We call this term *invariant core.* For the choice $A = J^{-1}$ we obtain by definition of $\chi_\perp(\lambda)$,

$$\big(AF\big)^T AJ\chi_\perp(\lambda) = \big(J^{-1}F\big)^T\chi_\perp(\lambda) = 0 \quad \forall\lambda \in \Lambda,$$

which implies that (3.9) simplifies to

$$\frac{T(x+\lambda\Delta x|J^{-1})}{T(x|J^{-1})} = \big(1-\lambda+\mu(\lambda)\big)^2 + \frac{\|\chi_\perp(\lambda)\|_2^2}{\|J^{-1}F\|_2^2}. \tag{3.11}$$

Since

$$\|\chi_\perp(\lambda)\|_2^2 \,/\, \|J^{-1}F\|_2^2 \geqslant 0 \quad \forall\lambda \in \Lambda,$$

the range of valid step sizes may be increased if we can damp out or even completely get rid of this term such that solely the invariant core remains.

Remark 3.5 The invariant core strongly depends on the choice of the decomposition of $\chi(\lambda)$. That our choice (3.7) is indeed reasonable will be discussed in Subsection 3.2.6. □

Theorem 3.6 *Let F fulfill Assumption 2.1 and let $x \in \mathcal{D}$ such that $F(x) \neq 0$ and $J := F'(x)$ is nonsingular. With the Newton correction Δx at x consider for σ with $0 \leqslant \sigma \leqslant 1$*

$$A(\sigma) := U\Sigma(\sigma)U^T J^{-1} \tag{3.12}$$

where

$$\Sigma(\sigma) := \operatorname{diag}(1,\sigma,\ldots,\sigma) \in \mathbb{R}^{n\times n}$$

and

$$U := \Big(\Delta x/\|\Delta x\|_2 \quad \tilde{U}\Big),\ \tilde{U} \in \mathbb{R}^{n\times n-1} \quad \textit{such that} \quad U^TU = I.$$

Then, $A(\sigma) \in M_x$ with M_x from (2.11) *and for $A = A(\sigma)$ the relation* (3.9) *reads as follows*

$$\frac{T\big(x+\lambda\Delta x|A(\sigma)\big)}{T\big(x|A(\sigma)\big)} = \big(1-\lambda+\mu(\lambda)\big)^2 + \sigma^2\cdot\frac{\|\chi_\perp(\lambda)\|_2^2}{\|J^{-1}F(x)\|_2^2} \quad \forall\lambda\in\Lambda. \tag{3.13}$$

Furthermore,

$$T\big(x|A(\sigma)\big) = T(x|J^{-1}) \tag{3.14a}$$

and

$$T\big(x+\lambda\Delta x|A(\sigma)\big) \leqslant T(x+\lambda\Delta x|J^{-1}) \quad \forall\lambda\in\Lambda. \tag{3.14b}$$

Proof. We abbreviate $F = F(x)$. It holds that $A(\sigma) \in M_x$ because

$$A(\sigma)F = U\Sigma(\sigma)U^TJ^{-1}F = U(-\|\Delta x\|_2,0,\ldots,0)^TF = J^{-1}F \neq 0 \tag{3.15}$$

by the assumption that $F \neq 0$. This result also implies that (3.14a) holds. Furthermore,

$$\begin{aligned}\big(A(\sigma)F\big)^TA(\sigma)J &= (J^{-1}F)^TU\Sigma(\sigma)U^T\\ &= (-\|\Delta x\|_2,0,\ldots,0)U^T = (J^{-1}F)^T.\end{aligned} \tag{3.16}$$

Therefore, by the definition of $\chi_\perp(\lambda)$ in (3.7) we have

$$\big(A(\sigma)F\big)^TA(\sigma)J\chi_\perp(\lambda) = (J^{-1}F)^T\chi_\perp = 0 \qquad \forall\lambda\in\Lambda.$$

Also,

$$A(\sigma)\chi_\perp = U\Sigma(\sigma)U^T\chi_\perp = \sigma \cdot \begin{pmatrix} 0 & \tilde{U} \end{pmatrix} \begin{pmatrix} 0 \\ \tilde{U}^T\chi_\perp \end{pmatrix} = \sigma \cdot \chi_\perp$$

since $\chi_\perp \in \mathrm{img}(\tilde{U})$. Hence, for such an $A(\sigma)$ the relation (3.9) becomes

$$\frac{T\big(x + \lambda\Delta x | A(\sigma)\big)}{T\big(x|A(\sigma)\big)} = \big(1 - \lambda + \mu(\lambda)\big)^2 + \sigma^2 \cdot \frac{\|\chi_\perp(\lambda)\|_2^2}{\|J^{-1}F\|_2^2}$$

which is just (3.13). The inequality (3.14b) is a direct consequence of $0 \leqslant \sigma \leqslant 1$, (3.11) and (3.13). ■

Even if only nonsingular matrices are considered, as it is done in [10, 11, 26], this result shows that there are matrices such that larger step sizes are possible compared to the choice J^{-1}. This improvement is maximal for the choice $\sigma = 0$, which, however, is related to a *singular* matrix, since

$$A(0) = P_N$$

by the definition of $A(\sigma)$ in (3.12) and of P_N in (3.6).

Definition 3.7 (Projected natural level function) *Suppose F fulfills Assumption 2.1. Let $x_l \in \mathcal{D}$, $F(x_l) \neq 0$ and $F'(x_l)$ be nonsingular. Then for*

$$P_{N_l} := \frac{\Delta x_l \Delta x_l^T}{\Delta x_l^T \Delta x_l}, \quad \Delta x_l := -F'(x_l)^{-1}F(x_l),$$

being the orthogonal projection onto the Newton correction at x_l we call

$$T(x|P_{N_l}F'(x_l)^{-1}) = \tfrac{1}{2}\|P_{N_l}F'(x_l)^{-1}F(x)\|_2^2$$

the projected natural level function *(at x_l) or in short* PNLF.

3.2 Basic Relations

In this section we will collect several basic aspects related to the PNLF. We will briefly discuss the advantageous to take affine covariance into account for a globalization approach based on the PNLF. Also, we will provide a first basic scheme how to determine step sizes in a damped Newton iteration which is monitored by the PNLF. The step size strategies in Section 3.4 are refinements of this scheme. Generally, we will follow the idea from Deuflhard, [10, 11], and use a polynomial model of the behavior of the PNLF to determine step sizes. However, in contrast to Deuflhard's approach, our model will be given in terms of an affine covariant nonlinearity bound instead of an affine covariant Lipschitz condition on the Jacobian. Additionally, we will make use of the concept of nonlinearity bounds to give a refinement of the affine covariant Newton-Mysovskikh Theorem 2.3.

In Section 2.2 we stated some of the advantageous properties of the NLF like the relation to Steepest Descent and asymptotic error measurement. In this section we will investigate what changes occur regarding these properties if instead level functions of the type $T\big(x|A(\sigma)\big)$ with $A(\sigma)$ from (3.12) are considered. Special emphasis will be put on the case $\sigma = 0$, i.e., on the PNLF.

Furthermore, we will provide an example which illustrates the potential of the PNLF-concept.

3.2.1 The role of affine covariance

For $A(\sigma)$ as defined in (3.12) and by Theorem 3.6 the relation

$$\frac{T\big(x+\lambda\Delta x|A(\sigma)\big)}{T\big(x|A(\sigma)\big)} = \big(1-\lambda+\mu(\lambda)\big)^2 + \sigma^2\cdot\frac{\|\chi_\perp(\lambda)\|_2^2}{\|J^{-1}F(x)\|_2^2} \quad \forall\lambda\in\Lambda$$

holds. Assume that the Newton path $\overline{x}$ at x, i.e., $\overline{x}(0)=x$ is well defined. If we consider $\overline{x}(\lambda)$ instead of the correction step $x+\lambda\Delta x$ we obtain from Theorem 2.7 the result

$$\frac{T\big(\overline{x}(\lambda)|A(\sigma)\big)}{T\big(x|A(\sigma)\big)} = (1-\lambda)^2.$$

Hence, $\mu(\lambda)$ and $\beta(\lambda) := \sigma^2\cdot\frac{\|\chi_\perp(\lambda)\|_2^2}{\|J^{-1}F(x)\|_2^2}$ characterize the deviation $\overline{x}(\lambda)-(x+\lambda\Delta x)$. By definition $\mu(\lambda)$ and $\beta(\lambda)$ are affine covariant. This is a property which also holds for the deviation itself! Therefore, we think it is reasonable to preserve this property in an analysis of the level function $T\big(x|A(\sigma)\big)$. In case of the PNLF, i.e. $\sigma=0$, this means to provide affine covariant bounds for $\mu(\lambda)$. Taking affine covariance into account also provides the advantage that reasonable and cheaply computable numerical estimates of these bounds are available, for details refer to the discussions of step size controls in Section 3.4.

3.2.2 Polynomial model to determine step sizes

From Theorem 3.6 it follows that for the PNLF solely the invariant core remains to describe the relative change of the level function, i.e., (re-)introducing indices, we have

$$T(x_l+\lambda\Delta x_l|P_{N_l}F'(x_l)^{-1}) = \big(1-\lambda+\mu_l(\lambda)\big)^2 T(x_l|P_{N_l}F'(x_l)^{-1}). \tag{3.17}$$

Note that for arbitrary $z\in\mathbb{R}^n$ it holds that $\|P_{N_l}z\|_2 = |\Delta x_l^T z|/\|\Delta x_l\|_2$. So in terms of the affine covariant *projected* nonlinearity bound

$$2\|P_N(x)F'(x)^{-1}\big(F(y)-F(x)-F'(x)(y-x)\big)\|_2 \leqslant \omega\|y-x\|_2^2 \quad \forall x,y\in\mathcal{D} \tag{3.18}$$

with $P_N(x)$ being the orthogonal projection onto the Newton direction at x or the identity matrix in case $F(x)=0$ we may estimate for $\Lambda_l\ni\lambda>0$,

$$\begin{aligned}\mu_l(\lambda) &= -\frac{\Delta x_l^T}{\|\Delta x_l\|_2^2}F'(x_l)^{-1}\big(F(x_l+\lambda\Delta x_l)-F(x_l)-\lambda F'(x_l)\Delta x_l\big)\\ &\leqslant |\mu_l(\lambda)| = \frac{\|P_{N_l}F'(x_l)^{-1}\big(F(x_l+\lambda\Delta x_l)-F(x_l)-\lambda F'(x_l)\Delta x_l\big)\|_2}{\lambda^2\|\Delta x_l\|_2^2}\|\Delta x_l\|_2\lambda^2 \qquad (3.19)\\ &\leqslant \tfrac{1}{2}\omega\|\Delta x_l\|_2\lambda^2\end{aligned}$$

which leads to

Theorem 3.8 *Let F fulfill Assumption 2.1 and let $F'(x)$ be nonsingular for all $x\in\mathcal{D}$. For a given current iterate $x_l\in\mathcal{D}$ with $F(x_l)\neq 0$ let P_{N_l} be the orthogonal projector onto the Newton correction Δx_l. Furthermore, for the level set $G\big(x_l|P_{N_l}F'(x_l)\big)$ defined according to* (2.14) *let the closure of the path-connected component of $G\big(x_l|P_{N_l}F'(x_l)\big)$ in x_l be a subset of $\mathcal{D}$. Assume that the affine covariant projected nonlinearity bound* (3.18) *holds. Then,*

$$T(x_l+\lambda\Delta x_l|P_{N_l}F'(x_l)^{-1}) \leqslant \big(1-\lambda+\tfrac{1}{2}\omega\|\Delta x_l\|_2\lambda^2\big)^2 T(x_l|P_{N_l}F'(x_l)^{-1}) \tag{3.20}$$

for all $\lambda \in \Lambda_l$ *with* Λ_l *according to* (3.1). *Also,* $[0, \min(1, 2\overline{\lambda}_l)] \subseteq \Lambda_l$ *where*

$$\overline{\lambda}_l := \min\left(1, \frac{1}{\omega \|\Delta x_l\|_2}\right)$$

with ω *from* (3.18) *is the unique minimizer in* $[0,1]$ *of the above polynomial estimate.*

Proof. The estimate (3.20) is a direct consequence of (3.17)-(3.19). The polynomial

$$p(\lambda) := \left(1 - \lambda + \tfrac{1}{2}\omega \|\Delta x_l\|_2 \lambda^2\right)^2$$

of (3.20) is strictly convex on $[0,1]$ and it holds that $p'(0) = -2$. If $\overline{\lambda}_l = 1/\omega\|\Delta x_l\|_2$ a short calculation shows that $p'(\overline{\lambda}_l) = 0$. If $\overline{\lambda}_l = 1$ then $p(s) > p(\overline{\lambda}_l)$ for all $s \in [0,1)$ by the strict convexity of p and $p'(0) = -2$. So in either case $\overline{\lambda}_l$ is the unique minimizer of p in $[0,1]$. It holds that $p(\lambda) \leqslant 1$ for all $\lambda \in [0, \min(1, 2\overline{\lambda}_l)]$. Hence, any statement of the form $\hat{\lambda} \in [0, \min(1, 2\overline{\lambda}_l)]$ but $\hat{\lambda} \notin \Lambda_l$ would either contradict the assumption about the closure of the path-connected component of $G\big(x_l | P_{N_l} F'(x_l)\big)$ in x_l or the estimate (3.20). ■

Remark 3.9 Similar to the strategy which we discussed in Subsection 2.2.1 for the NLF the step size strategy $\lambda_l = \overline{\lambda}_l$ with $\overline{\lambda}_l$ as defined in the above theorem can also be interpreted in terms of the Newton path. As before let $\overline{x}_l$ be the Newton path at x_l, i.e., $\overline{x}_l(0) = x_l$. Multiplying the relation (2.27) from the left by P_{N_l} yields

$$P_{N_l} \cdot \big(\overline{x}_l(\lambda) - x_l\big) = \lambda \Delta x_l - P_{N_l} \chi_l(\lambda) + O(\lambda^3).$$

By means of the definition of $\chi_l(\lambda)$ in (2.22) and by means of the nonlinearity bound (3.18) we obtain

$$\|P_{N_l} \chi_l(\lambda)\|_2 \leqslant \tfrac{1}{2}\omega \|\Delta x_l\|_2^2 \lambda^2$$

and therefore

$$1 - \frac{1}{2} \leqslant \frac{\|P_{N_l} \cdot \big(\overline{x}_l(\lambda) - x_l\big)\|_2}{\lambda \|\Delta x_l\|_2} + \mathcal{O}(\lambda^2) \quad \text{and} \quad \frac{\|P_{N_l} \cdot \big(\overline{x}_l(\lambda) - x_l\big)\|_2}{\lambda \|\Delta x_l\|_2} \leqslant 1 + \frac{1}{2} + \mathcal{O}(\lambda^2) \quad \forall \lambda \in (0, \lambda_l].$$

Let us neglect the higher order term $\mathcal{O}(\lambda^2)$. Then, in contrast to the NLF, up to λ_l it is the change of the Newton path *in the direction of the Newton correction* which is essentially represented by $\lambda \Delta x_l$. □

For $P_N(x)$ from (3.18) we have $\|P_N(x) z\|_2 \leqslant \|z\|_2 \; \forall z \in \mathbb{R}^n$ which directly leads to an extension of Proposition 3.1 and Corollary 3.2.

Corollary 3.10 *Under the assumptions of Proposition 3.1 and Corollary 3.2 it holds that*

$$\omega_{(3.18)} \leqslant \omega_{(3.3)} \leqslant \omega_{(2.23)}.$$

Hence,

$$\overline{\lambda}_{l,(3.18)} \geqslant \overline{\lambda}_{l,(3.3)} \geqslant \overline{\lambda}_{l,(2.23)}$$

where $\overline{\lambda}_{l,(3.18)} = \overline{\lambda}_l$ *from Theorem 3.8 and* $\overline{\lambda}_{l,(3.3)}$, $\overline{\lambda}_{l,(2.23)}$ *are defined as in Corollary 3.2.*

If in a damped iteration (2.20) the step size λ_l is chosen via $\lambda_l = \overline{\lambda}_{l,(3.18)}$ and the iterates converge to a solution x_* of $F(x) = 0$ with nonsingular Jacobian $F'(x_*)$ we may argue in the same manner as in Remark 2.9 to finally ensure quadratic convergence. However, Theorem 2.3 is not stated in terms of nonlinearity bounds and no projection is considered. These concepts are introduced by the following theorem yielding a refinement of Theorem 2.3.

Theorem 3.11 *Let F fulfill Assumption 2.1 and suppose that $F'(x)$ is invertible for each $x \in \mathcal{D}$. Assume that the following affine covariant projected nonlinearity bound holds:*

$$\|P_N(z)F'(z)^{-1}\big(F(y) - F(x) - F'(x)(y-x)\big)\|_2 \leqslant \omega\|y-x\|_2^2 \tag{3.21}$$

for collinear $x, y, z \in \mathcal{D}$ and

$$P_N(z) := \begin{cases} \frac{\Delta z \Delta z^T}{\Delta z^T \Delta z} & \text{with } \Delta z = -F'(z)^{-1}F(z) \ \text{ if } \ F(z) \neq 0 \\ I & \text{otherwise.} \end{cases}$$

For the initial guess $x_0 \in \mathcal{D}$ assume that

$$h_0 := \omega\|\Delta x_0\|_2 < 1, \quad \Delta x_0 := -F'(x_0)^{-1}F(x_0).$$

Furthermore, suppose that for the closed ball $\bar{B}(x_0, \rho)$ with $\rho = \dfrac{\|\Delta x_0\|_2}{1-h_0}$ it holds that $\bar{B}(x_0, \rho) \subset \mathcal{D}$.

Then the sequence $\{x_l\}$ of ordinary Newton iterates defined via (2.1) remains in $\bar{B}(x_0, \rho)$ and converges to a solution $x_* \in \bar{B}(x_0, \rho)$ of $F(x) = 0$. Moreover,

$$\|x_{l+1} - x_l\|_2 \leqslant \omega\|x_l - x_{l-1}\|_2^2, \tag{3.22}$$

$$\|x_l - x_*\|_2 \leqslant \frac{\|x_l - x_{l+1}\|_2}{1 - \omega\|x_l - x_{l+1}\|_2}. \tag{3.23}$$

Proof. The basic scheme of the proof is adapted from the proof of Theorem 2.2 in [11].

From the definition of ρ it follows that $x_1 \in \bar{B}(x_0, \rho)$. Assume that $x_l \in \bar{B}(x_0, \rho)$ for $l \geqslant 1$. Due to the definition of the Newton iterates and P_N we have

$$\|\Delta x_l\|_2 = \|P_N(x_l)\Delta x_l\|_2 = \|P_N(x_l)F'(x_l)^{-1}\big(F(x_l) - F(x_{l-1}) - F'(x_{l-1})\Delta x_{l-1}\big)\|_2.$$

Applying the nonlinearity bound yields

$$\|\Delta x_l\|_2 \leqslant \omega\|\Delta x_{l-1}\|_2^2,$$

i.e., (3.22). With the notation $h_l := \omega\|\Delta x_l\|_2$ this inequality leads to

$$h_l \leqslant h_{l-1}^2. \tag{3.24}$$

From this and by the assumption $h_0 < 1$ a simple induction argument shows that

$$h_l < h_{l-1} < \cdots < h_0 < 1 \quad \text{and} \quad \|\Delta x_l\|_2 \leqslant h_k^{l-k}\|\Delta x_k\|_2 \quad \text{for } k \leqslant l.$$

Hence, repeated application of the triangular inequality yields

$$\|x_{l+1} - x_k\|_2 \leqslant \sum_{j=k}^{l}\|\Delta x_j\|_2 \leqslant \|\Delta x_k\|_2 \sum_{j=0}^{l-k} h_k^j = \|\Delta x_k\|_2 \frac{1 - h_k^{l-k+1}}{1-h_k} \leqslant \frac{\|\Delta x_k\|_2}{1-h_k}. \tag{3.25}$$

From the case $k = 0$ it is readily seen that $x_{l+1} \in \bar{B}(x_0, \rho)$. Therefore, all Newton iterates are well defined and remain in $\bar{B}(x_0, \rho)$. By means of $h_0 < 1$ and (3.24) contraction of the $\{h_l\}$ is obtained, i.e., $\lim_{l\to\infty} h_l = 0$ holds. Hence, by (3.25) $\{x_l\}$ is a Cauchy sequence converging to some $x_* \in \bar{B}(x_0, \rho)$. Since $F'(x)$ is continuous and $F'(x_*)$ is nonsingular it holds that $F(x_*) = 0$. The estimate (3.23) follows from (3.25) by considering the limit $l \to \infty$.

■

In analogy to Corollary 2.4 we may state

Corollary 3.12 *Under the assumptions of the above theorem there exist a $\kappa > 0$ and an index $\underline{l}$ such that for the Newton iterates it holds that*

$$\|x_{l+1} - x_*\|_2 \leqslant \kappa \|x_l - x_*\|_2^2 \quad \forall l \geqslant \underline{l},$$

i.e., the convergence is q-quadratic.

Proof. Follow the lines of the proof of Corollary 2.4.

■

Remark 3.13 A further refinement of Theorem 3.11 is obtained if we assume that the nonlinearity bound

$$\|F'(y)^{-1}\big(F(y) - F(x) - F'(x)(y - x)\big)\|_2 \leqslant \omega \|y - x\|_2^2, \tag{3.26}$$

for all x, $y \in \mathcal{D}$ with $y - x = -F'(x)^{-1}F(x)$ holds which is an adaption of the Lipschitz condition in [6]. Such a bound also fits into the context of *projected* nonlinearity bounds since

$$\begin{aligned} F'(y)^{-1}\big(F(y) - F(x) - F'(x)(y - x)\big) &= F'(y)^{-1}F(y) = P_N(y)F'(y)^{-1}F(y) \\ &= P_N(y)F'(y)^{-1}\big(F(y) - F(x) - F'(x)(y - x)\big) \end{aligned}$$

for $y - x = -F'(x)^{-1}F(x)$. For this specific way to describe the nonlinearity of F the concepts of projected and non-projected bounds coincide. □

Remark 3.14 It follows from an argument similar to the one we used to proof Proposition 3.1 that $2 \cdot \omega_{(3.21)} \leqslant \omega_{(2.5)}$. Hence, by the above theorem local convergence is guaranteed for a wider range of initial values x_0 compared to Theorem 2.3. □

In analogy to Remark 2.9 and by means of the above local convergence result we can ensure quadratic convergence of a damped iteration (2.20) where λ_l is chosen as $\overline{\lambda}_l$ from Theorem 3.8 if the sequence of iterates $\{x_l\}$ remains well defined and if there is an index $\underline{l}$ such that

$$\omega_{(3.18)}\|\Delta x_{\underline{l}}\|_2 \leqslant 1 \qquad \text{and} \qquad \omega_{(3.21)}\|\Delta x_{\underline{l}}\|_2 < 1.$$

Remark 3.15 The above conditions are true if the sequence of iterates reached already the local contraction domain from Theorem 3.11. We cannot state a global convergence result based on the PNLF. In analogy to the NLF the occurrence of cycles in the iterate cannot be excluded. However, for the NLF cycles have not been observed in practical applications, [6], and fortunately they also do not occur in our numerical tests for the PNLF. No step size restrictions for the PNLF are considered like they are developed in [6] to avoid 2-cycles in the NLF context. Also, we do not pursue a back projection strategy like (2.21). The numerical tests in Chapter 6 show that the PNLF-algorithm performs very well without such supporting techniques. □

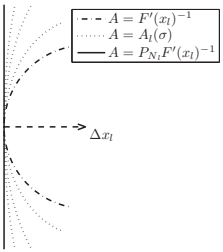


Figure 3.1: Cutout of level sets $C_l(A)$ for $A = F'(x_l)^{-1}$, $A = A_l(\sigma)$ and $A = A_l(0) = P_{N_l}F'(x_l)^{-1}$. In the limit $\sigma = 0$ the sphere becomes a plane.

3.2.3 Relation to Steepest Descent method

Recall from (2.28) that for the general level function defined in (2.10) it holds that

$$\operatorname{grad} T(x|A) = \big(AF(x)\big)^T AF'(x).$$

Introducing indices in (3.12) and by means of (3.15) and (3.16) we obtain

$$-\operatorname{grad} T\big(x_l|A_l(\sigma)\big) = -\big(F'(x_l)^{-1}F(x_l)\big)^T = \Delta x_l^T$$

which also holds for $\sigma = 0$, i.e., for the PNLF. So we do not lose the property to deal with a level function for which the Newton direction locally provides the steepest descent.

3.2.4 Local dewarping

As stated in Subsection 2.2.3 the level set $C_l(A)$ from (2.29) turns out to be a sphere for $A = F'(x_l)^{-1}$. Choosing $A = A_l(\sigma)$ and considering $\sigma \to 0$ the sphere stretches such that in the limit, i.e., for the PNLF it becomes a *plane*—see Figure 3.1. This limit may be considered as an optimum in local dewarping.

3.2.5 Asymptotic error measurement

Since

$$T\big(x_l|A_l(\sigma)\big) = T(x_l|F'(x_l)^{-1})$$

we have by (2.30)

$$T\big(x_l|A_l(\sigma)\big) = \tfrac{1}{2}\|x_l - x_*\|_2^2 + o(\|x_l - x_*\|_2^2), \quad 0 \leqslant \sigma < 1,$$

for twice continuously differentiable F. This means, at the current iterate asymptotically the error is reflected by the value of the level function, also for the singular choice $A_l(0)$, i.e., for the PNLF. However, such a relation cannot be guaranteed in general for the next iterate x_{l+1} like it is the

case for the NLF, cf. (2.31). But this is not a drawback concerning a termination criterion. We can and will make use of

$$F'(x_l)^{-1}F(x_{l+1})$$

to estimate the error. This quantity is zero if and only if x_{l+1} is a solution to $F(x) = 0$. Since

$$\tfrac{1}{2}\|F'(x_l)^{-1}F(x_{l+1})\|_2^2 = T(x_{l+1}|F'(x_l)^{-1})$$

this approach can be interpreted in a way that we use the PNLF to determine step sizes and the NLF to provide a termination criterion.

3.2.6 General decomposition of χ

As the previous section shows the relative change of the projected natural level function at $x_l \in \mathcal{D}$ with $F(x_l) \neq 0$ along the Newton direction, i.e.,

$$T\big(x_l + \lambda\Delta x_l|P_{N_l}F'(x_l)^{-1}\big) \,/\, T\big(x_l|P_{N_l}F'(x_l)^{-1}\big)$$

is described by the invariant core

$$\big(1 - \lambda + \mu_l(\lambda)\big)^2$$

which is based on an *orthogonal* decomposition of the nonlinearity quantity $\chi(\lambda)$. Here we consider a generalized decomposition and discuss the influence of it in terms of level functions whose relative change is given by the associated invariant core.

We drop the indices and set $J := F'(x)$ and $F := F(x)$. Assume that $\tilde{V} \in \mathbb{R}^{n\times n-1}$ is chosen such that

$$V = \begin{pmatrix} J^{-1}F & \tilde{V} \end{pmatrix}$$

is nonsingular. Then, with the decomposition

$$\begin{aligned}\chi(\lambda) &= V\begin{pmatrix} 1 & 0^T \\ 0 & O \end{pmatrix}V^{-1}\chi(\lambda) + V\begin{pmatrix} 0 & 0^T \\ 0 & I \end{pmatrix}V^{-1}\chi(\lambda) \\ &=: \tilde{\mu}(\lambda)J^{-1}F + \chi_{\tilde{V}}(\lambda)\end{aligned} \tag{3.27}$$

we obtain

$$AF(x + \lambda\Delta x) = \big(1 - \lambda + \tilde{\mu}(\lambda)\big)AF + AJ\chi_{\tilde{V}}(\lambda). \tag{3.28}$$

Since $\operatorname{img}(\tilde{V})$ is a hyperplane in $\mathbb{R}^n$ its orthogonal space is given by an one dimensional subspace of $\mathbb{R}^n$. Let w be a basis vector of this subspace. Because of

$$\chi_{\tilde{V}}(\lambda) \in \operatorname{img}(\tilde{V}) \quad \forall\lambda \in \Lambda$$

the choice $A = uw^TJ^{-1}$, $u \in \mathbb{R}^n \setminus \{0\}$, yields

$$\frac{T\big(x + \lambda\Delta x|uw^TJ^{-1}\big)}{T\big(x|uw^TJ^{-1}\big)} = \big(1 - \lambda + \tilde{\mu}(\lambda)\big)^2.$$

We reduce the above fraction by dropping common factors of the enumerator and denominator. For arbitrary $a, b \in \mathbb{R}^n$ it holds that $\|ab^T\|_2 = \|a\|_2\|b\|_2$. Hence, the above relation can be stated as

$$\frac{\big(w^T J^{-1} F(x + \lambda\Delta x)\big)^2}{\big(w^T J^{-1} F\big)^2} = \big(1 - \lambda + \tilde{\mu}(\lambda)\big)^2.$$

Consider the reduced level function at $x \in \mathcal{D}$

$$R_w : \begin{cases} \mathcal{D} \to \mathbb{R}_+ \\ z \mapsto \big(w^T J^{-1} F(z)\big)^2. \end{cases}$$

Its gradient w.r.t. z at x is given via

$$\operatorname{grad} R_w(x) = 2w^T J^{-1} F \cdot w^T.$$

Since both V and $\begin{pmatrix} w & \tilde{V} \end{pmatrix}$ are nonsingular it holds that $w^T \Delta x \neq 0$ which implies that $\operatorname{grad} R_w(x) \cdot \Delta x < 0$. Particulary,

$$\angle\big(\Delta x, -\operatorname{grad} R_w^T(x)\big) = 0 \quad \Leftrightarrow \quad w = \alpha \cdot J^{-1} F,\ \alpha \neq 0.$$

So in order to ensure a steepest descent property of the Newton correction w.r.t. R_w and at x we have to choose $w = \alpha \cdot J^{-1} F$. Let us opt for this choice. By definition w is a basis vector of $\operatorname{img}(\tilde{V})^{\perp}$. Hence, the decomposition (3.27) implies that

$$\chi_{\tilde{V}}(\lambda) = (I - P_N)\chi(\lambda)$$

and also

$$\tilde{\mu}(\lambda) = \mu(\lambda)$$

holds. This yields our orthogonal decomposition of $\chi(\lambda)$. Thus it is indeed reasonable to opt for this type of decomposition.

3.2.7 An illustrative example

In order to obtain an impression of the potential of the PNLF-concept compared to the NLF-concept we provide the following example. It is an adaption of the example which is presented in Chapter 3 of [5] and also in [6].

We consider for given $a \in \mathbb{R} \setminus \{0\}$ and $x = \big(x_{(1)}, x_{(2)}\big)^T \in \mathbb{R}^2$ the problem

$$F(x) = 0, \quad \text{with} \quad F(x) := \begin{pmatrix} x_{(1)} \\ a \cdot x_{(2)} + \frac{1}{4}\big(x_{(1)} - 50\big)^2 \end{pmatrix} \tag{3.29}$$

with the unique solution $x_* = (0, -625 \cdot a^{-1})^T$ and for the initial guess $x_0 = (50, 1)^T$. For all $x \in \mathbb{R}^2$ and $a \neq 0$ the Jacobian $J(x) := F'(x)$ is nonsingular and it is readily seen that the Newton iteration converges in at most two steps for an arbitrary initial guess. The example from [5, 6] is given via setting $a = 50$.

At x_0 we investigate the behavior of

$$Q_0(\lambda; A) := \frac{T(x_0 + \lambda\Delta x_0 | A)}{T(x_0|A)}, \quad \lambda \in [0, 1],$$

for $A \in \{J(x_0)^{-1}, P_{N_0}J(x_0)^{-1}\}$ and $a \in \{1/8, 1/5, 1, 50\}$ in terms of valid step sizes.

The Newton correction at x_0 is $\Delta x_0 = -x_0$. Also,

$$F(x + \lambda\Delta x) - F(x) - \lambda J(x)\Delta x = \frac{1}{2}\lambda^2 J'[\Delta x]^2, \quad J' \in \mathbb{R}^{2\times 2\times 2},$$

which leads to

$$\chi(\lambda) = \lambda^2 \begin{pmatrix} 0 \\ \frac{1}{4}a^{-1}(\Delta x_{(1)})^2 \end{pmatrix}, \quad \text{i.e., at } x_0\text{:} \quad \chi_0(\lambda) = \lambda^2 \begin{pmatrix} 0 \\ 625 \cdot a^{-1} \end{pmatrix}.$$

Some calculations show that

$$\mu_0(\lambda) = \underbrace{\frac{625}{2501}}_{\approx 1/4} a^{-1}\lambda^2, \qquad \frac{\|\chi_{0,\perp}(\lambda)\|_2^2}{\|\Delta x_0\|_2^2} = \underbrace{\left(\frac{31250}{2501}\right)^2}_{\approx 156} a^{-2}\lambda^4 =: \beta_0(\lambda).$$

Hence,

$$\begin{aligned} Q_0(\lambda; P_{N_0}J(x_0)^{-1}) &= \left(1 - \lambda + \tfrac{625}{2501}a^{-1}\lambda^2\right)^2, \\ Q_0(\lambda; J(x_0)^{-1}) &= Q_0(\lambda; P_{N_0}J(x_0)^{-1}) + \beta_0(\lambda). \end{aligned}$$

So in the case $a = 50$ the influence of the quantities μ_0 and β_0 is rather negligible leading to a valid full step for both level functions—see Figure 3.2(a). Setting $a = 1$ the quantity $\beta_0(\lambda)$ becomes of significant magnitude and as a result the range of valid step sizes for the NLF reduces drastically. On the contrary, for the PNLF the full step is still valid—see Figure 3.2(b). This changes with further decreasing the parameter a. Due to the influence of μ_0 one is no longer a valid step size. In fact, for any $\lambda \in (0, 1]$ there is an a such that λ is no longer valid. However, comparing the two level functions the range of valid step sizes for the PNLF is still considerably larger—see Figure 3.2(c) and 3.2(d). Note that for $x_0 = (50, 0)^T$ we have $\Delta x_0 = -x_0$ and $\Delta x_0 \perp \chi_0(\lambda)$. Hence, $\mu_0(\lambda) \equiv 0$ leading for any value of $a \neq 0$ to a valid step size of $\lambda = 1$ if the PNLF is considered.

3.3 Extension to Least Squares Problems

In this section we will extend the concept of the projected natural level function to the context of least squares problems. We will give basic ideas such that by means of the stated results an adaption of the upcoming step size strategies in Section 3.4 will be possible.

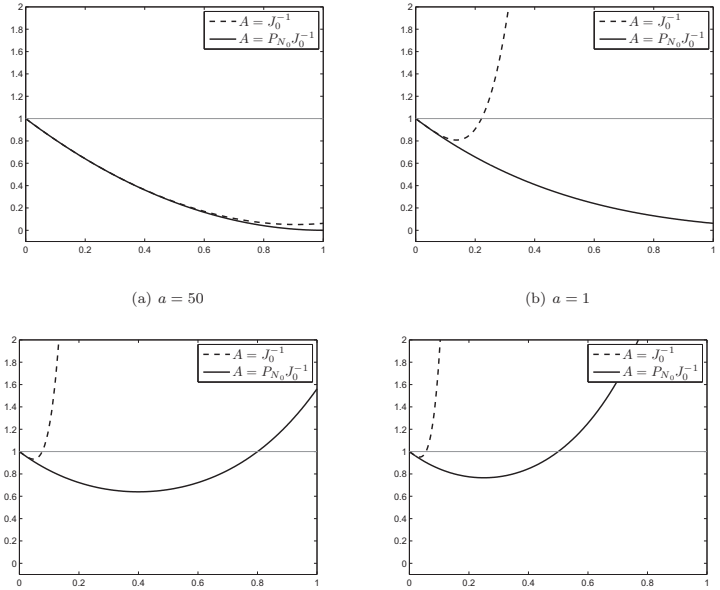
Let the nonlinear functions F_0 and F_c fulfill the following assumption.

Assumption 3.16 *$F_i : \mathcal{D} \to \mathbb{R}^{n_i}$, $i \in \{0, c\}$, are continuously differentiable on $\mathcal{D} \subseteq \mathbb{R}^n$ nonempty, open and convex and it holds that $n_0 \geqslant n \geqslant n_c$. Furthermore,*

$$\operatorname{rank}\big(F_c'(x)\big) = n_c \quad \forall x \in \mathcal{D}. \tag{3.30}$$

We consider the constrained least squares problem

$$\begin{aligned} &\|F_0(x)\|_2^2 = \min! \\ &\text{s.t. } F_c(x) = 0. \end{aligned} \tag{3.31}$$



(a) $a = 50$ (b) $a = 1$

(c) $a = 1/5$ (d) $a = 1/8$

Figure 3.2: Comparison of $Q_0(\lambda; A)$ for $A \in \{J_0^{-1}, P_{N_0} J_0^{-1}\}$ and various a

To solve such a problem we consider a damped Gauß Newton iteration

$$x_{l+1} = x_l + \lambda_l \Delta x_l, \quad \Delta x_l = -J(x_l)^- F(x_l), \quad \lambda_l \in (0, 1],$$

with the definitions

$$F(x) := \begin{pmatrix} F_0(x) \\ F_c(x) \end{pmatrix}, \quad J(x) := F'(x) = \begin{pmatrix} F_0'(x) \\ F_c'(x) \end{pmatrix} \tag{3.32}$$

and $J(x_l)^- \in \mathbb{R}^{n \times (n_0 + n_c)}$ being some generalized inverse of $J(x_l)$ such that Δx_l is a solution of the linearization

$$\begin{aligned} &\|F_0(x_l) + F_0'(x_l)\Delta x\|_2^2 = \min! \\ &\text{s.t. } F_c(x_l) + F_c'(x_l)\Delta x = 0. \end{aligned} \tag{3.33}$$

We will first discuss the general problem (3.31). Afterwards we will take care of problems with special structured F_c' which arise in the context of solving boundary value problems and parameter estimation problems via a multiple shooting approach.

For local convergence results of the full step Gauß Newton iteration refer, e.g., to [12] and [5]. These results are special since they take the concept of *S-invariance* as an extension of affine

covariance into account. For a problem of the form (3.31) there is always a set S of matrices

$$S := \left\{ A \in \mathbb{R}^{n_0'+n_c' \times n_0+n_c} \;\middle|\; A = \begin{pmatrix} A_0 \\ A_c \end{pmatrix},\ A_0 \in \mathbb{R}^{n_0' \times n_0+n_c},\ A_c \in \mathbb{R}^{n_c' \times n_0+n_c},\ n_0' \geqslant n_0,\ n_c' \geqslant n_c \right\}$$

such that (3.31) and

$$\begin{aligned} \|A_0 F(x)\|_2^2 &= \min! \\ \text{s.t. } A_c F(x) &= 0 \end{aligned} \tag{3.34}$$

have the same solutions. To verify this let us consider the Example 3.1.41 from [5]: Choose

$$A_0 := \begin{pmatrix} U & C_1 \\ 0 & C_2 \end{pmatrix} \qquad \text{and} \qquad A_c := \begin{pmatrix} 0 & C_3 \end{pmatrix}$$

with orthogonal $U \in \mathbb{R}^{n_0 \times n_0}$, arbitrary $C_1 \in \mathbb{R}^{n_0 \times n_c}$, $C_2 \in \mathbb{R}^{n_0'-n_0 \times n_c}$ and nonsingular $C_3 \in \mathbb{R}^{n_c \times n_c}$. Then (3.31) and (3.34) have the same solutions.

Let $A \in S$. Under certain regularity conditions on $J(x)$, [5], it holds that

$$\big(AJ(x)\big)^- A = J(x)^- \tag{3.35}$$

which means that the iterates Δx are invariant under the transformation provided by A. In [5] this invariance property is also respected in a globalization approach by employing a generalization of the natural level function, i.e.

$$T\big(x|J(x_l)^-\big) := \tfrac{1}{2}\|J(x_l)^- F(x)\|_2^2, \tag{3.36}$$

at x_l to determine a step size λ_l. In analogy to the Newton case the above natural level function is a particular instance of the general level function

$$T(x|A) := \tfrac{1}{2}\|AF(x)\|_2^2, \quad A \in \mathbb{R}^{n \times n_0+n_c}.$$

Though (3.30) is assumed to be true, it may be that at some $x \in \mathcal{D}$ the matrix $J(x)$ is not of full rank n, i.e., there is some nonnegative q such that

$$\operatorname{rank}\big(J(x)\big) = n_c + q < n.$$

We will briefly review the construction of a generalized inverse $J^-(x_l)$ such that $\Delta x_l = -J^-(x_l)F(x_l)$ is the unique solution to (3.33) of smallest Euclidean norm. We will apply techniques from [31]. The idea is to obtain an unrestricted least squares problem by means of some suitable transformations of (3.33) and to solve the resulting problem by an application of the respective Moore-Penrose pseudo-inverse. In the following we will drop the dependence on x_l.

Let $Q \in \mathbb{R}^{n \times n}$ be orthogonal and $P_1 \in \mathbb{R}^{n_c \times n_c}$ a permutation matrix such that

$$P_1 F_c' Q = \begin{pmatrix} R_c & 0 \end{pmatrix}$$

with an upper triangular matrix $R_c \in \mathbb{R}^{n_c \times n_c}$. This matrix is nonsingular due to (3.30). Define

$$\begin{pmatrix} \xi \\ \zeta \end{pmatrix} := Q^T \Delta x, \quad \xi \in \mathbb{R}^{n_c},\ \zeta \in \mathbb{R}^{n_d} \quad \text{where} \quad n_d := n - n_c,$$

and

$$\tilde{F}_c := P_1 F_c \quad \text{as well as} \quad \begin{pmatrix} A & B \end{pmatrix} := F_0' Q, \quad A \in \mathbb{R}^{n_0 \times n_c},\ B \in \mathbb{R}^{n_0 \times n_d}.$$

Then, (3.33) becomes

$$\|F_0 + A\xi + B\zeta\|_2^2 = \min_{\xi,\zeta}!$$
$$\text{s.t. } \tilde{F}_c + R_c \xi = 0.$$

The linearized constraints uniquely determine ξ to be

$$\xi = -R_c^{-1} \tilde{F}_c.$$

We substitute this result into the above problem. Additionally, let $P_2 \in \mathbb{R}^{n_d \times n_d}$ be a permutation matrix such that we obtain a QR-decomposition $Q_0 R_0$ of BP_2 where the diagonal entries $r_{ii}^{(0)}$, $i = 1, \ldots, n_d$, of R_0 are ordered such that $|r_{11}^{(0)}| \geqslant |r_{22}^{(0)}| \geqslant \cdots \geqslant |r_{n_d n_d}^{(0)}|$. Define

$$\tilde{F}_0 := Q_0^T F_0, \quad \tilde{A} := Q_0^T A, \quad \tilde{\zeta} := P_2^T \zeta.$$

Hence, the unconstrained least squares problem

$$\|\tilde{F}_0 - \tilde{A} R_c^{-1} \tilde{F}_c + R_0 \tilde{\zeta}\|_2^2 = \min_{\tilde{\zeta}}!$$

is obtained. Rank deficiency of J is directly reflected by the rank of R_0. As it is seen from (3.39) below it holds that $\operatorname{rank}(R_0) + n_c = \operatorname{rank}(J)$. The pseudo-inverse solution of the unconstrained problem is

$$\tilde{\zeta} = -R_0^{\dagger} (\tilde{F}_0 - \tilde{A} R_c^{-1} \tilde{F}_c).$$

Putting things together, $(\xi^T, \tilde{\zeta}^T)^T$ is determined via

$$\begin{pmatrix} \xi \\ \tilde{\zeta} \end{pmatrix} = - \begin{pmatrix} 0 & R_c^{-1} \\ R_0^{\dagger} & -R_0^{\dagger} \tilde{A} R_c^{-1} \end{pmatrix} \begin{pmatrix} \tilde{F}_0 \\ \tilde{F}_c \end{pmatrix} =: -\tilde{J}^{-} \begin{pmatrix} \tilde{F}_0 \\ \tilde{F}_c \end{pmatrix} \tag{3.37}$$

which is the unique solution of smallest Euclidean norm to the problem

$$\|\tilde{F}_0 + \tilde{A}\xi + R_0 \tilde{\zeta}\|_2^2 = \min_{\xi,\tilde{\zeta}}!$$
$$\text{s.t. } \tilde{F}_c + R_c \xi = 0.$$

The matrix $\tilde{J}^{-}$ is a generalized inverse of

$$\tilde{J} := \begin{pmatrix} \tilde{A} & R_0 \\ R_c & 0 \end{pmatrix}. \tag{3.38}$$

Taking the above transformations into account we obtain

$$J = \begin{pmatrix} Q_0 & 0 \\ 0 & P_1^T \end{pmatrix} \tilde{J} \begin{pmatrix} I_c & 0 \\ 0 & P_2^T \end{pmatrix} Q^T \tag{3.39}$$

and

$$\Delta x = -Q \begin{pmatrix} I_c & 0 \\ 0 & P_2 \end{pmatrix} \tilde{J}^{-} \begin{pmatrix} Q_0^T & 0 \\ 0 & P_1 \end{pmatrix} \begin{pmatrix} F_0 \\ F_c \end{pmatrix} =: -J^{-} \begin{pmatrix} F_0 \\ F_c \end{pmatrix} \tag{3.40}$$

where I_c is the identity matrix in $\mathbb{R}^{n_c \times n_c}$. Since

$$\|\Delta x\|_2 = \|(\xi^T, \tilde{\zeta}^T)^T\|_2$$

Δx is the unique solution of (3.33) with smallest Euclidean norm.

Lemma 3.17 *Let $\tilde{J}$, $\tilde{J}^-$, J and J^- be given as in* (3.37)-(3.40)*. Then,*

$$\tilde{J}^-\tilde{J}\tilde{J}^- = \tilde{J}^- \quad \textit{and} \quad J^-JJ^- = J^- \tag{3.41}$$

This means that $\tilde{J}^-$ and J^- are outer inverses to $\tilde{J}$ and J, respectively.

Also,

$$\left(\tilde{J}^-\right)^T\tilde{J}^-\tilde{J} = \left(\tilde{J}^-\right)^T \quad \textit{and} \quad \left(J^-\right)^TJ^-J = \left(J^-\right)^T. \tag{3.42}$$

Proof. Let $A \in \mathbb{R}^{s\times t}$ be some arbitrary rectangular matrix. Then by the definition of the Moore-Penrose pseudo-inverse, see e.g. [17], it holds that

$$A^\dagger AA^\dagger = A^\dagger. \tag{3.43}$$

Furthermore, it is directly seen from a singular value decomposition of A that

$$\left(A^\dagger\right)^TA^\dagger A = \left(A^\dagger\right)^T. \tag{3.44}$$

Calculating the products $\tilde{J}^-\tilde{J}\tilde{J}^-$ and J^-JJ^-, respectively, by means of the given decompositions in (3.37)-(3.40) and exploiting (3.43) for $A = R_0$ yields the first two relations.

To show that the relations (3.42) hold calculate the products $\left(\tilde{J}^-\right)^T\tilde{J}^-\tilde{J}$ and $\left(J^-\right)^TJ^-J$, respectively, by means of the decompositions in (3.37)-(3.40) and employ (3.44) for $A = R_0$.

■

Remark 3.18 The relations (3.41) regarding an outer inverse property of $\tilde{J}^-$ and J^- are also stated and proved in [5]. However, there the relations (3.42) are not considered for the rank deficient case. We present these relations here because we will make use of them in Paragraph *General decomposition of* $\chi(\lambda)$ below. □

Remark 3.19 In practice ill-conditioning of J is handled by applying regularization methods to R_0. This is reasonable since small perturbations in R_0 refer to small perturbations in J as it is seen from (3.38) and (3.39). □

3.3.1 Introducing the projection

We turn to an adaption of the analysis which led to (3.13). Recall that we suppose Assumption 3.16 to hold. For $x \in \mathcal{D}$ we abbreviate: $F := F(x)$, $J := J(x)$ and $\Delta x := -J^-F$ where J^- is the outer inverse of J from (3.40). Also, we assume that $\Delta x \neq 0$. This means that x is not a stationary point of the problem (3.31) (cf. Lemma 3.1.18 and Theorem 3.1.31 in [5]). Furthermore, let

$$\Lambda := \{\lambda \in (0,1] \mid x + \lambda\Delta x \in \mathcal{D}\}. \tag{3.45}$$

In analogy to (3.12) we define for $0 \leqslant \sigma \leqslant 1$

$$A(\sigma) := U\Sigma(\sigma)U^TJ^- \tag{3.46}$$

with $U, \Sigma(\sigma)$ given as in (3.12) though Δx being $\Delta x = -J^- F$. It holds that $A(\sigma)F = J^- F \neq 0$ by the assumption $\Delta x \neq 0$, cf. (3.15). Furthermore, since J^- is an outer inverse and by the definition of U and $\Sigma(\sigma)$,

$$A(\sigma)J\Delta x = -U\Sigma(\sigma)U^T J^- JJ^- F = -U\Sigma(\sigma)U^T J^- F = \Delta x. \tag{3.47}$$

Define for $\lambda \in \Lambda$,

$$\chi(\lambda) := J^- \big(F(x + \lambda\Delta x) - F - \lambda J\Delta x\big). \tag{3.48}$$

Let P_{GN} be the orthogonal projector onto the Gauß Newton correction Δx. With the decomposition

$$\begin{aligned} \chi(\lambda) &= P_{GN}\chi(\lambda) + (I - P_{GN})\chi(\lambda) \\ &=: \mu(\lambda)J^- F + \chi_\perp(\lambda) \end{aligned} \tag{3.49}$$

we obtain

$$U\Sigma(\sigma)U^T\chi(\lambda) = \mu(\lambda)J^- F + \sigma \cdot \chi_\perp(\lambda).$$

Hence, it holds that

$$\begin{aligned} A(\sigma)F(x + \lambda\Delta x) &= A(\sigma)(F + \lambda J\Delta x + F(x + \lambda\Delta x) - F - \lambda J\Delta x) \\ &= \big(1 - \lambda + \mu(\lambda)\big)J^- F + \sigma \cdot \chi_\perp(\lambda) \end{aligned} \tag{3.50}$$

and therefore

Proposition 3.20 *Let F_0 and F_c fulfill Assumption 3.16 and for $x \in \mathcal{D}$ non-stationary let $F := F(x)$, $J := J(x)$ be defined via* (3.32)*. Furthermore, let J^- be an outer inverse of J of type* (3.40)*. Then, with $A(\sigma)$ from* (3.46)*, by the decomposition* (3.49) *and for $\lambda \in \Lambda$ with Λ from* (3.45)*, for $0 \leqslant \sigma \leqslant 1$ we obtain*

$$\frac{T\big(x + \lambda\Delta x|A(\sigma)\big)}{T\big(x|A(\sigma)\big)} = \frac{T\big(x + \lambda\Delta x|A(\sigma)\big)}{T\big(x|J^-\big)} = \big(1 - \lambda + \mu(\lambda)\big)^2 + \sigma^2\frac{\|\chi_\perp(\lambda)\|_2^2}{\|\Delta x\|_2^2}.$$

Considering the extremal values $\sigma = 1$ and $\sigma = 0$ yields

$$\sigma = 1: \qquad \frac{T\big(x + \lambda\Delta x|J^-\big)}{T\big(x|J^-\big)} = \big(1 - \lambda + \mu(\lambda)\big)^2 + \frac{\|\chi_\perp(\lambda)\|_2^2}{\|\Delta x\|_2^2}$$

and

$$\sigma = 0: \qquad \frac{T\big(x + \lambda\Delta x|P_{GN}J^-\big)}{T\big(x|P_{GN}J^-\big)} = \frac{T\big(x + \lambda\Delta x|P_{GN}J^-\big)}{T\big(x|J^-\big)} = \big(1 - \lambda + \mu(\lambda)\big)^2.$$

This result motivates an extension of Definition 3.7.

Definition 3.21 (Generalized projected natural level function) *Let F_0 and F_c fulfill Assumption 3.16 and for $x_l \in \mathcal{D}$ non-stationary let $F(x_l)$, $J(x_l)$ be defined via* (3.32)*. With $J(x_l)^-$ being an outer inverse of $J(x_l)$ of type* (3.40) *and with the orthogonal projector*

$$P_{GN_l} := \frac{\Delta x_l \Delta x_l^T}{\Delta x_l^T \Delta x_l}, \quad \Delta x_l := -J(x_l)^- F(x_l),$$

we call

$$T(x|P_{GN_l}J(x_l)^-) = \tfrac{1}{2}\|P_{GN_l}J(x_l)^- F(x)\|_2^2$$

the generalized projected natural level function *(at x_l).*

As the above proposition shows and in analogy to the Newton case there is a good chance that larger step sizes are possible compared to the generalized natural level function if we employ the generalized projected natural level function at some iterate x_l.

Remark 3.22 Provided (3.35) holds true it is readily seen that the generalized projected natural level function features the same S-invariance properties as the generalized natural level function. □

Remark 3.23 The results of Proposition 3.20 are obtained by exploiting the outer inverse property of J^-. So if there is a second outer inverse $\bar{J}^-$ of J with $\bar{J}^- F \neq 0$ we may substitute $\bar{J}^-$ for J^- in all quantities of Proposition 3.20 which depend on J^- to obtain analogous results to the stated ones. □

3.3.1.1 Steepest Descent

In our simplified notation it follows from Lemma 3.17 that at $x \in \mathcal{D}$,

$$\begin{aligned}\operatorname{grad} T\big(x|A(\sigma)\big) &= \big(A(\sigma)F\big)^T A(\sigma)J \\ &= \big(J^-F\big)^T U\Sigma(\sigma)U^T J^- J = \big(J^-F\big)^T J^- J \\ &= \big(J^-F\big)^T = -\Delta x^T.\end{aligned}$$

So just like in the Newton case determining step sizes by means of level functions of the above type, which especially includes the generalized natural and projected natural level function, this procedure may be seen as a modified Steepest Descent method.

3.3.1.2 Invariant core

The term $\big(1-\lambda+\mu(\lambda)\big)^2$ from (3.50) may be interpreted as an invariant core in the following sense. Let $\operatorname{rank}(J) =: n' \leqslant n$ but still $n' \geqslant n_c$. It can be shown that there is a $\hat{J} \in \mathbb{R}^{n_0+n_c \times n_0+n_c-n'}$ with

$$\operatorname{img}\left(\begin{pmatrix} J & \hat{J} \end{pmatrix}\right) = \mathbb{R}^{n_0+n_c} \quad \text{and} \quad J^-\hat{J} = 0.$$

Also, there is a mapping $\hat{\chi} : \Lambda \to \mathbb{R}^{n_0+n_c-n'}$ such that

$$F(x+\lambda\Delta x) = F + J\cdot\big(\lambda\Delta x + \chi(\lambda)\big) + \hat{J}\hat{\chi}(\lambda) \quad \forall\lambda \in \Lambda.$$

Consider weights

$$A \in M := \{W \in \mathbb{R}^{n\times(n_0+n_c)} \,|\, W(I-JJ^-)F = 0 \text{ and } WF \neq 0\}.$$

A restriction is justifiable: By choosing $A \in M$ both first order conditions (2.12) and (2.13) for the associated level function T are fulfilled. With the definition

$$\begin{aligned}r(\lambda|A) := 2\Big[&\big(1-\lambda+\mu(\lambda)\big)(AF)^T\big(AJ\chi_\perp(\lambda) + A\hat{J}\hat{\chi}(\lambda)\big) \\ &+ \big(AJ\chi_\perp(\lambda)\big)^T A\hat{J}\hat{\chi}(\lambda)\Big] + \|A\hat{J}\hat{\chi}(\lambda)\|_2^2\end{aligned}$$

a short calculation shows that

$$\frac{T(x+\lambda\Delta x|A)}{T(x|A)} = \big(1-\lambda+\mu(\lambda)\big)^2 + \frac{\|AJ\chi_\perp(\lambda)\|_2^2}{\|AF\|_2^2} + \frac{r(\lambda|A)}{\|AF\|_2^2} \qquad \forall\lambda\in\Lambda.$$

It is seen from this relation that $\big(1-\lambda+\mu(\lambda)\big)^2$ cannot be changed by A. Note that $A(\sigma) \in M$ and that $r\big(\lambda|A(\sigma)\big) \equiv 0\ \forall\lambda\in\Lambda$.

3.3.1.3 General decomposition of $\chi(\lambda)$

In analogy to the Newton case the invariant core $\big(1-\lambda+\mu(\lambda)\big)^2$ depends on the choice of decomposition of $\chi(\lambda)$. An orthogonal decomposition is again reasonable: Follow the lines of Subsection 3.2.6 until (3.28) and adapt the quantities to fit into the current context. Then we have for $A \in M$,

$$AF(x+\lambda\Delta x) = \big(1-\lambda+\tilde{\mu}(\lambda)\big)AF + AJ\chi_{\tilde{V}}(\lambda) + A\hat{J}\hat{\chi}(\lambda)$$

where M, $\hat{J}$ and $\hat{\chi}$ are defined as in the above paragraph. Consider the choice $A = uw^TJ^-$ in analogy to the one in Subsection 3.2.6. It holds that $uw^TJ^- \in M$ since $uw^TJ^-(I-JJ^-)F = 0$ by means of J^- being an outer inverse and $uw^TJ^-F \neq 0$ since $w^TJ^-F \neq 0$ by means of the fact that V and $\begin{pmatrix} w & \tilde{V} \end{pmatrix}$ are nonsingular, cf. Subsection 3.2.6. Furthermore, it follows from $J^-JJ^- = J^-$ and the definition of $\chi(\lambda)$ in (3.48) that

$$J^-J\chi_{\tilde{V}}(\lambda) = J^-J\cdot\big(\chi(\lambda)-\tilde{\mu}(\lambda)J^-F\big) = \chi(\lambda)-\tilde{\mu}(\lambda)J^-F = \chi_{\tilde{V}}(\lambda).$$

Hence,

$$uw^TJ^-F(x+\lambda\Delta x) = \big(1-\lambda+\tilde{\mu}(\lambda)\big)uw^TJ^-F.$$

We continue following and adapting the lines in Subsection 3.2.6. In the current context the gradient of the reduced level function R_w at x turns out to be

$$\operatorname{grad} R_w(x) = 2w^TJ^-F\cdot w^TJ^-J.$$

For the choice $w = \alpha\cdot J^-F$, $\alpha \neq 0$, we obtain by means of Lemma 3.17,

$$w^TJ^-J = \alpha\cdot F^T\big(J^-\big)^TJ^-J = \alpha\cdot F^T\big(J^-\big)^T = w^T$$

which means that Δx is a direction of steepest descent for R_w at x and this particular choice of w. Opting for this choice of w it follows from the same arguments we used in Subsection 3.2.6 that

$$\chi_{\tilde{V}}(\lambda) = (I-P_N)\chi(\lambda)$$

and

$$\tilde{\mu}(\lambda) = \mu(\lambda)$$

holds which reflects our choice of orthogonal decomposition.

3.3.1.4 Asymptotic error measurement

Let $x_* \in \mathcal{D}$ be a stationary point of (3.33) such that $J(x_*)$ has full rank. For a full rank Jacobian $J(x)$ it holds that $J^-(x)J(x) = I$. This follows directly from the decompositions (3.39) and (3.40) and the fact that R_0 has full rank and hence $R_0^\dagger R_0 = I$.

Taylor expansion of $J^-(x_*)F(x)$ yields

$$J^-(x_*)F(x) = x - x_* + o(\|x-x_*\|_2).$$

So it is reasonable in analogy to the Newton case to define a termination criterion based on $J^-(x_l)F(x_{l+1})$, continuing the iteration as long as this criterion is not matched, even if the value of the generalized projected natural level function at the current iterate is zero.

3.3.1.5 Structured Jacobians from a multiple shooting context

The matrix $F_c'(x)$ is of special structure in a multiple shooting context. To give a reason why this is the case we will briefly describe the techniques of multiple shooting in the context of solving boundary value problems (BVPs) and parameter estimation problems in ordinary differential equations (ODEs). Detailed information on the multiple shooting approach in the aforementioned contexts may be found in [32, 5, 11]. The special structure of F_c' is exploited in a condensing algorithm, [32, 10, 5], to obtain a linearized problem which is of much smaller dimension than (3.33). We will discuss this approach in terms of a related general inverse of the Jacobian and provide associated projected natural level functions.

BVPs

Consider the BVP

$$y' = f(y), \qquad r\big(y(a), y(b)\big) = 0 \tag{3.51}$$

where both $f : \mathbb{R}^{n_d} \to \mathbb{R}^{n_d}$, the right side of the ODE, and $r : \mathbb{R}^{n_d} \times \mathbb{R}^{n_d} \to \mathbb{R}^{n_d}$, the boundary conditions, are twice continuously differentiable. Assume that the above BVP has a locally unique solution. If one applies multiple shooting techniques to solve this problem a partition of the interval $[a, b]$ is considered, i.e.,

$$a = \tau_1 < \tau_2 < \cdots < \tau_m = b, \quad m > 2.$$

Let s_j be estimates of the unknown values of y at the nodes τ_j, $j = 1, \ldots, m$. Then $m - 1$ initial value problems (IVPs) of the form

$$y' = f(y), \quad y(\tau_j) = s_j, \quad t \in [\tau_j, \tau_{j+1}], \quad j = 1, \ldots, m-1,$$

are solved. The solution of the j-th IVP may be denoted by $y(t; s_j)$. Since the solution of the problem (3.51) is continuous the $m - 1$ matching conditions

$$h_j(s_j, s_{j+1}) := y(\tau_{j+1}; s_j) - s_{j+1} = 0, \quad j = 1, \ldots, m-1$$

have to be met. Also,

$$r(s_1, s_m) = 0$$

has to be fulfilled. Putting things together, by defining $n := n_d \cdot m$ and

$$x := \begin{pmatrix} s_1 \\ \vdots \\ s_m \end{pmatrix} \in \mathbb{R}^n, \quad F(x) := \begin{pmatrix} r(s_1, s_m) \\ h_1(s_1, s_2) \\ \vdots \\ h_{m-1}(s_{m-1}, s_m) \end{pmatrix}$$

we obtain the problem

$$F(x) = 0.$$

This problem may be solved by means of Newton's method. The arising Jacobians are of special structure, i.e.,

$$F'(x) = \begin{pmatrix} B_1 & & & B_m \\ G_1 & -I & & \\ & \ddots & \ddots & \\ & & G_{m-1} & -I \end{pmatrix} \tag{3.52}$$

with

$$B_1 := \frac{\partial}{\partial s_1} r(s_1, s_m), \qquad B_m := \frac{\partial}{\partial s_m} r(s_1, s_m)$$

and the Wronskian matrices

$$G_j := \frac{\partial}{\partial s_j} y(\tau_{j+1}; s_j), \quad j = 1, \ldots, m-1.$$

Before we will consider the condensing algorithm which exploits the structure of the above Jacobian we will turn to the problem of parameter estimation in ODEs. The structure of the associated Jacobians is very similar to the one above, in fact, it is a generalization of it. So condensing may be discussed in this context also covering the above Jacobian form as a special case.

Parameter estimation in ODEs

Consider the parameter dependent IVP

$$y' = f(y, p), \quad y(t_1) = s_1$$

with $f : \mathbb{R}^{n_d} \times \mathbb{R}^{n_p} \to \mathbb{R}^{n_d}$ continuously differentiable and $p \in \mathbb{R}^{n_p}$. In the context of parameter estimations in ODEs p and s_1 are to be determined such that the solution $y(t; s_1, p)$ is closest to some given measured data

$$\tilde{y}_i \quad \text{at} \quad t_i \geqslant t_1, \quad i = 1, \ldots, \iota,$$

in a least squares sense which is in the most simple case – neglecting statistical considerations and assuming complete measures – represented via

$$\sum_{i=1}^{\iota} \| y(t_i; s_1, p) - \tilde{y}_i \|_2^2 = \min_{s_1, p}!$$

This problem may be approached via single shooting techniques. However, multiple shooting turns out to be a lot more robust, see e.g. the notorious test problem 1 in [4]. Hence, we select additional nodes out of the set of measurement nodes, i.e.,

$$\{\tau_1, \ldots, \tau_m\} \subseteq \{t_1, \ldots, t_\iota\}, \quad \tau_1 = t_1, \ \tau_m = t_\iota,$$

where usually $m \ll \iota$. Accordingly, the additional unknowns $s_2, \ldots, s_m \in \mathbb{R}^{n_d}$ are introduced. By means of $s_1, \ldots, s_{m-1}$ we define $m-1$ sub-trajectories via

$$y' = f(y, p), \quad y(\tau_j) = s_j, \quad t \in [\tau_j, \tau_{j+1}], \quad j = 1, \ldots m-1.$$

With an appropriate index function $j : \{2, \dots, \iota - 1\} \to \{1, \dots m - 1\}$ and

$$r_1 = r_1(s_1, \dots, s_m, p) := \begin{pmatrix} s_1 - \tilde{y}_1 \\ y(t_2; s_{j(2)}, p) - \tilde{y}_2 \\ \vdots \\ y(t_{\iota-1}; s_{j(\iota-1)}, p) - \tilde{y}_{\iota-1} \\ s_m - \tilde{y}_\iota \end{pmatrix}$$

of dimension $n_1 := \iota \cdot n_d$ we are led to the constrained least squares problem

$$\begin{aligned} \|r_1(s_1, \dots, s_m, p)\|_2^2 &= \min! \\ \text{s.t. } h_j = h_j(s_j, s_{j+1}) &:= y(\tau_{j+1}; s_j) - s_{j+1} = 0, \quad j = 1, \dots, m-1. \end{aligned} \tag{3.53a}$$

This problem may be further extended by additional constraints

$$r_2 = r_2(s_1, \dots, s_m, p) = 0 \tag{3.53b}$$

with r_2 of dimension n_2 and continuously differentiable.[1] Define $n := n_d \cdot m + n_p$ and

$$x := \begin{pmatrix} s_1 \\ \vdots \\ s_m \\ p \end{pmatrix} \in \mathbb{R}^n, \quad F_0(x) := r_1, \quad F_c(x) := \begin{pmatrix} r_2 \\ h_1 \\ \vdots \\ h_{m-1} \end{pmatrix}, \quad F(x) := \begin{pmatrix} F_0(x) \\ F_c(x) \end{pmatrix}. \tag{3.54}$$

Hence, the problem (3.53) can be written in the notation of (3.31). The Jacobian $J(x) := F'(x)$ is structured, i.e.,

$$J(x) = \begin{pmatrix} F_0'(x) \\ \dots\dots\dots \\ F_c'(x) \end{pmatrix} = \begin{pmatrix} B_{1|1} & \cdots & B_{m-1|1} & B_{m|1} & B_{p|1} \\ \dots\dots\dots & & & & \dots\dots\dots \\ B_{1|2} & \cdots & B_{m-1|2} & B_{m|2} & B_{p|2} \\ G_1 & -I & & & G_{p|1} \\ & \ddots & \ddots & & \vdots \\ & & G_{m-1} & -I & G_{p|m-1} \end{pmatrix} \tag{3.55}$$

with

$$B_{j|k} := \frac{\partial}{\partial s_j} r_k, \quad j = 1, \dots, m, \;\; k = 1, 2,$$

and the Wronskian matrices

$$G_j := \frac{\partial}{\partial s_j} y(\tau_{j+1}; s_j, p), \qquad G_{p|j} := \frac{\partial}{\partial p} y(\tau_{j+1}; s_j, p), \quad j = 1, \dots, m-1.$$

As it is readily seen the above Jacobian is indeed a generalization of (3.52).

We assume that $F_c'(x)$ is of full rank for all $x \in \mathcal{D}$.

[1] Regarding proper incorporation and handling of *inequality* constraints we refer the interest reader to [5].

Remark 3.24 Full rank of $F_c'(x)$ is definitely guaranteed if no further constrains r_2 are considered. Then $F_c'(x)$ solely consists of the derivatives of the matching conditions from (3.53a), i.e.,

$$F_c'(x) = \begin{pmatrix} G_1 & -I & & & G_{p|1} \\ & \ddots & \ddots & & \vdots \\ & & G_{m-1} & -I & G_{p|m-1} \end{pmatrix}.$$

By means of the identity matrices in the first super block diagonal full rank of $F_c'(x)$ is ensured for all $x \in \mathcal{D}$. □

The condensing algorithm, [5], consists of three steps:

(I) Run

Algorithm 3.1 (Backward recursion)

1: initialize: $u_{m|k} := r_k$, $P_{m|k} := B_{p|k}$, $E_{m|k} := B_{m|k}$ for $k = 1, 2$.
2: **for** $j = m : -1 : 2$ **do**
3: $\quad u_{j-1|k} := u_{j|k} + E_{j|k} h_{j-1}$
4: $\quad P_{j-1|k} := P_{j|k} + E_{j|k} G_{p|j-1}$
5: $\quad E_{j-1|k} := B_{j-1|k} + E_{j|k} G_{j-1}$
6: **end for**

(II) Solve for Δs_1 and Δp the condensed problem

$$\begin{aligned} \|u_{1|1} + E_{1|1}\Delta s_1 + P_{1|1}\Delta p\|_2^2 &= \min! \\ \text{s.t. } u_{1|2} + E_{1|2}\Delta s_1 + P_{1|2}\Delta p &= 0 \end{aligned} \tag{3.56}$$

(III) Determine the remaining corrections $\Delta s_2, \ldots, \Delta s_m$ via

Algorithm 3.2 (Forward recursion)

1: **for** $j = 1 : 1 : m - 1$ **do**
2: $\quad \Delta s_{j+1} := G_j \Delta s_j + G_{p|j} \Delta p + h_j$
3: **end for**

Remark 3.25 The backward recursion which leads to the condensed problem can be interpreted in the following way. Let $H_c \in \mathbb{R}^{(m-1)\cdot n_d \times n}$ be the submatrix of $F_c'(x)$ which contains the derivatives of the matching conditions and let $h := (h_1^T, \ldots, h_{m-1}^T)^T$. Then, to solve the linearized problem (3.33) it is necessary that the increment Δx fulfills

$$H_c \Delta x + h = 0. \tag{3.57}$$

Since H_c has full rank the set of solutions to the above linear system is an affine space with an associated vector space of dimension $n-(m-1)\cdot n_d = n_d + n_p$. Because of the special structure of H_c all solutions of (3.57) can be expressed as

$$b + A \begin{pmatrix} \Delta s_1 \\ \Delta p \end{pmatrix} \tag{3.58}$$

for some specific $b \in \mathbb{R}^n$ and $A \in \mathbb{R}^{n \times n_d + n_p}$. E.g., for $m = 3$ we have

$$b = \begin{pmatrix} 0 \\ h_1 \\ G_2 h_1 + h_2 \\ 0 \end{pmatrix} \quad \text{and} \quad A = \begin{pmatrix} I & 0 \\ G_1 & G_{p|1} \\ G_2 G_1 & G_2 G_{p|1} + G_{p|2} \\ 0 & I \end{pmatrix}.$$

Setting Δx to (3.58) and multiplying Δx with $J(x)$ resembles the backward recursion. Appropriate rearranging of terms leads to the condensed problem (3.56). Via this interpretation of the backward recursion it is readily seen that in case of a unique solution of (3.33) this solution is also obtained by means of the condensing algorithm. □

The above system (3.56) is dense but of small dimension $(n_1 + n_2) \times (n_d + n_p)$. The matrix $E_{1|2}$ has full rank by the assumption that $F_c'(x)$ has full rank—see Remark 3.26 below. Defining

$$E_1 := \begin{pmatrix} E_{1|1} \\ E_{1|2} \end{pmatrix} \quad \text{and} \quad P_1 := \begin{pmatrix} P_{1|1} \\ P_{1|2} \end{pmatrix}$$

the solution (of smallest Euclidean norm) of (3.56) is given via

$$\begin{pmatrix} \Delta p \\ \Delta s_1 \end{pmatrix} = - \begin{pmatrix} P_1 & E_1 \end{pmatrix}^- \begin{pmatrix} u_{1|1} \\ u_{1|2} \end{pmatrix} \tag{3.59}$$

where the generalized inverse is of the type defined in (3.40). The whole process of determining all corrections can be stated in terms of a generalized inverse of $J := J(x)$. Therefore, define E_j, $j = 2, \ldots, m$ in analogy to E_1. Then, by Lemma 4.1.13 in [5],

$$J\Pi = LSR \tag{3.60}$$

with

$$\begin{aligned} \Pi &:= \begin{pmatrix} & I_{n_d \cdot m} \\ I_{n_p} & \end{pmatrix}, & L &:= \begin{pmatrix} I_{n_1+n_2} & -E_2 & \cdots & -E_m \\ & I_{n_d} & & \\ & & \ddots & \\ & & & I_{n_d} \end{pmatrix}, \\ S &:= \begin{pmatrix} P_1 & E_1 & & & \\ & & I_{n_d} & & \\ & & & \ddots & \\ & & & & I_{n_d} \end{pmatrix}, & R &:= \begin{pmatrix} I_{n_p} & & & & \\ & I_{n_d} & & & \\ G_{p|1} & G_1 & -I_{n_d} & & \\ \vdots & & \ddots & \ddots & \\ G_{p|m-1} & & & G_{m-1} & -I_{n_d} \end{pmatrix}. \end{aligned} \tag{3.61}$$

Since L and R are nonsingular matrices it follows from basic linear algebra, see also Lemma 4.1.15 in [5], that

$$\operatorname{rank}(J) = \operatorname{rank}(S) = \operatorname{rank}\left(\begin{pmatrix} P_1 & E_1 \end{pmatrix}\right) + (m-1)\cdot n_d$$

which means that rank deficiency of J implies rank deficiency of S.

Remark 3.26 Full rank of $F_c'(x)$ implies full rank of $\begin{pmatrix} P_{1|2} & E_{1|2} \end{pmatrix}$. This is easily seen by contradiction. Let $v \in \mathbb{R}^{n_2} \setminus \{0\}$ such that $v^T \begin{pmatrix} P_{1|2} & E_{1|2} \end{pmatrix} = 0$. Then with 0_k being the zero vector in $\mathbb{R}^k$ and $\tilde{v} := L^{-T}(0_{n_1}^T, v^T, 0_{(m-1)\cdot n_d}^T)^T$ it holds that $\tilde{v}^T J = 0$ by the above decomposition (3.60). Furthermore, by the definition of L in (3.61) the vector $\tilde{v}$ may be written as $\tilde{v} = (0_{n_1}^T, v^T, w^T)^T$ for some $w \in \mathbb{R}^{(m-1)\cdot n_d}$. Hence, $\tilde{v}^T J = 0$ implies $(v^T, w^T)F_c'(x) = 0$. This is a contradiction. □

Proposition 3.27 *Under Assumption 3.16 let $F := F(x)$ be given as in* (3.54) *and $J := J(x)$ from* (3.55) *via the decomposition* (3.60) *for some $x \in \mathcal{D}$. Then, with*

$$S^- := \begin{pmatrix} \begin{pmatrix} P_1 & E_1 \end{pmatrix}^- & & & \\ & I_{n_d} & & \\ & & \ddots & \\ & & & I_{n_d} \end{pmatrix}$$

and $\begin{pmatrix} P_1 & E_1 \end{pmatrix}^-$ from (3.59) *the matrix*

$$J^- := \Pi R^{-1} S^- L^{-1} \tag{3.62}$$

is an outer generalized inverse of J, i.e.,

$$J^- J J^- = J^-.$$

The correction

$$\Delta x := -J^- F$$

is the unique solution of the linearized problem (3.33) *such that the subvector $(\Delta p^T, \Delta s_1^T)^T$ of Δx is of smallest Euclidean norm.*

Proof. The outer inverse property of J^- is directly verified via the definition of J^-, the decomposition (3.60) and Lemma 3.17 since $\begin{pmatrix} P_1 & E_1 \end{pmatrix}^-$ is a generalized inverse of the type defined in (3.40). That Δx is a solution to (3.33) follows from the fact that the product which defines Δx reflects the condensing algorithm in a matrix vector notation. Uniqueness and the minimal property of Δx are consequences of (3.59) and the forward recursion.

■

Note that the above result is also true for rank deficient J.

Remark 3.28 If there are no additional constrains r_2 the generalized inverse in (3.59) reduces to the Moore-Penrose pseudo-inverse of $\begin{pmatrix} P_{1|1} & E_{1|1} \end{pmatrix}$. Furthermore, in the context of solving BVPs there is no dependence on a parameter vector p and hence $\begin{pmatrix} P_{1|1} & E_{1|1} \end{pmatrix}$ simplifies to $E_{1|1}$. L and R in (3.60) are changed accordingly as well as Π becomes the identity matrix to match (3.52) via the decomposition (3.60). If J is also nonsingular then $J^- = J^{-1}$. □

By the above proposition and according to Remark 3.23 an adaption of the results of Proposition 3.20 to the case where J^- is defined via (3.62) is straightforward. This is also true for the definition of the generalized projected natural level function from Definition 3.21. However, utilizing the above defined J^- in the definition of $A(\sigma)$ in (3.46) the correction $\Delta x = -J^- F$ is not necessarily a direction of steepest descent for $T\big(\cdot\,|A(\sigma)\big)$ since

$$-\operatorname{grad} T\big(x|A(\sigma)\big) = -F^T L^{-T}(S^-)^T R^{-T} R^{-1} S^- S R \Pi^T.$$

To gain this property consider the transformation of variables

$$y := \tilde{R}x, \quad \tilde{R} := R\Pi^T, \tag{3.63}$$

which yields

$$G(y) := F\big(\tilde{R}^{-1}y\big), \quad G'(y) = F'\big(\tilde{R}^{-1}y\big)\tilde{R}^{-1} = LS =: J_y \tag{3.64}$$

and

$$\Delta y := -J_y^- G(y) = -S^- L^{-1} G(y) = \tilde{R}\Delta x \tag{3.65}$$

where S^- is given as in Proposition 3.27. Define for $0 \leqslant \sigma \leqslant 1$

$$B(\sigma) := U_y \Sigma(\sigma) U_y^T J_y^-$$

with

$$\Sigma(\sigma) := \operatorname{diag}(1, \sigma, \ldots, \sigma) \in \mathbb{R}^{n\times n}, \quad U_y := \Big(\Delta y/\|\Delta y\|_2 \quad \tilde{U}_y\Big),\ \tilde{U}_y \in \mathbb{R}^{n\times n-1} \quad \text{s.t.} \quad U_y^T U_y = I$$

and also

$$T\big(y|B(\sigma)\big) := \tfrac{1}{2}\|B(\sigma)G(y)\|_2^2.$$

Then,

$$\begin{aligned}\operatorname{grad} T\big(y|B(\sigma)\big) &= \big(B(\sigma)G(y)\big)^T B(\sigma) J_y \\ &= G(y)^T L^{-T}\big(S^-\big)^T S^- L^{-1} L S \\ &= G(y)^T L^{-T}\big(S^-\big)^T = -\Delta y^T.\end{aligned}$$

So for the transformed system a steepest descent property of the Gauß Newton correction is ensured. The projected natural level function associated with the transformed system reads as follows

$$T\big(y|B(0)\big) = \frac{1}{2}\left\| \frac{\Delta y \Delta y^T}{\Delta y^T \Delta y} J_y^- G(y) \right\|_2^2. \tag{3.66}$$

Undoing the transformation (3.63) gives rise to the definition of a second projected natural level function.

Definition 3.29 ($\tilde{R}_l$-norm related projected natural level function) *Let F_0 and F_c fulfill Assumption 3.16 and for $x_l \in \mathcal{D}$ non-stationary let $F(x_l)$, $J(x_l)$ be defined via (3.54) and (3.55), respectively. Consider the Gauß Newton correction $\Delta x_l = -J^-(x_l)F(x_l)$ where $J^-(x_l)$ is an outer inverse of $J(x_l)$ defined via Proposition 3.27. Let $P_{GN,\tilde{R}_l}$ be the orthogonal projector onto the Gauß Newton correction Δx_l w.r.t. the inner product*

$$\langle z_1, z_2\rangle_{\tilde{R}} := z_1 \tilde{R}_l^T \tilde{R}_l z_2 \quad \forall z_1, z_2 \in \mathbb{R}^n,$$

i.e.,

$$P_{GN,\tilde{R}_l} = \frac{\Delta x_l \Delta x_l^T \tilde{R}_l^T \tilde{R}_l}{\Delta x_l^T \tilde{R}_l^T \tilde{R}_l \Delta x_l}$$

and let $\|\cdot\|_{\tilde{R}_l}$ *be the norm induced by the above inner product. Then we call*

$$T_{\tilde{R}_l}\big(x|P_{GN}J^-(x_l)\big) := \tfrac{1}{2}\|P_{GN,\tilde{R}_l}J^-(x_l)F(x)\|_{\tilde{R}_l}^2 = \frac{1}{2}\left\|\tilde{R}_l\frac{\Delta x_l \Delta x_l^T \tilde{R}_l^T \tilde{R}_l}{\Delta x_l^T \tilde{R}_l^T \tilde{R}_l \Delta x_l}J^-(x_l)F(x)\right\|_2^2$$

the $\tilde{R}_l$-norm related projected natural level function *(at* x_l*).*

Corollary 3.30 *For* $T_{\tilde{R}_l}\big(x|P_{GN}J^-(x_l)\big)$ *from the above definition its relative change in the direction of* Δx_l *and for* $\lambda \in \Lambda_l$ *is given via*

$$\frac{T_{\tilde{R}_l}\big(x_l + \lambda\Delta x_l|P_{GN}J^-(x_l)\big)}{T_{\tilde{R}_l}\big(x_l|P_{GN}J^-(x_l)\big)} = \big(1 - \lambda + \tilde{\mu}_l(\lambda)\big)^2$$

with

$$\tilde{\mu}_l(\lambda) = -\frac{\Delta x_l^T \tilde{R}_l^T \tilde{R}_l}{\Delta x_l^T \tilde{R}_l^T \tilde{R}_l \Delta x_l} J(x_l)^-\big(F(x_l + \lambda\Delta x_l) - F(x_l) - \lambda J(x_l)\Delta x_l\big)$$

and Λ_l *defined in accordance to* (3.45)*.*

Proof. Since J_y^- from (3.65) is an outer inverse of J_y from (3.64) and according to Remark 3.23 the results of Proposition 3.20 may be exploited for the level function from (3.66). Undoing the change of variables (3.63) and introducing indices yields the above result. ■

Remark 3.31 A natural level function which is related to the $\tilde{R}_l$-norm is already stated in [10] in the context of solving BVPs via multiple shooting techniques. In our notation and neglecting scaling this function is defined at x_l via

$$T_{\tilde{R}_l}\big(x|J^-(x_l)\big) = \tfrac{1}{2}\|J^-(x_l)F(x)\|_{\tilde{R}_l}^2.$$

With the level function from Definition 3.29 we provide a natural extension of the above level function to the family of projected natural level functions though not only in the context of solving BVPs by means of multiple shooting but also in the context of parameter estimation problems approached via multiple shooting methods. Note that

$$T_{\tilde{R}_l}\big(x|P_{GN}J^-(x_l)\big) \leqslant T_{\tilde{R}_l}\big(x|J^-(x_l)\big)$$

which once again means that it is very likely that larger step sizes are possible. □

3.4 Step Size Control

In this section we will provide algorithms to determine the step sizes λ_l in a damped Newton iteration

$$x_{l+1} = x_l + \lambda_l \Delta x_l, \quad \Delta x_l = -F'(x_l)^{-1} F(x_l), \quad \lambda_l \in (0,1], \tag{3.67}$$

based on the concept of the projected natural level function (PNLF)

$$\begin{aligned} T\big(x|P_{N_l}F'(x_l)^{-1}\big) &= \tfrac{1}{2}\|P_{N_l}F'(x_l)^{-1}F(x)\|_2^2, \\ P_{N_l} &= \frac{\Delta x_l \Delta x_l^T}{\Delta x_l^T \Delta x_l}, \quad \Delta x_l = -F'(x_l)^{-1}F(x_l). \end{aligned} \tag{3.68}$$

We give some introductory remarks:

- The algorithms we will present here are related to Newton's method for solving systems of nonlinear equations. However, an extension to Gauß Newton methods for solving least squares problems is straightforward.
- We assume that the Jacobian is computed each step, e.g., by means of Automatic Differentiation techniques. Furthermore, to solve upcoming linear systems we suppose that a reasonable decomposition of the Jacobian exists such that the complexity of solving these systems is of $\mathcal{O}(n^2)$ floating point operations.
- We do not consider any emergency procedures in case $F'(x_l)$ becomes rank deficient. By means of Theorem 2.7 and the change of variable $\lambda = 1 - \exp(-t)$ it is readily seen that the trajectory of the Newton-path $\overline{x}$ for $\lambda \in [0,1]$ can also be characterized by the solution to the IVP

  $$\frac{dx}{dt} = -F'(x)^{-1}F(x), \quad x(0) = x_0,$$

  which fulfills

  $$\lim_{t \to +\infty} x(t) = x_* \ \text{ with } \ F(x_*) = 0.$$

  Hence, the iteration (3.67) may be interpreted as an application of Euler's method in order to follow the Newton path which starts at x_0—see also [6] for this kind of interpretation. By definition the Newton path cannot cross interfaces of singular Jacobians so we like our iteration to behave in the same way. For an iterate x_l in the vicinity of such an interface the related correction Δx_l is usually of large magnitude resulting in the necessity of rigorous damping. If our upcoming step size control suggests a step size smaller than some prescribed λ_{min} we will stop the iteration and abort the algorithm with a convergence failure.

In Subsection 3.2.2 we determined step sizes by means of the global affine covariant nonlinearity bound (3.18), i.e.,

$$2\|P_N(x)F'(x)^{-1}\big(F(y) - F(x) - F'(x)(y-x)\big)\|_2 \leqslant \omega \|y-x\|_2^2 \quad \forall x, y \in \mathcal{D}. \tag{3.69}$$

However, at some iterate x_l we know from (3.17) that the behavior of the PNLF along the Newton correction is described via

$$T(x_l + \lambda \Delta x_l | P_{N_l}F'(x_l)^{-1}) = \big(1 - \lambda + \mu_l(\lambda)\big)^2 T(x_l | P_{N_l}F'(x_l)^{-1}) \tag{3.70}$$

where

$$\mu_l(\lambda) = -\frac{\Delta x_l^T}{\|\Delta x_l\|_2^2} F'(x_l)^{-1}\big(F(x_l + \lambda\Delta x_l) - F(x_l) - \lambda F'(x_l)\Delta x_l\big). \tag{3.71}$$

This means, it is the *local* nonlinearity of F characterized by $\mu(\lambda)$ which has to be taken into account to determine the step size λ_l. So we will substitute the above stated *global* bound by some appropriate *local* bounds and will use these bounds as a basis for a step size control. This way we will obtain local refinements of the polynomial model from Theorem 3.8.

The theoretical framework will be complemented by practically computable estimates of the locally defined theoretical bounds. These estimates are adaptations of already existing quantities from [10, 11, 26, 6, 5]. However, they are in closer relationship to our theoretical quantities since we use nonlinearity bounds instead of Lipschitz conditions on the Jacobian as it is done in the aforementioned literature.

We will present two step size controls.

In Subsection 3.4.1 we will discuss the first approach which is an adaption from the strategies presented by Nowak and Weimann in [26] and by Deuflhard in [11]. Step sizes will be determined according to locally defined polynomial models of the behavior of the PNLF and accepted if *simple monotonicity*, i.e.,

$$T(x_l + \lambda\Delta x_l | P_{N_l} F'(x_l)^{-1}) < T(x_l | P_{N_l} F'(x_l)^{-1})$$

is fulfilled.

In Subsection 3.4.2 we will consider a second approach which is inspired by the work of Bock in [5] and Bock, Kostina and Schlöder in [6]. Like in the first approach, step sizes will be determined by means of locally defined polynomial models. However, step sizes have to pass a *restricted monotonicity* check which is more demanding than the requirement of simple monotonicity. On the other hand this check provides a more satisfactory relationship to its theoretical background. We will explain this check in detail in Subsection 3.4.2. The theoretical framework is slightly more sophisticated than the one for the first approach due to step size dependent nonlinearity bounds.

Though both approaches differ in detail they share common basic concepts. E.g., both step size controls are based on a *predictor-corrector-scheme* to determine step sizes. We will describe these concepts in more detail for the simple monotonicity approach and briefly reconsider them for the second one.

For the upcoming discussions we abbreviate for a given iterate $x_l \in \mathcal{D}$,

$$F_l := F(x_l), \qquad J_l := F'(x_l).$$

In accordance to the definition of Λ in (3.1) we define

$$\Lambda_l := \{\lambda \in (0, 1] \mid x_l + \lambda\Delta x_l \in \mathcal{D}\}. \tag{3.72}$$

Furthermore, some of the upcoming results will require the iterate x_l to fulfill the following assumption:

Assumption 3.32 *It holds that $x_l \in \mathcal{D}$ with $F_l \neq 0$ and J_l nonsingular. With the associated orthogonal projector P_{N_l} defined according to* (3.68) *the closure of the path-connected component $\mathcal{D}_l$ of the level set $\{z \in \mathcal{D} \,|\, T(z|P_{N_l} J_l^{-1}) \leqslant T(x_l|P_{N_l} J_l^{-1})\}$ which contains x_l is a subset of $\mathcal{D}$.*

3.4.1 Simple monotonicity

We will first provide some theoretical background. Our analysis will be based on a local counterpart of the nonlinearity bound (3.69). Then, we will discuss several aspects of an algorithmic realization of a step size control based on simple monotonicity: By means of the estimates for the local nonlinearity bounds a predictor-corrector-scheme will be established to determine step sizes. We will not only provide adaptations of existing predictors from [26, 11] and [5] but will also introduce a new one. This new predictor is cheaply available due to the projectional aspect of the PNLF. We will provide a termination criterion and discuss when an increase of already accepted step sizes will be considered. In Algorithm 3.5 we will combine all the considered aspects to establish a step size control based on simple monotonicity. Furthermore, we will analyze the computational costs of the predictor-corrector-scheme for the several choices of predictors.

3.4.1.1 Theoretical framework

Instead of the global nonlinearity bound (3.69) we assume that for a given iterate $x_l \in \mathcal{D}$ with $\Delta x_l \neq 0$ the local affine covariant projected nonlinearity bound

$$2\|P_{N_l} J_l^{-1}\big(F(y) - F_l - J_l(y - x_l)\big)\|_2 \leqslant \omega_l \|y - x_l\|_2^2 \tag{3.73}$$

for all $y \in \mathcal{D}$ with $y - x_l = \lambda \Delta x_l$, $\lambda \in [0, 1]$, holds. By means of this bound we obtain a local version of Theorem 3.8.

Theorem 3.33 *Let F fulfill Assumption 2.1 and let the iterate x_l fulfill Assumption 3.32. Define Λ_l as in* (3.72) *and $\mathcal{D}_l$ as in Assumption 3.32. Suppose that the local affine covariant projected nonlinearity bound* (3.73) *holds and let*

$$\Lambda_{\mathcal{D}_l} := \{\lambda \in [0, 1] \mid x_l + \lambda \Delta x_l \in \mathcal{D}_l\}.$$

Then,

$$T(x_l + \lambda \Delta x_l | P_{N_l} J_l^{-1}) \leqslant \big(1 - \lambda + \tfrac{1}{2}\omega_l \|\Delta x_l\|_2 \lambda^2\big)^2 T(x_l | P_{N_l} J_l^{-1}) \tag{3.74}$$

for all $\lambda \in \Lambda_l$ and $[0, \min(1, 2\overline{\lambda}_l)] \subseteq \Lambda_{\mathcal{D}_l}$ where

$$\overline{\lambda}_l := \min\left(1, \frac{1}{\omega_l \|\Delta x_l\|_2}\right)$$

with ω_l from (3.73) *is the unique minimizer in $[0, 1]$ of the above polynomial estimate.*

Proof. We follow the lines of the proof of Theorem 3.8 and exploit that for $\lambda \in \Lambda_l$, $\lambda > 0$, (3.19) holds with ω_l substituted for $\omega_{(3.18)}$. Note that the set $\Lambda_{\mathcal{D}_l} \setminus \{0\}$ is not empty since $x_l \in \mathcal{D}$, $\mathcal{D}$ is open and Δx_l is a direction of descent for $T(x|P_{N_l} J_l^{-1})$ at x_l.

■

This theorem gives rise to the following algorithm

Algorithm 3.3 (Theoretical step size control, a basic scheme)

---

given: x_0 which fulfills Assumption 3.32
determine Δx_0
set $l = 0$
while $\|\Delta x_l\|_2 \neq 0$ **do**
    determine ω_l from (3.73)
    set $\lambda_l = \overline{\lambda}_l$ from Theorem 3.33, i.e., $\lambda_l = \min\left(1, \frac{1}{\omega_l \|\Delta x_l\|_2}\right)$
    set $x_{l+1} = x_l + \lambda_l \Delta x_l$
    set $l = l + 1$
    determine Δx_l
end while

Bounded influence of the local nonlinearity

For $x_l + \lambda \Delta x_l \in \mathcal{D}$ we introduce indices in (3.2) and set $A = P_{N_l} J_l^{-1}$ to obtain

$$P_{N_l} J_l^{-1} F(x_l + \lambda \Delta x_l) - P_{N_l} J_l^{-1} F_l = \lambda \Delta x_l + P_{N_l} \chi_l(\lambda) = \lambda \Delta x_l + o(\lambda)$$

since $P_{N_l} \Delta x_l = \Delta x_l$. Furthermore, by the definition of ω_l in (3.73) and for given $\lambda \in (0, \lambda_l]$ with λ_l from the above algorithm it holds that

$$\begin{aligned} \frac{\|P_{N_l} \chi_l(\lambda)\|_2}{\lambda \|\Delta x_l\|_2} &= \frac{\|P_{N_l} J_l^{-1}\big(F(x_l + \lambda \Delta x_l) - F_l - \lambda J_l \Delta x_l\big)\|_2}{\lambda \|\Delta x_l\|_2} \\ &\leqslant \tfrac{1}{2} \lambda \omega_l \|\Delta x_l\|_2 \leqslant \frac{1}{2}. \end{aligned} \tag{3.75}$$

Hence, by means of the triangular inequality we obtain

$$1 - \frac{1}{2} \leqslant \frac{\|P_{N_l} J_l^{-1} F(x_l + \lambda \Delta x_l) - P_{N_l} J_l^{-1} F_l\|_2}{\lambda \|\Delta x_l\|_2} \leqslant 1 + \frac{1}{2}. \tag{3.76}$$

This means that λ_l is the maximum step size for which the polynomial model in (3.74) guarantees that the change of $P_{N_l} J_l^{-1} F$ is dominated by the first order term, i.e., $\lambda \Delta x_l$. The influence of the nonlinearity is restricted to the second binary digit. Since $\|P_{N_l} \chi_l(\lambda)\|_2 = |\mu(\lambda)| \|\Delta x_l\|_2$ in terms of λ and $\mu(\lambda)$ this relation becomes

$$|\mu(\lambda)| \leqslant \tfrac{1}{2} \lambda \quad \forall \lambda \in [0, \lambda_l].$$

This feature of λ_l is also reflected by the minimizer-property of λ_l: Exactly up to λ_l the polynomial model of the relative change of $P_{N_l} J_l^{-1} F$ is strictly decreasing in $[0, 1]$. This means that

$$\delta_l := \lambda_l \|\Delta x_l\| \leqslant 1/\omega_l \tag{3.77}$$

is a reasonable choice to characterize the local trust region in an affine covariant trust region approach of type (2.8) where instead of the weight J_l^{-1} the factor $P_{N_l} J_l^{-1}$ is considered. Note that such an approach will still result in a damped Newton iteration. This follows directly from some basic geometrical arguments.

With the Newton path $\overline{x}_l$ at x_l we can argue in exactly the same way as in Remark 3.9 to obtain

$$1-\frac{1}{2} \leqslant \frac{\|P_{N_l}\cdot\big(\overline{x}_l(\lambda)-x_l\big)\|_2}{\lambda\|\Delta x_l\|_2}+\mathcal{O}(\lambda^2) \quad \text{and} \quad \frac{\|P_{N_l}\cdot\big(\overline{x}_l(\lambda)-x_l\big)\|_2}{\lambda\|\Delta x_l\|_2} \leqslant 1+\frac{1}{2}+\mathcal{O}(\lambda^2) \quad \forall \lambda\in(0,\lambda_l].$$

Let us denote the step size from Remark 3.9 by $\lambda_{l,(3.69)}$. This step size is based on the nonlinearity quantity $\omega_{(3.69)}$. Since ω_l is a local counterpart of $\omega_{(3.69)}$ and hence not larger, this implies that $\lambda_l \geqslant \lambda_{l,(3.69)}$. This means the above relation is likely to hold true for a wider range of step sizes.

Conditions for local quadratic convergence

Under the assumption that the sequence of iterates generated by the above algorithm is well defined and that the bound from (3.21) holds we have $\omega_l \leqslant 2\cdot\omega_{(3.21)}\ \forall l$. If there is an index $\underline{l}$ such that $2\cdot\omega_{(3.21)}\|\Delta x_{\underline{l}}\|_2 \leqslant 1$ then by means of the choice of λ_l in the above algorithm and by Theorem 3.11 the iteration turns into a full step method and eventually quadratic convergence to an x_* with $F(x_*)=0$ is achieved.

3.4.1.2 Computable estimates for ω_l and the simple monotonicity check

The above theoretical considerations motivate to establish a practical step size control which is based on estimations of the step sizes λ_l from Algorithm 3.3. To provide such an estimate for λ_l means to provide an approximation $[\omega]_l$ of ω_l. Since ω_l is related to the nonlinearity bound (3.73) an obvious choice is given by

$$[\omega]_l(\lambda) := 2\frac{\|P_{N_l}J_l^{-1}\big(F(x_l+\lambda\Delta x_l)-F_l-\lambda J_l\Delta x_l\big)\|_2}{\lambda^2\|\Delta x_l\|_2^2} \tag{3.78}$$

for some $\lambda\in\Lambda_l\setminus\{0\}$. Substituting I for P_{N_l} in the above definition we obtain an estimate $[\tilde{\omega}]_l(\lambda)$ for $\tilde{\omega}_l$ which evolves from ω_l by the same substitution process. This estimate is employed in the literature, e.g. [26, 11], as an estimate for several types of affine covariant Lipschitz constants. By providing a theoretical step size control which is based on nonlinearity bounds we fill a gap between classical Lipschitz constants and associated computationally available estimates as the following lemma shows.

Lemma 3.34 *Let ω_l and $[\omega]_l(\lambda)$ be given as in (3.73) and (3.78), respectively. Assume that the Lipschitz condition*

$$\|P_{N_l}J_l^{-1}\big(J(y)-J_l\big)(y-x_l)\|_2 \leqslant \Omega_l\|y-x_l\|_2^2 \tag{3.79}$$

for all $y\in\mathcal{D}$ with $J(y)=F'(y)$, $y-x_l=\lambda\Delta x_l$, $\lambda\in[0,1]$ holds. Then for $\lambda\in\Lambda_l$,

$$[\omega]_l(\lambda) \leqslant \omega_l \leqslant \Omega_l.$$

Such a relation is also true for quantities $[\tilde{\omega}]_l(\lambda)$, $\tilde{\omega}_l$ and $\tilde{\Omega}_l$ which evolve from the respective above stated non-tilde quantities by substituting the identity matrix I for the projector P_{N_l}.

Proof. The first inequality follows directly from the definitions of $[\omega]_l(\lambda)$ and ω_l. The second one is obtained by an argument similar to the one we used to prove Proposition 3.1. Arguing in the same manner also proves the stated relations for the tilde quantities.

■

To evaluate the estimate (3.78) we need an appropriate step size λ. We will exploit this estimate in a corrector step of a predictor-corrector-scheme to determine λ_l.

Assume that an initial guess, i.e. a predictor, $\lambda_{l,0}$ of λ_l from Algorithm 3.3 is given. We will discuss several choices of predictors in the next paragraph. These predictors are also based on an estimate for ω_l. It is not guaranteed that descent occurs and hence it has to be checked.

If no descent occurs we can use $\lambda_{l,0}$ to define a corrector step size by means of the estimate (3.78). Generally, if $\lambda_{l,j}$ fails to provide descent we define the corrector step size $\lambda_{l,j+1}$ via

$$\lambda_{l,j+1} := \frac{1}{[\omega]_l(\lambda_{l,j})\|\Delta x_l\|_2}. \tag{3.80}$$

We will see in Proposition 3.36 that $\lambda_{l,j+1} < \lambda_{l,j}$. The corrector is the unique minimizer of the polynomial model

$$p_l\big(s; [\omega]_l(\lambda)\big) := \big(1 - s + \tfrac{1}{2}[\omega]_l(\lambda)\|\Delta x_l\|_2 s^2\big)^2, \quad s \in [0,1], \tag{3.81}$$

for $\lambda = \lambda_{l,j}$. We check whether $\lambda_{l,j+1}$ provides descent and in case of failure compute a new corrector via (3.80). This procedure is summarized in Algorithm 3.4.

Algorithm 3.4 (Simple monotonicity check)

1: given: $\mathring{\Lambda}_{\mathcal{D}_l}$ as the set of inner points of $\Lambda_{\mathcal{D}_l}$ from Theorem 3.33, Λ_l from (3.72),
2: a predictor $\lambda_{l,0} \in \Lambda_l$
3: set $j = 0$
4: **if** $T(x_l + \lambda_{l,j}\Delta x_l | P_{N_l} J_l^{-1}) < T(x_l | P_{N_l} J_l^{-1})$ **then**
5: set $\lambda_l = \lambda_{l,j}$
6: **return**
7: **else** $\triangleright\ \lambda_{l,j} \notin \mathring{\Lambda}_{\mathcal{D}_l}$ $\dashrightarrow$ reduce step size
8: correct the step size by determining a new aspirant $\lambda_{l,j+1} < \lambda_{l,j}$ according to (3.80)
9: set $j = j + 1$ and **go to** line 4
10: **end if**

The corrector is cheaply available since $F(x_l + \lambda_{l,j}\Delta x_l)$ is already computed for the check in line 4 of Algorithm 3.4. To show that $\lambda_{l,j+1} < \lambda_{l,j}$ we need the following auxiliary lemma which is also of importance for the considerations in Subsection 3.4.2 where the restricted monotonicity concept will be discussed.

Lemma 3.35 *For one $\lambda \in (0,1]$ assume that $[\omega]_l(\lambda)$ from (3.78) is well defined and let $p_l\big(s; [\omega]_l(\lambda)\big)$ be defined as in (3.81). Then,*

$$T(x_l + \lambda\Delta x_l | P_{N_l} J_l^{-1}) \leqslant p_l\big(\lambda; [\omega]_l(\lambda)\big) \cdot T(x_l | P_{N_l} J_l^{-1}).$$

Proof. The statement follows directly from the relation (3.70), the definition of $[\omega]_l(\lambda)$ and the first estimate in (3.19).

■

Note that the above result does not necessarily imply descent since it may hold that $p\big(\lambda; [\omega]_l(\lambda)\big) > 1$.

Proposition 3.36 *Assume that* $\lambda_{l,j} \in \left(\Lambda_l \setminus \mathring{\Lambda}_{\mathcal{D}_l}\right) \neq \emptyset$ *which implies that the check in line 4 of Algorithm 3.4 returns false. Then for* $\lambda_{l,j+1}$ *defined via* (3.80) *it holds that*

$$\lambda_{l,j+1} \leqslant \tfrac{1}{2}\lambda_{l,j}. \tag{3.82}$$

Proof. First, we show that $q_{l,j}(\lambda) := p_l\big(\lambda; [\omega]_l(\lambda_{l,j})\big) - 1$ has exactly two roots in $[0,1]$. One root is obviously $\lambda = 0$. Since p_l is strictly convex on $[0,1]$ there can be at most one more root of $q_{l,j}(\lambda)$ in $[0,1]$. To argue by contradiction let us assume there is no second root. By the strict convexity of p_l and by $p_l'\big(0; [\omega]_l(\lambda_{l,j})\big) = -2$ the assumption of no second root implies that $0 \leqslant p\big(\lambda; [\omega]_l(\lambda_{l,j})\big) < 1$ $\forall \lambda \in (0,1]$. So in particular, $p\big(\lambda_{l,j}; [\omega]_l(\lambda_{l,j})\big) < 1$. But by Lemma 3.35 this implies that the check in line 4 of Algorithm 3.4 returns true. A contradiction to the assumptions. Hence, there is a second root of $q_{l,j}$ in $[0,1]$. From the definition of $\lambda_{l,j+1}$ it is readily seen that this quantity is the unique minimizer of p_l in $[0,1]$. A short calculation shows that the second root of $q_{l,j}$ is twice the unique minimizer of p_l in $[0,1]$, i.e., $q_{l,j}(2 \cdot \lambda_{l,j+1}) = 0$. By the convexity of p_l it holds that $p_l\big(\lambda; [\omega]_l(\lambda_{l,j})\big) < 1$ $\forall \lambda \in (0, 2 \cdot \lambda_{l,j+1})$. So any statement of the form $0 < \lambda_{l,j} < 2 \cdot \lambda_{l,j+1}$ would lead to a contradiction by the same arguments we used above. Therefore, (3.82) holds true.

■

This result implies that Algorithm 3.4 terminates after a finite number of steps. Since Δx_l is a direction of descent for the PNLF at $x_l \in \mathcal{D}$ the set $\mathring{\Lambda}_{\mathcal{D}_l}$ is not empty and the above reduction finally yields a step size $\lambda_{l,j} \in \mathring{\Lambda}_{\mathcal{D}_l}$ terminating the algorithm.

Remark 3.37 The relation (3.82) also holds true in the context of the NLF. Simply substitute $[\tilde{\omega}]_l$ from Lemma 3.34 for $[\omega]_l$ and the NLF for the PNLF in all instances of appearance. Hence, the above result shows that a correction strategy of the form

$$\lambda_{l,j+1} := \min\left(\frac{1}{[\tilde{\omega}]_l(\lambda_{l,j})\|\Delta x_l\|_2}, \tfrac{1}{2}\lambda_{l,j}\right)$$

which is applied in [26] and [11] just introduces unnecessary redundancy. □

Define

$$\overline{\Delta x}_{l_j+} := -J_l^{-1} F(x_l + \lambda_{l,j}\Delta x_l). \tag{3.83}$$

A short calculation shows that the check in line 4 of Algorithm 3.4 is equivalent to

$$|\Delta x_l^T \overline{\Delta x}_{l_j+}| < \|\Delta x_l\|_2^2. \tag{3.84}$$

We use this form of the check since it is cheaper to evaluate than the one stated in Algorithm 3.4.

To determine the corrector step size $\lambda_{l,j+1}$ we also make use of (3.83). By means of the definition of $[\omega]_l(\lambda)$ in (3.78) and the definition of the corrector in (3.80) we to obtain

$$\lambda_{l,j+1} = \frac{1}{2} \cdot \frac{\lambda_{l,j}^2 \|\Delta x_l\|_2^2}{\left|\Delta x_l^T\left(\overline{\Delta x}_{l_j+} - (1-\lambda_{l,j})\Delta x_l\right)\right|} = \frac{1}{2} \cdot \left[\left|\frac{\Delta x_l^T \overline{\Delta x}_{l_j+}}{\|\Delta x_l\|_2^2} - (1-\lambda_{l,j})\right|\right]^{-1} \cdot \lambda_{l,j}^2. \tag{3.85}$$

We prefer the second term for actual computing: Since the check (3.84) is already done $\Delta x_l^T \overline{\Delta x}_{l_j+}$ and $\|\Delta x_l\|_2^2$ are available for free. Hence, to determine the corrector only a complexity of $\mathcal{O}(1)$ floating point operations arises!

Remark 3.38 For a practical step size control in the context of the NLF in [26, 11] a corrector step size based on $[\tilde{\omega}]_l(\lambda)$ from Lemma 3.34 is defined. It is given via

$$\widetilde{\lambda}_{l,j+1} := \frac{1}{2} \cdot \frac{\lambda_{l,j}^2 \|\Delta x_l\|_2}{\|\overline{\Delta x}_{l_j+} - (1-\lambda_{l,j})\Delta x_l\|_2}.$$

Direct comparison shows that $\widetilde{\lambda}_{l,j+1} \leqslant \lambda_{l,j+1}$. So an enlargement of steps by introducing the projection is also true for the actual computed step sizes. In the context of the natural level function simple monotonicity is checked via

$$\|\overline{\Delta x}_{l_j+}\|_2 < \|\Delta x_l\|_2. \tag{3.86}$$

So Δx_l, $\|\Delta x_l\|_2$ and $\overline{\Delta x}_{l_j+}$ are at hand *but not* the denominator in the definition of the above corrector. Hence, the computational effort to determine $\widetilde{\lambda}_{l,j+1}$ is $\mathcal{O}(4n)$ floating point operations. Our corrector is therefore much cheaper to evaluate. However, to compare the two concepts in terms of the computational effort for one iteration step, we must also take the calculation of a predictor into account. This is investigated in the next paragraph. □

3.4.1.3 Evaluating a predictor step size

For an execution of Algorithm 3.4 a predictor step size $\lambda_{l,0}$ is required. To obtain a predictor only minor computational effort should arise. This means, the number of floating point operations should not exceed $\mathcal{O}(c \cdot n)$ for some constant $c \in \mathbb{N}$. This condition is met by exploiting already computed quantities from the previous step $l-1$ and the current correction Δx_l and its norm $\|\Delta x_l\|_2$, respectively. Note that making use of Δx_l and $\|\Delta x_l\|_2$ does not contradict the demand to provide a predictor in a cheap way since Δx_l must be computed anyway. This is also true for its norm which is necessary for the simple monotonicity check in (3.84). Additionally, it is employed in a termination criterion—see the next paragraph. Since $\|\Delta x_l\|_2$ is considered to be available the crucial point of a prediction step is to provide a reasonable estimate for ω_l. For the first step, $l = 0$, the user has to supply an initial step size.

We will present three methods to determine a predictor step size. The first two are adaptations of classical predictors whereas the third one will be introduced in this work. Additionally, we will present an analysis of the computational cost of the predictor-corrector-scheme for all three predictors. For comparison we will also state the costs of the NLF related step size control from [11].

Simple predictor.
A rather obvious choice of predictor is obtained by employing the latest corrector estimate from the previous step, i.e., $[\omega]_{l-1}(\lambda_{l-1})$,

$$\lambda_{l,0_1} := \min\left(1, \frac{1}{[\omega]_{l-1}(\lambda_{l-1})\|\Delta x_l\|_2}\right). \tag{3.87}$$

A predictor of this type, excluding the projectional aspect, is for example exploited in [6]. Since there no projection is considered in general our predictor provides a larger step size. From the

discussion in the previous paragraph about the computational costs of a corrector step size it follows that $[\omega]_{l-1}(\lambda_{l-1})$ is computable within a complexity of $\mathcal{O}(1)$ floating point operations. Since $\|\Delta x_l\|_2$ is known, the computational effort of determining the predictor (3.87) is also of order $\mathcal{O}(1)$. So it is very cheap to evaluate. However, this estimate is based on an estimation of the nonlinearity at x_{l-1}. The next two choices are associated with estimations of the nonlinearity at x_l, the current iterate.

Projected Deuflhard's predictor.
This predictor is an adaption of the one presented in [11]. To provide an estimation for the nonlinearity at the current iterate by means of the available data a *'backward looking'* concept is applied. This means that we deal with the nonlinearity of F at x_l along the direction pointing to the *previous* iterate x_{l-1}. To motivate this choice of predictor from a theoretical point of view we have to introduce a local Lipschitz condition. In the spirit of the prediction strategy in Chapter 3.3 of [11] let us assume that

$$\|P_{N_l} J_l^{-1}\big(J(y) - J_l\big)v\|_2 \leqslant \hat{\Omega}_l \|y - x_l\|_2 \|v\|_2 \tag{3.88}$$

holds for all $y - x_l = -\lambda \Delta x_{l-1}$, $\lambda \in [0, \lambda_{l-1}]$, and for v 'not too far away from' $y - x_l$.[2] This condition may be interpreted as a modified 'backward looking' counterpart of (3.79). Again following [11], we set $y = x_{l-1}$ and $v = \overline{\Delta x}_l$ where

$$\overline{\Delta x}_l := -J_{l-1}^{-1} F_l \tag{3.89}$$

and therefore obtain an estimate for $\hat{\Omega}_l$ via

$$\frac{\|P_{N_l} J_l^{-1}\big(J_l - J_{l-1}\big)\overline{\Delta x}_l\|_2}{\lambda_{l-1}\|\Delta x_{l-1}\|_2 \|\overline{\Delta x}_l\|_2} = \frac{\|P_{N_l}(\overline{\Delta x}_l - \Delta x_l)\|_2}{\lambda_{l-1}\|\Delta x_{l-1}\|_2 \|\overline{\Delta x}_l\|_2} =: \big[\hat{\Omega}\big]_l. \tag{3.90}$$

So we define $\lambda_{l,0_2}$ according to

$$\begin{aligned} \lambda_{l,0_2} := \min(1, \hat{\lambda}_{l,0_2}), \quad \hat{\lambda}_{l,0_2} &:= \frac{1}{\big[\hat{\Omega}\big]_l \|\Delta x_l\|_2} = \frac{\|\Delta x_{l-1}\|_2}{\|P_{N_l}(\overline{\Delta x}_l - \Delta x_l)\|_2} \cdot \frac{\|\overline{\Delta x}_l\|_2}{\|\Delta x_l\|_2} \cdot \lambda_{l-1} \\ &= \frac{\|\Delta x_{l-1}\|_2 \cdot \|\overline{\Delta x}_l\|_2}{|\Delta x_l^T(\overline{\Delta x}_l - \Delta x_l)|} \cdot \lambda_{l-1}. \end{aligned} \tag{3.91}$$

Substituting the identity matrix I for P_{N_l} we obtain the predictor $\widetilde{\lambda}_{l,0_2}$ from Chapter 3.3 of [11], i.e.,

$$\widetilde{\lambda}_{l,0_2} := \min\left[1, \frac{\|\Delta x_{l-1}\|_2}{\|\overline{\Delta x}_l - \Delta x_l\|_2} \cdot \frac{\|\overline{\Delta x}_l\|_2}{\|\Delta x_l\|_2} \cdot \lambda_{l-1}\right]. \tag{3.92}$$

It holds that $\lambda_{l,0_2} \geqslant \widetilde{\lambda}_{l,0_2}$. In order to compare the computational effort of determining the two predictors it is safe to neglect the costs to evaluate $\|\Delta x_i\|_2$, $i = l-1, l$. For our predictor we need to take the computation of $|\Delta x_l^T(\overline{\Delta x}_l - \Delta x_l)| = |\Delta x_l^T \overline{\Delta x}_l - \Delta x_l^T \Delta x_l|$ and $\|\overline{\Delta x}_l\|$ into account. Since $\Delta x_l^T \Delta x_l = \|\Delta x_l\|_2^2$ this information is essentially for free and hence the computational effort to determine it can also be discarded. Note that the vector $\overline{\Delta x}_l$ is available from the previous step, however, usually not its norm. It is only available if the termination criterion (3.100) was considered in the previous step—see the below Paragraph *Termination criterion.* In any case we

[2] The vague term 'not too far away from' is adopted from the respective discussion in Chapter 3.3 of [11].

also need to calculate the inner product $\Delta x_l^T \overline{\Delta x}_l$. Hence, a computational effort of $\mathcal{O}(4n)$ or $\mathcal{O}(2n)$, respectively, arises to determine $\lambda_{l,0_2}$. Considering $\widetilde{\lambda}_{l,0_2}$ only $\|\overline{\Delta x}_l - \Delta x_l\|_2$ needs some computational effort worth mentioning. Note that $\|\overline{\Delta x}_l\|_2$ is at hand from the previous step due to the check (3.86). Thus, $\widetilde{\lambda}_{l,0_2}$ can be computed with a complexity of only $\mathcal{O}(3n)$ floating point operations. However, we have to take the whole step into account. Both monotonicity tests (3.84) and (3.86) introduce the same amount of computational work but recall that our corrector is available essentially for free whereas the classical corrector needs $\mathcal{O}(4n)$ floating point operations.

The theoretical justification for this predictor is based on a Lipschitz condition. This does not fit very well in our overall context of nonlinearity bounds. We will overcome this disadvantage by the next choice of predictor.

Projected nonlinearity bound predictor.
This predictor is rather special. To the best of our knowledge there is no classical counterpart to it. Just like for the projected Deuflhard's predictor a 'backward looking' concept is employed. However, this concept is not related to a Lipschitz condition but to a nonlinearity bound, hence, it fits very well into our overall concept how to describe the nonlinearity of F. As a 'backward looking' counterpart to ω_l from (3.73) we assume that

$$2\|P_{N_l} J_l^{-1}\big(F(y) - F_l - J_l(y - x_l)\big)\|_2 \leqslant \hat{\omega}_l \|y - x_l\|_2^2 \tag{3.93}$$

holds for all y with $y - x_l = -\lambda \Delta x_{l-1}$, $\lambda \in [0, \lambda_{l-1}]$. This bound readily gives rise to the estimate

$$[\hat{\omega}]_l := 2\frac{\|P_{N_l} J_l^{-1}\big(F_{l-1} - F_l + \lambda_{l-1} J_l \Delta x_{l-1}\big)\|_2}{\lambda_{l-1}^2 \|\Delta x_{l-1}\|_2^2}. \tag{3.94}$$

At first glance this estimate introduces an unjustifiable amount of work since the product $J_l^{-1} F_{l-1}$ is not computed yet and takes $\mathcal{O}(n^2)$ floating point operations. A closer look reveals that due to the projection we are actually in need of the term $\Delta x_l^T J_l^{-1} F_{l-1}$ which we may compute via the

$$w_l\text{-strategy:} \qquad \text{I. solve} \quad J_l^T w_l = \Delta x_l, \qquad \text{II. do the multiplication} \quad w_l^T \cdot F_{l-1}. \tag{3.95}$$

Again, a linear system is to be solved. But the vector w_l can also be used for the evaluation of the check (3.84) and the corrector (3.85) because

$$\Delta x_l^T \overline{\Delta x}_{l_j+} = -w_l^T \cdot F(x_l + \lambda_{l,j} \Delta x_l). \tag{3.96}$$

This saves the computation of $\overline{\Delta x}_{l_j+}$, an $\mathcal{O}(n^2)$-operation which is executed $(1 + \#$ of correction steps)-times. Instead we only need to compute the above multiplication which is of order $\mathcal{O}(2n)$ for each evaluation. Considering the predictor we obtain

$$\begin{aligned} \lambda_{l,0_3} := \min(1, \hat{\lambda}_{l,0_3}), \quad \hat{\lambda}_{l,0_3} &:= \frac{1}{[\hat{\omega}]_l \|\Delta x_l\|_2} = \frac{1}{2} \cdot \frac{\lambda_{l-1}^2 \|\Delta x_{l-1}\|_2^2}{\|P_{N_l} J_l^{-1}\big(F_{l-1} - F_l + \lambda_{l-1} J_l \Delta x_{l-1}\big)\|_2 \cdot \|\Delta x_l\|_2} \\ &= \frac{1}{2} \cdot \frac{\lambda_{l-1}^2 \|\Delta x_{l-1}\|_2^2}{|w_l^T F_{l-1} + \|\Delta x_l\|_2^2 + \lambda_{l-1} \Delta x_l^T \Delta x_{l-1}|}. \end{aligned} \tag{3.97}$$

To determine this predictor $\mathcal{O}(4n)$ floating point operations are necessary since $w_l^T F_{l-1}$ and $\Delta x_l^T \Delta x_{l-1}$ are not known beforehand.

For this approach $\overline{\Delta x}_{l_j+}$ is never computed. We have to take this into account for our termination criterion—see the next paragraph. Also, some 'tweaking' ideas are not applicable. Refer to Paragraph 3.4.1.7 for details.

Remark 3.39 Since the predictor (3.87) is also not in need of $\|\overline{\Delta x}_{l_j+}\|_2$ we may apply the w_l-strategy (3.95) and (3.96) in the context of this projector too. □

As a summary of the computational costs of the predictor-corrector-scheme per iteration step based on which predictor is exploited we provide Table 3.1.

	# of floating point operations in $\mathcal{O}(\cdot)$					
	Δx_l	$\|\Delta x_l\|_2$	$\Delta x_l^T J_l^{-1}$ or $\overline{\Delta x}_{l,j}$	predictor	monotonicity check	corrector
simple	n^2	$2n$	$n^2 + (n^2)^*$ or $(\overline{j}+1)\cdot n^2$	1	$(\overline{j}+1)\cdot 2n$	$(\overline{j}+1)$
proj Dflh	n^2	$2n$	$(\overline{j}+1)\cdot n^2$	$4n$ or $(2n)^{**}$	$(\overline{j}+1)\cdot 2n$	$(\overline{j}+1)$
proj nonlin	n^2	$2n$	$n^2 + (n^2)^*$	$4n$	$(\overline{j}+1)\cdot 2n$	$(\overline{j}+1)$
Dflh (NLF)	n^2	$2n$	$(\overline{j}+1)\cdot n^2$	$3n$	$(\overline{j}+1)\cdot 2n$	$(\overline{j}+1)\cdot 4n$

Table 3.1: Computational costs of the predictor-corrector-scheme per iteration step for different predictor strategies. The first three rows are related to the PNLF (see above for detailed descriptions). The last row is related to the NLF and the step size control from Chapter 3.3 in [11]. It is stated for the purposes of comparison. The quantity $\overline{j}$ reflects the number of corrector steps. The term $(n^2)^*$ is related to the additional computational cost if the termination criterion (3.100) is considered in the current step. See the next paragraph for details. The term $(2n)^{**}$ refers to the reduced cost of the predictor if (3.100) was considered in the previous step.

3.4.1.4 Termination criterion

Our implementation of a termination criterion is inspired by the criteria in [26] and [11]. Let XTOL be some prescribed error tolerance. Then each time Δx_l is computed we check

$$\|\Delta x_l\|_2 \leqslant \text{XTOL}. \tag{3.98}$$

If this is true we terminate the iteration with the final estimate

$$x_{*,l+1} := x_l + \Delta x_l.$$

Otherwise we proceed with calculating a predictor. If the predictor $\lambda_{l,0}$ equals one and the monotonicity check of Algorithm 3.4 succeeds for the predictor step size we also check $\overline{\lambda}_{l,1} := \min(1, \lambda_{l,1})$ to be one. Additionally, we add the test

$$\|\Delta x_l\|_2 \leqslant \sqrt{10 \cdot \text{XTOL}} \tag{3.99}$$

from [26]. If the considered step sizes are one and the additional test is passed it gives good reason to hope that we reached the local contraction domain of Newton's method. The next iterate $x_{l+1} = x_l + \Delta x_l$ is readily available. However, to apply the error estimate (3.23) of the local

convergence Theorem 3.11 for the error of x_{l+1} we are in need of Δx_{l+1}. This correction is not available without further expensive computation. Assume that $\overline{\Delta x}_{l+1} = -J_l^{-1} F_{l+1}$ is available. In the spirit of (2.31) we use its norm as a substitute quantity for the norm of the error at x_{l+1}. So if

$$\lambda_{l,0} = \overline{\lambda}_{l,1} = 1$$

and (3.100)

$$\|\Delta x_l\|_2 \leqslant \sqrt{10 \cdot \mathrm{XTOL}} \qquad \text{and} \qquad \|\overline{\Delta x}_{l+1}\|_2 \leqslant \mathrm{XTOL}$$

we terminate the iteration with the final estimate

$$x_{*,l+1} := x_l + \overline{\Delta x}_{l+1}.$$

If the w_l-strategy, (3.95) and (3.96), is applied no $\overline{\Delta x}_{l+1}$ is determined yet. We have to compute it separately to be available for the termination criterion (3.100). This will be done only if already the other conditions of the termination criterion are fulfilled.

If we apply the w_l-strategy our numerical tests confirm that due to the above additional test the quantity $\overline{\Delta x}_{l+1}$ is computed only once per run of the algorithm. One still may argue that there is at least one step (the final one) where both w_l and $\overline{\Delta x}_{l+1}$ are computed. But recall that for the w_l-strategy every time we need to compute $\delta x_l^T \overline{\delta x}_{l_j+}$ that this is possible in $\mathcal{O}(2n)$ operations. So for every rejected step size in the course of the algorithm we save $\mathcal{O}(n^2)$ operations compared to a scenario where $\overline{\delta x}_{l_j+}$ is evaluated. This means that if there is at least one single rejected step size during the whole iteration the final additional cost of computing $\overline{\Delta x}_{l+1}$ is compensated. Note that often in the first iteration step of the algorithm there will be rejected step sizes may it be that the user provided initial guess $\lambda_{0,0}$ is too restrictive or too optimistic, respectively.

3.4.1.5 Increasing step sizes

Let $\lambda_{l,j} \in (0,1)$ fulfill the monotonicity check in line 4 of Algorithm 3.4 and let $\overline{\lambda}_{l,j+1} := \min(1, \lambda_{l,j+1})$. This second step size is the unique minimizer in $[0,1]$ of $p_l\big(\lambda; [\omega]_l(\lambda_{l,j})\big)$ from (3.81) and therefore the latest estimate of λ_l from Algorithm 3.3. If $\lambda_{l,j} < \overline{\lambda}_{l,j+1}$ this suggests that a further decrease of the level function can be expected for $\overline{\lambda}_{l,j+1}$ compared to $\lambda_{l,j}$. Also, a larger step size may result in a faster convergence to a solution. However, checking simple monotonicity for $\overline{\lambda}_{l,j+1}$ results in extra computational effort. So we opt for $\overline{\lambda}_{l,j+1}$ only if a noticeable increase of the step size is given. This means, we check for prescribed $\underline{\eta}$ with $0 < \underline{\eta} < 1$ if

$$\lambda_{l,j} \leqslant \underline{\eta} \cdot \overline{\lambda}_{l,j+1} \tag{3.101a}$$

holds. We add a second test. For a constant BADTOL with $0 < \mathrm{BADTOL} < 1$ we demand that $\overline{\lambda}_{l,j+1}$ fulfills

$$\overline{\lambda}_{l,j+1} \leqslant \mathrm{BADTOL} \cdot \lambda_{bad}. \tag{3.101b}$$

If both checks are passed we opt for increasing the step size by redoing the monotonicity check for $\overline{\lambda}_{l,j+1}$. The step size λ_{bad} is initialized each step by some value bigger than $1/\mathrm{BADTOL}$. Each time there is some $\lambda_{l,j}$ which fails the monotonicity check we set $\lambda_{bad} = \lambda_{l,j}$. The quantity BADTOL is a safety factor. It reflects the confidence we put into the larger step size $\overline{\lambda}_{l,j+1}$ to pass the monotonicity check.

3.4.1.6 Algorithmic illustration of the simple monotonicity concept

Summarizing all the above discussed aspects a basic step size control employing the simple monotonicity check, i.e. Algorithm 3.4, looks as follows:

Algorithm 3.5 (Step size control at iterate $x_l \in \mathcal{D}$, based on simple monotonicity)

1: given: $0 < \underline{\eta} < 1$, $0 < \text{XTOL} \ll 1$, $0 < \lambda_{min} \ll 1$, $\lambda_{user} \in (0, 1]$, valid$\lambda$ = false
2: $0 < \text{BADTOL} < 1$, $\lambda_{bad} > 1/\text{BADTOL}$
3: $x_l \in \mathcal{D}$
4: determine $F_l = F(x_l)$, $J_l = F'(x_l)$ and solve the linear system $J_l \Delta x_l = -F_l$
5: **if** $\|\Delta x_l\|_2 \leqslant \text{XTOL}$ **then** ▷ simple convergence test
6: set $x_{*,l+1} = x_l + \Delta x_l$ ▷ sufficient approximation for solution found
7: **terminate, solution found**
8: **else if** $l > 0$ **then** ▷ predictor step
9: choose $\lambda_{l,0}$ as $\lambda_{l,0} = \begin{cases} \lambda_{l,0_1} & \text{from (3.87) or} \\ \lambda_{l,0_2} & \text{from (3.91) or} \\ \lambda_{l,0_3} & \text{from (3.97)} \end{cases}$
10: **else**
11: set $\lambda_{l,0} = \lambda_{user}$
12: **end if**
13: set $\lambda_{l,0} = \max(\lambda_{l,0}, \lambda_{min})$ and $j = 0$
14: determine $F_{l_j+} = F(x_l + \lambda_{l,j}\Delta x_l)$ and $\Delta x_l^T \overline{\Delta x}_{l_j+} = \Delta x_l^T J_l^{-1} F_{l_j+}$ ▷ trial iterate quantities
15: determine corrector $\lambda_{l,j+1}$ from (3.85) and set $\lambda_{l,j+1} = \min(1, \lambda_{l,j+1})$
16: **if** $|\Delta x_l^T \overline{\Delta x}_{l_j+}| < \|\Delta x_l\|_2^2$ **then** ▷ simple monotonicity check
17: set validλ = true
18: **if** $j = 0$ && $\lambda_{l,0} = \lambda_{l,1} = 1$ **then** ▷ check for convergence
19: **if** $\|\Delta x_l\|_2 \leqslant \sqrt{10 \cdot \text{XTOL}}$ **then**
20: **if** $\lambda_{l,0} = \lambda_{l,0_3}$ **then** ▷ no $\overline{\Delta x}_{l_j+}$ computed yet
21: solve the linear system $J_l \overline{\Delta x}_{l_j+} = -F_{l_j+}$
22: **end if**
23: **if** $\|\overline{\Delta x}_{l_j+}\|_2 \leqslant \text{XTOL}$ **then**
24: set $x_{*,l+1} = x_l + \overline{\Delta x}_{l_j+}$ ▷ sufficient approximation for solution found
25: **terminate, solution found**
26: **else**
27: set $\lambda_l = 1$ and $x_{l+1} = x_l + \Delta x_l$
28: **invoke this algorithm for the next iterate** x_{l+1}
29: **end if**
30: **else**
31: set $\lambda_l = 1$ and $x_{l+1} = x_l + \Delta x_l$
32: **invoke this algorithm for the next iterate** x_{l+1}
33: **end if**
34: **else if** $\lambda_{l,j} \leqslant \underline{\eta} \cdot \lambda_{l,j+1}$ && $\lambda_{l,j+1} \leqslant \text{BADTOL} \cdot \lambda_{bad}$ **then** ▷ increase step size
35: set $j = j + 1$ **go to** line 14
36: **else** ▷ use current step size aspirant
37: set $\lambda_l = \lambda_{l,j}$ and $x_{l+1} = x_l + \lambda_l \Delta x_l$

38: **invoke this algorithm for the next iterate** x_{l+1}
39: **end if**
40: **else if** validλ **then** ▷ current step size fails, previous was valid, take it
41: set $\lambda_l = \lambda_{l,j-1}$ and $x_{l+1} = x_l + \lambda_l \Delta x_l$
42: **invoke this algorithm for the next iterate** x_{l+1}
43: **else if** $\lambda_{l,j} \leqslant \lambda_{min}$ **then** ▷ no descent for smallest step size
44: **stop iteration, abort algorithm** – convergence failure
45: **else** ▷ corrector step
46: set $\lambda_{bad} = \lambda_{l,j}$
47: set $\lambda_{l,j+1} = \max(\lambda_{l,j+1}, \lambda_{min})$ and $j = j + 1$ **go to** line 14
48: **end if**

Remark 3.40 For each λ_l determined by the above algorithm simple monotonicity is ensured. However, we cannot guarantee that this is also true for all $\tilde{\lambda}$ with $0 < \tilde{\lambda} < \lambda_l$. The numerical estimates of ω_l cannot provide this information. Also, we cannot ensure that a bound of the form (3.75) and therefore of the form (3.76) holds for all $\tilde{\lambda} \leqslant \lambda_l$. In the next subsection we will discuss an approach where at least for the taken step λ_l bounds of the kind (3.75) and (3.76) can be shown to hold true—see (3.116) and (3.117). □

Conditions for local quadratic convergence

Assume that the sequence of iterates defined by the above algorithm is well defined and that $x_l \neq x_* \ \forall l$ where x_* is a solution of $F(x) = 0$. Also assume that the bound (3.21) from the local convergence Theorem 3.11 is given. It holds that

$$\begin{matrix}[\omega]_l(\lambda_l) \\ [\hat{\omega}]_l\end{matrix} \leqslant \omega_l \leqslant 2 \cdot \omega_{(3.21)} \quad \forall l.$$

Recall that the simple predictor (3.87) depends on $[\omega]_l(\lambda_l)$. The nonlinearity bound predictor (3.97) depends on $[\hat{\omega}]_l$. The quantity ω_l is related to the step size $\overline{\lambda}_{\underline{l}}$ from Theorem 3.33 which we employed in the theoretical step size control in Algorithm 3.3.

If there is an index $\underline{l}$ with $2 \cdot \omega_{(3.21)} \|\Delta x_{\underline{l}}\|_2 \leqslant 1$ the predictors (3.87) and (3.97) are equal to one. The same is true for $\overline{\lambda}_{\underline{l}}$. This means that the predictor passes the monotonicity check in line 16 of the above algorithm and becomes the actual step size. By Theorem 3.11 the same results hold true for the indices $l > \underline{l}$. Hence, we obtain a full step Newton iteration and again by Theorem 3.11 eventually quadratic convergence of the iterates to a solution x_* of $F(x) = 0$ is ensured (if XTOL = 0) or the above algorithm terminates either in line 7 or 25 after a finite number of steps since also $\lambda_{l,1} = 1 \ \forall l \geqslant \underline{l}$. If the predictor (3.91) is employed and if there is a constant $\hat{\Omega} < \infty$ such that $\hat{\Omega}_l \leqslant \hat{\Omega} \ \forall l$ for the Lipschitz constants from (3.88) then we obtain the same results under the assumption that there is an index $\underline{l}$ such that $2 \cdot \max\big(\omega_{(3.21)}, \hat{\Omega}\big) \|\Delta x_{\underline{l}}\|_2 \leqslant 1$.

3.4.1.7 Extensions to Algorithm 3.5

Reducing the step size if $x_l + \lambda \Delta x \notin \mathcal{D}$

The above algorithm implicitly assumes that for all $\lambda_{l,j}$ it holds that $x_l + \lambda_{l,j} \Delta x \in \mathcal{D}$, i.e., $\lambda_{l,j} \in \Lambda_l$. This is not always true. E.g., a negative argument for an evaluation of $\log(\cdot)$ or $\sqrt{\cdot}$ may arise. In case of $\lambda_{l,j} \notin \Lambda_l$ the following algorithm may be exploited

Algorithm 3.6 (Step size reduction due to $\lambda_{l,j} \notin \Lambda_l$)

given: $\lambda_{l,j} \notin \Lambda_l$ with Λ_l from (3.72)
  $0 < \text{red\_fac} < 1$
if validλ **then**
  set $\lambda_l = \lambda_{l,j-1}$ and $x_{l+1} = x_l + \lambda_l \Delta x_l$
  return
else
  while $\lambda_{l,j} \notin \Lambda_l$ **do**
    if $\lambda_{l,j} \leqslant \lambda_{min}$ **then**
      stop iteration, abort algorithm, convergence failure
    end if
    set $\lambda_{bad} = \lambda_{l,j}$
    set $\lambda_{l,j} = \text{red\_fac} \cdot \lambda_{l,j}$
  end while
end if

This algorithm terminates after a finite number of steps since $x_l \in \mathcal{D}$ which is by assumption an open set.

Employing unprojected predictors

If at step l the intermediate correction $\overline{\Delta x}_l$ is available and if the predictor is chosen via the simple predictor (3.87) or the projected Deuflhard's predictor (3.91) then in case of a failed monotonicity check for the predictor we may exploit the respective *unprojected* smaller counterpart, i.e., (3.87) with $[\bar{\omega}]_{l-1}(\lambda_{l-1})$ instead of $[\omega]_{l-1}(\lambda_{l-1})$ or (3.92), respectively, for a second check. However, these are only considered if they are not too close to the failed projected predictor. More precisely, the next step size will be $\lambda_{l,1} = \max(\lambda_{l,1}, \lambda^{unproj} \cdot \text{PRED\_BADTOL})$ where λ^{unproj} denotes the unprojected predictor and PRED_BADTOL a constant in $(0, 1)$. Such a tweaking strategy is taken into account in our algorithmic realization of the above Algorithm 3.5.

An unprojected predictor is not always cheaply available, like it is the case for the nonlinearity bound predictor (3.97) since $J_l^{-1} F_{l-1}$ is unknown and costly to evaluate. In this situation and if the projected predictor provides a too optimistic step size we choose the next step size to be $\lambda_{l,1} = \max(\lambda_{l,1}, \lambda_{l,0} \cdot \text{PRED\_RED})$ for some constant PRED_RED with $0 < \text{PRED\_RED} < 1$.

3.4.2 Restricted monotonicity

Recall from Lemma 3.34 that for the computable estimate $[\omega]_l(\lambda)$ from (3.78) and the theoretical nonlinearity bound ω_l from (3.73) it holds that

$$[\omega]_l(\lambda) \leqslant \omega_l, \quad \lambda \in \Lambda_l.$$

Thorough investigation of (3.70) reveals that there is some theoretical framework which provides a closer relationship to the estimates $[\omega]_l$. This framework relies on the step size dependent affine

covariant projected nonlinearity bound

$$\omega_l(\lambda) := \sup_{s\in(0,\lambda]} 2\frac{\|P_{N_l}J_l^{-1}\big(F(x_l+s\Delta x_l)-F_l-sJ_l\Delta x_l\big)\|_2}{s^2\|\Delta x_l\|_2^2}. \tag{3.102}$$

In this subsection we will use this bound to define for $\eta\in(0,2)$ specific step sizes $\lambda_{l_m}(\eta)$ which in turn define a polynomial model for the relative change of the PNLF. Therefore, we will call these step sizes *modeling step sizes*. By means of these polynomial models we will see that descent is ensured for each λ in the range of $(0,\lambda_{l_m}(\eta)]$. Such step sizes are also employed for the step size strategies in [5] and [6].[3] However, those step sizes are based on step size dependent *Lipschitz conditions*. We will compare our modeling step sizes to the ones from the literature in terms of magnitude. Furthermore, we will discuss the influence of η. As it will turn out it is reasonable for a practical realization to choose λ_l as an estimate of the modeling step size $\lambda_{l_m}(1)$. The step size algorithm is based on the same corrector-predictor-scheme as the step size control from the previous subsection. However, we will combine it with a more demanding monotonicity check to provide a reasonable estimate for $\lambda_{l_m}(1)$. This *restricted monotonicity check* is an adaption of the check proposed in [6] to the context of projected nonlinearity quantities. We will give an algorithmic outline of this check and show that only a finite number of corrections are necessary to provide a step that passes the check.

3.4.2.1 Theoretical framework

Right from the definition of the quantities $[\omega]_l(\lambda)$, $\omega_l(\lambda)$ and ω_l it follows that

$$[\omega]_l(\lambda)\leqslant\omega_l(\lambda)\leqslant\omega_l \quad\text{and}\quad \sup_{\lambda\in\Lambda_l}\omega_l(\lambda)=\omega_l. \tag{3.103}$$

The computable estimate $[\omega]_l(\lambda)$ and the theoretical bound $\omega_l(\lambda)$ are in close relationship as the following result shows.

Lemma 3.41 *For given $\lambda\in(0,1]$ assume that the theoretical bound $\omega_l(\lambda)$ and the computable estimate $[\omega]_l(\lambda)$ from* (3.102) *and* (3.78)*, respectively, are well defined. Also, let F be three-times continuously differentiable on $\mathcal{D}$. Then, it holds that*

$$[\omega]_l(\lambda)=\omega_l(\lambda)+\mathcal{O}(\lambda).$$

This means that especially in the case where λ is very small the approximation is of high quality.

Proof. Three cases are to be considered:

- Both quantities are equal. And since $0\in\mathcal{O}(\lambda)$ the stated inequality holds.
- It holds that $[\omega]_l(\lambda)<\omega_l(\lambda)$ and $\omega_l(\lambda)$ is determined by some $\tilde{s}$ where $\lambda>\tilde{s}>0$. Then we have by the triangle inequality

$$\begin{aligned}&\omega_l(\lambda)-[\omega]_l(\lambda)\\ &\leqslant \frac{\Big\|P_{N_l}J_l^{-1}\Big(\tfrac{1}{3}F'''(x_l)[\Delta x_l]^3\tilde{s}+o(\tilde{s})-\big(\tfrac{1}{3}F'''(x_l)[\Delta x_l]^3\lambda+o(\lambda)\big)\Big)\Big\|_2}{\|\Delta x_l\|_2^2}\\ &\leqslant \frac{1}{3}\frac{\|P_{N_l}J_l^{-1}F'''(x_l)[\Delta x_l]^3\|_2}{\|\Delta x_l\|_2^2}(\lambda-\tilde{s})+o(\lambda)\in\mathcal{O}(\lambda).\end{aligned}$$

[3] though they are not explicitly termed as *modeling step sizes*

- It holds that $[\omega]_l(\lambda) < \omega_l(\lambda)$ but $\omega_l(\lambda)$ is determined via

$$\omega_l(\lambda) = \lim_{s\to 0} 2\frac{\|P_{N_l} J_l^{-1}\big(F(x_l + s\Delta x_l) - F(x_l) - sF'(x_l)\Delta x_l\big)\|_2}{s^2\|\Delta x_l\|_2^2}.$$

  By the assumed smoothness of F this means that

$$\omega_l(\lambda) = \frac{\|P_{N_l} J_l^{-1} F''(x_l)[\Delta x_l]^2\|_2}{\|\Delta x_l\|_2^2}$$

  which yields

$$\begin{aligned}\omega_l(\lambda) - [\omega]_l(\lambda) &\leqslant \frac{\|P_{N_l} J_l^{-1}\big(\frac{1}{3}F'''(x_l)[\Delta x_l]^3\lambda + o(\lambda)\big)\|_2}{\|\Delta x_l\|_2^2} \\ &\leqslant \frac{1}{3}\frac{\|P_{N_l} J_l^{-1} F'''(x_l)[\Delta x_l]^3\|_2}{\|\Delta x_l\|_2^2}\lambda + o(\lambda) \in \mathcal{O}(\lambda).\end{aligned}$$

  This concludes the proof.

■

Next we will employ the bound (3.102) to define and to prove the existence of the modeling step sizes $\lambda_{l_m}(\eta)$ and to show that descent holds for all $\lambda \in (0, \lambda_{l_m}(\eta)]$. The following theorem is an adaption of a result from [5] to the context of projected nonlinearity quantities. Compared to [5] the proof is slightly more elaborated.

Theorem 3.42 *Let F fulfill Assumption 2.1 and the iterate x_l Assumption 3.32. Let Λ_l be defined as in* (3.72)*. Then,*

$$T(x_l + \lambda\Delta x_l | P_{N_l} J_l^{-1}) \leqslant \big(1 - \lambda + \tfrac{1}{2}\omega_l(\lambda)\|\Delta x_l\|_2\lambda^2\big)^2 T(x_l | P_{N_l} J_l^{-1}) \quad \forall \lambda \in \Lambda_l. \tag{3.104}$$

If there is a $\bar{\omega} < \infty$ such that for the step size dependent nonlinearity bound (3.102) *it holds that*

$$\omega_l(\lambda) \leqslant \bar{\omega} \quad \forall \lambda \in \Lambda_l$$

then there exist for η with $0 < \eta < 2$ step sizes $\lambda_{l_m} = \lambda_{l_m}(\eta)$ which fulfill

$$\lambda_{l_m} = \min\left(1, \frac{\eta}{\omega_l(\lambda_{l_m})\|\Delta x_l\|_2}\right). \tag{3.105}$$

For all $\lambda \in (0, \lambda_{l_m}]$ we have

$$T(x_l + \lambda\Delta x_l | P_{N_l} J_l^{-1}) \leqslant \big(1 - \lambda + \tfrac{1}{2}\omega_l(\lambda_{l_m})\|\Delta x_l\|_2\lambda^2\big)^2 T(x_l | P_{N_l} J_l^{-1}) \tag{3.106}$$

and

$$1 - \lambda + \tfrac{1}{2}\omega_l(\lambda_{l_m})\|\Delta x_l\|_2\lambda^2 \leqslant 1 - \lambda(1 - \eta/2) < 1. \tag{3.107}$$

Thus, descent holds for these step sizes, i.e.,

$$T(x_l + \lambda\Delta x_l | P_{N_l} J_l^{-1}) < T(x_l | P_{N_l} J_l^{-1}) \quad \forall \lambda \in (0, \lambda_{l_m}]. \tag{3.108}$$

Proof. For ease of writing we define

$$X_l := \{x \in \mathcal{D} \mid x = x_l + \lambda \Delta x_l,\, \lambda \in \Lambda_l\}.$$

Since $\mathcal{D}$ is open and convex and $x_l \in \mathcal{D}$, Λ_l and X_l are not empty and with $\tilde{\lambda} \in \Lambda_l$ it holds for all λ with $0 < \lambda \leqslant \tilde{\lambda}$ that $\lambda \in \Lambda_l$.

The relation (3.104) directly follows from (3.70) and (3.19) with $\omega_l(\lambda)$ substituted for $\omega_{(3.18)}$.

Now we turn to the step sizes λ_{l_m}. For an arbitrary but fixed $\eta \in (0, 2)$ we consider the function

$$a : \begin{cases} \Lambda_l \to \mathbb{R}_+ \\ \lambda \mapsto \dfrac{\eta}{\omega_l(\lambda)\|\Delta x_l\|_2}. \end{cases}$$

By definition of $\omega_l(\lambda)$ this function is monotonically nonincreasing and it holds that

$$a(\lambda) \geqslant \eta/(\bar{\omega}\|\Delta x_l\|_2) > 0 \quad \forall \lambda \in \Lambda_l. \tag{3.109}$$

Hence, two cases may occur. In the first case, there is a $\lambda_{l_m} \in \Lambda_l$ with $a(\lambda_{l_m}) = \lambda_{l_m}$. Since the identity map id_{Λ_l} is strictly monotonically increasing there is no other $\lambda \in \Lambda_l$ with this property. In the second case, no such λ_{l_m} exists. This already implies that there is no $\lambda \in \Lambda_l$ with $\lambda > a(\lambda)$ because of (3.109). We will show by contradiction that the second case implies that $1 \in \Lambda_l$. So let us assume that for each $\lambda \in \Lambda_l$ it holds that $a(\lambda) > \lambda$ and $1 \notin \Lambda_l$. From $a(\lambda) > \lambda$ it follows that

$$1 - \lambda + \tfrac{1}{2}\omega(\lambda)\|\Delta x_l\|_2 \lambda^2 < 1 - \lambda(1 - \eta/2) < 1$$

which means that $X_l \subset \mathcal{D}_l$ with $\mathcal{D}_l$ as defined in Assumption 3.32. Since $\overline{\mathcal{D}}_l \subset \mathcal{D}$ and $1 \notin \Lambda_l$ is assumed there exists $\hat{\lambda} \in (0, 1)$ such that $x_l + \hat{\lambda}\Delta x_l \in \mathcal{D} \setminus \overline{\mathcal{D}}_l$. By the definition of Λ_l we also have $\hat{\lambda} \in \Lambda_l$ and hence $x_l + \hat{\lambda}\Delta x_l \in X_l$. But since $\left(\mathcal{D} \setminus \overline{\mathcal{D}}_l\right) \cap X_l = \emptyset$ this is a contradiction. Thus, $1 \in \Lambda_l$ and $a(1) > 1$ which yields $\lambda_{l_m} = 1$.

Let λ_{l_m} be determined according to one of the two above discussed cases. The polynomial estimate in (3.106) is true because $\omega_l(\lambda)$ is monotonically nondecreasing. The first inequality in (3.107) follows from $a(\lambda_{l_m}) \geqslant \lambda$ for all $\lambda \in (0, \lambda_{l_m}]$. The second inequality is a direct consequence of $\lambda \in (0, 1]$ and $\eta \in (0, 2)$. Finally, (3.108) follows from combining (3.104) and (3.107).

■

Comparison of various modeling step sizes

Here we compare our modeling step sizes to modeling step sizes which are also defined according to (3.105) but where $\omega_l(\lambda)$ is substituted by some different nonlinearity quantities. Note that these quantities and the associated modeling step sizes are exclusively defined for the analysis in this paragraph. The reader does not have to keep these values at the back of his mind for the subsequent paragraphs.

We define:

$$\begin{aligned}\breve{\Omega}_l(\lambda) &:= \sup_{s\in(0,\lambda]} \frac{\|J_l^{-1}(J(x_l+s\Delta x_l)-J_l)\|_2}{s\|\Delta x_l\|_2}\\ \tilde{\Omega}_l(\lambda) &:= \sup_{s\in(0,\lambda]} \frac{\|J_l^{-1}\big(J(x_l+s\Delta x_l)-J_l\big)\Delta x_l\|_2}{s\|\Delta x_l\|_2^2},\end{aligned} \tag{3.110}$$

$\Omega_l(\lambda)$ via substituting J_l^{-1} by $P_{N_l}J_l^{-1}$ in the definition of $\tilde{\Omega}_l$,

$\tilde{\omega}_l(\lambda)$ by substituting the identity matrix$I$ for P_{N_l} in the definition of $\omega_l(\lambda)$

and for $\eta\in(0,2)$ associated modeling step sizes $\lambda_{i,l_m}(\eta)$, $i\in\{\breve{\Omega},\tilde{\Omega},\Omega,\tilde{\omega}\}$, according to (3.105). The Lipschitz quantity $\breve{\Omega}$ is employed in [6]. $\tilde{\Omega}$ is defined in accordance to the Lipschitz quantity in [5].

Lemma 3.43 *Assume that for $\lambda\in(0,1]$ the quantity $\breve{\Omega}_l(\lambda)$ from* (3.110) *is well defined. Then, $\omega_l(\lambda)$ from* (3.102) *as well as the remaining quantities in* (3.110) *are well defined too and the estimates*

$$\omega_l(\lambda)\leqslant \begin{matrix}\tilde{\omega}_l(\lambda)\\ \Omega_l(\lambda)\end{matrix} \leqslant \tilde{\Omega}_l(\lambda)\leqslant \breve{\Omega}_l(\lambda)$$

hold which imply for the modeling step sizes that

$$\lambda_{l_m}(\eta)\geqslant \begin{matrix}\lambda_{\tilde{\omega},l_m}(\eta)\\ \lambda_{\Omega,l_m}(\eta)\end{matrix} \geqslant \lambda_{\tilde{\Omega},l_m}(\eta)\geqslant \lambda_{\breve{\Omega},l_m}(\eta)\quad \forall\eta\in(0,2).$$

Proof. Due to the submultiplicativity of the Euclidean norm $\tilde{\Omega}_l(\lambda)$ is well defined and $\tilde{\Omega}_l(\lambda)\leqslant\breve{\Omega}_l(\lambda)$.

Analogously to (3.5) we have for $s\in(0,\lambda]$,

$$\begin{aligned}2\frac{\|J_l^{-1}\big(F(x_l+s\Delta x_l)-F_l-sJ_l\Delta x_l\big)\|_2}{s^2\|\Delta x_l\|_2^2} &= 2\frac{\|\int_0^s J_l^{-1}\big(J(x_l+t\Delta x_l)-J_l\big)\Delta x_l dt\|_2}{s^2\|\Delta x_l\|_2^2}\\ \leqslant 2\int_0^s \frac{\|J_l^{-1}\big(J(x_l+t\Delta x_l)-J_l\big)\Delta x_l\|_2}{s^2\|\Delta x_l\|_2^2}dt &\leqslant 2\int_0^s \frac{t}{s^2}\tilde{\Omega}_l(s)dt=\tilde{\Omega}_l(s).\end{aligned} \tag{3.111}$$

Hence, $\tilde{\omega}_l(\lambda)$ is well defined and $\tilde{\omega}_l(\lambda)\leqslant\tilde{\Omega}_l(\lambda)$.

Since $\|P_{N_l}z\|_2\leqslant\|z\|_2$ for arbitrary $z\in\mathbb{R}^n$, it is readily seen that $\omega_l(\lambda)$ and $\Omega_l(\lambda)$ are well defined and also that $\omega_l(\lambda)\leqslant\tilde{\omega}_l(\lambda)$ and $\Omega_l(\lambda)\leqslant\tilde{\Omega}_l(\lambda)$ holds. Substituting J_l^{-1} by $P_{N_l}J_l^{-1}$ and hence $\tilde{\Omega}_l(s)$ by $\Omega_l(s)$ in (3.111) finally leads to $\omega_l(\lambda)\leqslant\Omega_l(\lambda)$.

The inequalities w.r.t to the modeling step sizes are direct consequences of the above shown inequalities for the Ω- and ω-quantities.

■

So by the above lemma and (3.103) we obtain in terms of projected bounds the relation

$$[\omega]_l(\lambda)\leqslant\omega_l(\lambda)\leqslant\Omega_l(\lambda).$$

An analogous result is true for the tilde quantities. Thus, by our choice of bounds we may not only describe the local nonlinearity of F in a reasonable way as it is seen in Theorem 3.42 but also provide closer relationship to the computable estimates.

Choice of η

We motivate the choice of $\eta = 1$ from the following perspectives:

Polynomial extremal property

For $\eta \in (0, 2)$ we consider the polynomial model

$$1 - \lambda + \tfrac{1}{2}\omega_l\big(\lambda_{l_m}(\eta)\big)\|\Delta x_l\|_2\lambda^2, \quad \lambda \in [0, \lambda_{l_m}(\eta)], \tag{3.112}$$

from (3.106). Its unique minimizer $\lambda^{min}(\eta)$ in $[0, \lambda_{l_m}(\eta)]$ is given via

$$\lambda^{min}(\eta) := \begin{cases} \lambda_{l_m}(\eta) & \text{for } \eta \in (0, 1], \\ \min\big(1, 1/\omega_l\big(\lambda_{l_m}(\eta)\big)\|\Delta x_l\|_2\big) & \text{for } \eta \in (1, 2). \end{cases}$$

Therefore, and since $\omega_l(\lambda)$ is nondecreasing and continuous we obtain

$$\lambda^{min}(1) = \lambda_{l_m}(1) \geqslant \lambda^{min}(\eta) \quad \forall \eta \in (0, 2).$$

So in terms of the magnitude of the minimizers of the polynomial models the step size $\lambda_{l_m}(1)$ provides an optimal choice. A result of this kind is also given in [5, 6]. However, there it is argued based on Lipschitz conditions instead of (projected) nonlinearity bounds.

Regarding the affine covariant trust region approach (2.8), again with $P_{N_l}J_l^{-1}$ substituted for J_l^{-1}, the extremal property of $\lambda_{l_m}(1)$ makes $\lambda_{l_m}(1)\|\Delta x_l\|_2 \leqslant 1/\omega_l\big(\lambda_{l_m}(1)\big)$ a natural choice for δ_l giving a refinement of (3.77).

Bounded influence of the local nonlinearity

In analogy to (3.75) we obtain by the definition of $\omega_l(\lambda)$ in (3.102) and for given $\eta \in (0, 2)$ and $\lambda \in (0, \lambda_{l_m}(\eta)]$ the result

$$\begin{aligned} \frac{\|P_{N_l}\chi_l(\lambda)\|_2}{\lambda\|\Delta x_l\|_2} &= \frac{\|P_{N_l}J_l^{-1}\big(F(x_l + \lambda\Delta x_l) - F_l - \lambda J_l\Delta x_l\big)\|_2}{\lambda\|\Delta x_l\|_2} \\ &\leqslant \tfrac{1}{2}\lambda\omega_l(\lambda)\|\Delta x_l\|_2 \leqslant \frac{\eta}{2} \end{aligned} \tag{3.113}$$

which yields

$$1 - \frac{\eta}{2} \leqslant \frac{\|P_{N_l}J_l^{-1}F(x_l + \lambda\Delta x_l) - P_{N_l}J_l^{-1}F_l\|_2}{\lambda\|\Delta x_l\|_2} \leqslant 1 + \frac{\eta}{2} \tag{3.114}$$

instead of (3.76). Hence, up to $\lambda_{l_m}(1)$ the polynomial model (3.112) guarantees that the influence of the nonlinearity is restricted to the second binary digit. The statements about $\mu(\lambda)$ and the relation to the Newton path given on page 49 and 50 also hold true with $\lambda_{l_m}(1)$ substituted for λ_l.

Conditions for local quadratic convergence

If the bound (3.21) from the local convergence Theorem 3.11 is given then we can choose in Theorem 3.42 the constant $\bar{\omega}$ as $\bar{\omega} := 2 \cdot \omega_{(3.21)}$ and thus provide a uniformly bound for $\omega_l(\lambda)$. Consider a step size control like Algorithm 3.3 where step sizes are defined via $\lambda_{l_m}(1)$ instead of $\overline{\lambda}_l$. If there is an index $\underline{l}$ such that $\bar{\omega}\|\Delta x_{\underline{l}}\|_2 \leqslant 1$ then $\lambda_{\underline{l}} = 1$ and from Theorem 3.11 it follows that $\lambda_l = 1 \forall l > \underline{l}$. Also, Theorem 3.11 guarantees eventually quadratic convergence to a solution x_* of $F(x) = 0$.

3.4.2.2 Step size control based on restricted monotonicity

In order to identify a computable substitute for $\lambda_{l_m}(1)$ we state the following result for the relative change of the PNLF.

Proposition 3.44 *For one $\lambda \in (0,1]$ assume that $[\omega]_l(\lambda)$ from (3.78) is well defined and let $p_l\big(s;[\omega]_l(\lambda)\big)$ be given according to (3.81).*

I) If $\lambda \in I_\lambda$ where

$$I_\lambda := (0, 2/[\omega]_l(\lambda)\|\Delta x_l\|_2)$$

then

$$p_l\big(\lambda;[\omega]_l(\lambda)\big) < 1$$

which implies that

$$T(x_l + \lambda\Delta x_l | P_{N_l} J_l^{-1}) < T(x_l | P_{N_l} J_l^{-1}).$$

II) For $\underline{\eta}$, $\overline{\eta}$ with $0 < \underline{\eta} \leqslant \overline{\eta} < 2$ let

$$\Lambda_l\big(\underline{\eta},\overline{\eta}\big) := \{\lambda \in (0,1] \mid \lambda = \tilde{\eta}/[\omega]_l(\lambda)\|\Delta x_l\|_2,\ \ \tilde{\eta} \in [\underline{\eta},\overline{\eta}]\}.$$

If $\lambda \in \Lambda_l\big(\underline{\eta},\overline{\eta}\big)$ then $\lambda \in I_\lambda$.

Assume that for all $\lambda \in (0,1]$ it holds that $\omega_l(\lambda)$ from (3.102) is uniformly bounded by some constant $\bar{\omega} < \infty$. Then, $\Lambda_l\big(\underline{\eta},\overline{\eta}\big) = \emptyset$ implies that $\lambda \in \big(0, \underline{\eta}/[\omega]_l(\lambda)\|\Delta x_l\|_2\big) \subset I_\lambda\ \forall \lambda \in (0,1]$.

Proof. I) Substituting one λ of λ^2 in $p_l\big(\lambda;[\omega]_l(\lambda)\big)$ by the upper bound $2/[\omega]_l(\lambda)\|\Delta x_l\|_2$ verifies the first estimate. The second one is a direct consequence of Lemma 3.35 and the first estimate.

II) The first implication follows from the definition of I_λ and $\Lambda_l\big(\underline{\eta},\overline{\eta}\big)$. Regarding the second implication we use a contradiction argument. $\lambda \geqslant \underline{\eta}/[\omega]_l(\lambda)\|\Delta x_l\|_2$ would either contradict the assumption $\Lambda_l\big(\underline{\eta},\overline{\eta}\big) = \emptyset$ or the fact that $\omega_l(\lambda) \leqslant \bar{\omega} < \infty\ \forall \lambda \in (0,1]$.

■

Motivated by this result we choose as an approximation for $\lambda_{l_m}(1)$ a step size λ which fulfills for $0 < \underline{\eta} < 1 < \overline{\eta} < 2$ the *restricted monotonicity check*

$$\Big(\lambda \in \Lambda_l(\underline{\eta},\overline{\eta})\Big) \qquad \text{or} \qquad \Big(\lambda = 1 \quad \text{and} \quad \lambda < \underline{\eta}/[\omega]_l(\lambda)\|\Delta x_l\|_2\Big). \tag{3.115}$$

A basic scheme how to determine a step size which passes the above check is given by Algorithm 3.7. This algorithm terminates after a finite number of steps if the conditions of paragraph II) from Proposition 3.44 are given: By line 22 for the next aspirant $\lambda_{l,j+1}$ it holds that

$$\lambda_{l,j+1} < \frac{1}{2} \cdot \left(1 + \frac{\eta}{\overline{\eta}}\right) \lambda_{l,j} \quad \text{if} \quad \eta_j > \overline{\eta}.$$

So reduction of the step size by a factor bounded away from zero is ensured. On the other hand, if the minimum in line 22 is not given by 1 we have

$$\lambda_{l,j+1} > \frac{1}{2} \cdot \left(1 + \frac{\overline{\eta}}{\underline{\eta}}\right) \lambda_{l,j} \quad \text{if} \quad \eta_j < \underline{\eta}$$

which is an increase in step size also in terms of a factor bounded away from zero. Such a factor may be arbitrarily small if the minimum is given by 1. But an increase is only considered if $\eta_j < \underline{\eta}$

and $\lambda_{l,j} < 1$. If in the next iteration step $\lambda_{l,j} = 1$ does not pass the check (3.115) then line 19 is executed because

$$\eta_{j-1} < \underline{\eta} \quad \text{and} \quad \eta_j > \overline{\eta}.$$

Generally, line 19 is executed if the above condition is true or this condition with swaped indices $j-1$ and j. In either way we use a method of Regula Falsi type like King's method, [18], to compute an approximation to a root of the function

$$g(\lambda) := \tfrac{1}{2}(\underline{\eta} + \overline{\eta}) - \lambda \cdot [\omega]_l(\lambda)\|\Delta x_l\|_2, \quad \lambda \in \mathcal{D}_g,$$

where $\mathcal{D}_g := [\min(\lambda_{l,j-1}, \lambda_{l,j}), \max(\lambda_{l,j-1}, \lambda_{l,j})]$. In line 19 it holds that

$$g(\lambda_{l,j-1}) < 0 \quad \text{and} \quad g(\lambda_{l,j}) > 0$$

or vice versa. Since g is continuous in λ either way King's method will converge superlinearly to a root of g in $\mathcal{D}_g$. We terminate the internal iteration if a λ with $\lambda \in \Lambda_l(\underline{\eta}, \overline{\eta})$ is found. Since $\underline{\eta} < \overline{\eta}$ only a finite number of internal steps is necessary. Summarizing, under the conditions of paragraph II) from Proposition 3.44 Algorithm 3.7 finds a step size λ_l which passes (3.115) in a finite number of steps.

Algorithm 3.7 (Restricted monotonicity check at x_l which fulfills Assumption 3.32)

1: given: predictor $\lambda_{l,0} \in (0,1]$, $0 < \underline{\eta} < \overline{\eta} < 2$, left = right = false
2: set $j = 0$
3: **while** true **do**
4: determine $\eta_j = \lambda_{l,j} \cdot [\omega]_l(\lambda_{l,j})\|\Delta x_l\|_2$
5: **if** $\eta_j \in [\underline{\eta}, \overline{\eta}]$ **then**
6: set $\lambda_l = \lambda_{l,j}$
7: **return**
8: **else if** $\eta_j < \underline{\eta}$ **then**
9: **if** $\lambda_{l,j} = 1$ **then**
10: set $\lambda_l = \lambda_{l,j}$
11: **return**
12: **else**
13: set left = true
14: **end if**
15: **else** ▷ i.e. $\eta_j > \overline{\eta}$
16: set right = true
17: **end if**
18: **if** left && right **then**
19: determine λ_l via a method of Regula Falsi type like King's method
20: **return**
21: **else**
22: set $\lambda_{l,j+1} = \min\left(1, \tfrac{1}{2}(\underline{\eta} + \overline{\eta})/[\omega]_l(\lambda_{l,j})\|\Delta x_l\|_2\right)$
23: set $j = j + 1$
24: **end if**
25: **end while**

Remark 3.45 The check (3.115) is clearly a more demanding condition than just asking for simple monotonicity. So usually the step sizes λ_l obtained from Algorithm 3.7 are smaller than or equal to the ones provided by Algorithm 3.5. This is in contrast to the theoretical quantities $\overline{\lambda}_l$ from Theorem 3.33 and the modeling step size $\lambda_{l_m}(1)$ from Theorem 3.42 since there we have

$$\overline{\lambda}_l \leqslant \lambda_{l_m}(1).$$

This discrepancy appears due to the fact that step sizes provided by Algorithm 3.5 are solely guaranteed to pass the simple monotonicity check. Other amiable properties of $\overline{\lambda}_l$ are not taken into account—see Remark 3.40. Let λ_l be defined via Algorithm 3.7. We can guarantee descent for λ_l but again not for $\tilde{\lambda}$ with $0 < \tilde{\lambda} < \lambda_l$.

However, a positive statement regarding the bounds (3.113) and (3.114) is possible: Since λ_l passes the restricted monotonicity test there is an $\hat{\eta} > 0$ with $\hat{\eta} \in [\underline{\eta}, \overline{\eta}]$ or $\hat{\eta} < \underline{\eta}$ such that $\lambda_l = \hat{\eta}/[\omega]_l(\lambda_l)\|\Delta x_l\|_2)$. Hence, for λ_l we obtain

$$\frac{\|P_{N_l}\chi_l(\lambda_l)\|_2}{\lambda_l\|\Delta x_l\|_2} = \tfrac{1}{2}\lambda_l[\omega]_l(\lambda_l)\|\Delta x_l\| = \frac{\hat{\eta}}{2} \tag{3.116}$$

and therefore

$$1 - \frac{\hat{\eta}}{2} \leqslant \frac{\|P_{N_l}J_l^{-1}F(x_l + \lambda_l\Delta x_l) - P_{N_l}J_l^{-1}F_l\|_2}{\lambda_l\|\Delta x_l\|_2} \leqslant 1 + \frac{\hat{\eta}}{2}. \tag{3.117}$$

Though not for all $\tilde{\lambda}$ with $0 < \tilde{\lambda} < \lambda_l$ bounds of the types (3.113) and (3.114) can be guaranteed *it is at least the case for the actual step size* λ_l. This is indeed an improvement compared to the step sizes provided by Algorithm 3.5. So the restricted monotonicity test is to be preferred in case of extremely nonlinear problems. □

Remarks on an algorithmic realization

We may embed Algorithm 3.7 like Algorithm 3.4 in a step size control like it is given via Algorithm 3.5. An adaption of the letter algorithm is straightforward, so we omit details but give some remarks instead. According to the restricted monotonicity test (3.115) we consider $\underline{\eta}$, $\overline{\eta}$ with $0 < \underline{\eta} < 1 < \overline{\eta} < 2$.

- To incorporate a minimum step size $0 < \lambda_{min} \ll 1$ we need to add after the computation of the corrector in line 22 the statement $\lambda_{l,j+1} = \max(\lambda_{l,j+1}, \lambda_{min})$. The iteration is aborted if $\eta_j > \overline{\eta}$ for $\lambda_{l,j} \leqslant \lambda_{min}$, i.e., if $\lambda_{l,j}$ does not pass the restricted monotonicity check and if it is considered to be too small to provide a reasonable advance towards a solution.

- If there is a $\lambda_{l,j} \notin \Lambda_l$ we can employ a scheme like Algorithm 3.6 to provide a step size in Λ_l. The flag validλ is set to true each time it holds that $\eta_j < \underline{\eta}$. In such a case for the corrector $\lambda_{l,j+1}$ it holds that $\lambda_{l,j+1} > \lambda_{l,j}$. We have to check that the corrector does not provide a step size bigger than BADTOL $\cdot\, \lambda_{bad}$. If that was the case we would opt for the current $\lambda_{l,j}$ to become λ_l. Though this step size does not pass the restricted monotonicity check it provides at least descent and also (3.116) and (3.117) are true for an $\hat{\eta} < \underline{\eta}$.

- As in the case of simple monotonicity for the step size $\lambda_{l,0}$ we can choose one of the predictors (3.87), (3.91) or (3.97). Note that for the predictors (3.87) and (3.97) we still can apply the

concept of computing w_l from (3.95) instead of $\overline{\Delta x}_{l_j+}$ since for the restricted monotonicity check we compute the main components of the corrector step size which in turn can be determined by means of w_l, cf. (3.85) and (3.96). In the context of projected quantities the computational costs stated in Table 3.1 do not change, though due to the close relationship between the check and the corrector, cf. line 4 and 22 of Algorithm 3.7, it is reasonable to merge their computational efforts. Note that for the last row in Table 3.1 the computational cost for both checking (restricted) monotonicity and calculating the corrector is reduced to $\mathcal{O}\big((\overline{j}+1)\cdot 4n\big)$. However, the predictor is more expensive due to the need to calculate the Euclidean norm of $\overline{\Delta x}_l$ which is of complexity $\mathcal{O}(2n)$.

- The termination criteria established in the previous subsection are applicable in the same way though we substitute the condition $\lambda_{l,0} = \overline{\lambda}_{l,1} = 1$ in (3.100) by

$$\lambda_{l,0} = 1 \quad \text{and} \quad \eta_0 \leqslant \tfrac{1}{2}(\underline{\eta} + \overline{\eta}).$$

  Although line 22 is never executed if the above is true it would also yield a step size equal to one. Hence, we have an equivalent step length condition to the one from (3.100) at hand.

- To determine step sizes we use the same computable estimates $[\omega]_l(\lambda)$, $[\hat{\omega}]_l$ and $[\hat{\Omega}]_l$ as in the previous subsection. So we can argue in a similar way as in Paragraph *Conditions for local quadratic convergence* on page 59 to ensure quadratic convergence of the iterates to a solution x_* of $F(x) = 0$: Let the sequence of iterates $\{x_l\}$ be well defined and assume that $x_l \neq x_* \ \forall l$. If there is an index $\underline{l}$ such that

$$2 \cdot \max\big(\omega_{(3.21)}, \hat{\Omega}\big)\|\Delta x_{\underline{l}}\|_2 \leqslant 1$$

  the predictor $\lambda_{\underline{l},0}$ is equal to one and the restricted monotonicity check is passed since either $\eta_0 < \underline{\eta}$ or $\underline{\eta} \leqslant \eta_0 \leqslant 1$ due to our choice $\underline{\eta} < 1$. Hence, $\lambda_{\underline{l}} = 1$. By means of Theorem 3.11 this is also true for all $l > \underline{l}$. And again by this theorem quadratic convergence is ensured.

3.4.3 Scaling invariance

Recall from Remark 2.2 that scaling invariance is a desirable property of an algorithm, this means that componentwise rescaling of variables should leave the performance of the algorithm invariant. Since we work in an affine covariant framework scaling generally refers to the domain space of F, for a particular exception see the Paragraph *Scaling of the linear systems* below. First we will give some basic notes about scaling. These considerations are along the lines of the respective discussion in [26]. We will also reconsider the computational costs of determining predictors in the light of scaling and we will discuss the impact of scaling on the w_l-strategy from (3.95).

Consider a change of units in the domain space which may be characterized by

$$y := S^{-1}x \qquad \text{with} \quad S := \operatorname{diag}(s_{11}, \ldots, s_{nn}),\ s_{ii} \neq 0,\ i = 1, \ldots, n.$$

Let $\hat{x} \in \mathcal{D}$ with $\hat{x}_{(i)} \neq 0$, $i = 1, \ldots, n$, and $D_{\hat{x}} := \operatorname{diag}(\hat{x})$. If we consider the *relative* quantities $x^{rel} := D_{\hat{x}}^{-1}x$ and accordingly $y^{rel} := D_{\hat{y}}^{-1}y$ where $\hat{y} := S^{-1}\hat{x}$ we have

$$y^{rel} = D_{\hat{y}}^{-1}y = D_{\hat{x}}^{-1}SS^{-1}x = D_{\hat{x}}^{-1}x = x^{rel}.$$

Let $G(y) := F(Sy)$. With the transformed systems

$$F^{rel}(x^{rel}) := F(D_{\hat{x}}x^{rel}) \qquad \text{and} \qquad G^{rel}(y^{rel}) := G(D_{\hat{y}}y^{rel}) = G(y)$$

we obtain

$$\big[(F^{rel})'(x^{rel})\big]^{-1} = D_{\hat{x}}^{-1}F'(x)^{-1} \qquad \text{and} \qquad \big[(G^{rel})'(y^{rel})\big]^{-1} = D_{\hat{y}}^{-1}G'(y)^{-1}.$$

Furthermore, since $G'(y) = F'(x)S$ and $D_{\hat{y}} = S^{-1}D_{\hat{x}}$,

$$G'(y)D_{\hat{y}} = F'(x)SS^{-1}D_{\hat{x}} = F'(x)D_{\hat{x}}$$

or equivalently (3.118)

$$D_{\hat{y}}^{-1}G'(y)^{-1} = D_{\hat{x}}^{-1}SS^{-1}F'(x)^{-1} = D_{\hat{x}}^{-1}F'(x)^{-1}.$$

Thus,

$$\big[(F^{rel})'(x^{rel})\big]^{-1} = \big[(G^{rel})'(y^{rel})\big]^{-1}.$$

Also, $F^{rel}(x^{rel}) = F(x)$ and $G^{rel}(y^{rel}) = F(x)$. Hence, at an iteration index l we obtain for the corresponding Newton and intermediate corrections

$$\begin{aligned}\Delta x_l^{rel} &= D_{\hat{x}}^{-1}\Delta x_l = D_{\hat{y}}^{-1}\Delta y_l = \Delta y_l^{rel}\\ \overline{\Delta x}_{l_j+}^{rel} &= D_{\hat{x}}^{-1}\overline{\Delta x}_{l_j+} = D_{\hat{y}}^{-1}\overline{\Delta y}_{l_j+} = \overline{\Delta y}_{l_j+}^{rel}.\end{aligned}$$

This means that by switching to relative quantities we ensure scaling invariance of our step size controls discussed in the above Subsections 3.4.1 and 3.4.2. Therefore, an associated damped iteration (3.67) also features this invariance property. However, note that the change from absolute to relative quantities, which is just a specific type of scaling, *does* change the behavior of the algorithm: Instead of scalar quantities like $\|\Delta x_l\|_2$, $\Delta x_l^T\overline{\Delta x}_{l_j+}$ we employ $\|D_{\hat{x}}^{-1}\Delta x_l\|_2$ and $\big(D_{\hat{x}}^{-1}\Delta x_l\big)^T D_{\hat{x}}^{-1}\overline{\Delta x}_{l_j+}$ which by no means have to have the same values.

The question arises how to choose $\hat{x}$. Note that components of $\hat{x}$ which are close to zero may cause overflow if $D_{\hat{x}}^{-1}$ is applied. So we will need some threshold value thrsh > 0 which we will substitute for critical components of $\hat{x}$. This certainly will destroy scaling invariance but will be inevitable in order to ensure numerical stability.

User provided scaling. An $\hat{x}^{user} \in \mathcal{D}$ provided by the user is a convenient choice. But usually this choice requires a user with experience and a good knowledge of the underlying problem. Formally, we obtain as a substitute for $D_{\hat{x}}$,

$$D_{user} := \operatorname{diag}(d_1, \ldots, d_n) \quad \text{where} \quad d_i := \max\big(|\hat{x}_{(i)}^{user}|, \text{thrsh}\big).$$

Error-oriented scaling. Thinking in terms of error estimates and termination criteria a theoretical optimal choice would be $\hat{x} = x_*$ leading to

$$D_* := \operatorname{diag}(d_1, \ldots, d_n) \quad \text{where} \quad d_i := \max\big(|x_{*,(i)}|, \text{thrsh}\big).$$

By this choice the termination criteria from (3.98) and (3.100), i.e.,

$$\|\Delta x_l\|_2 \leqslant \mathrm{XTOL}$$
$$\|\Delta x_l\|_2 \leqslant \sqrt{10 \cdot \mathrm{XTOL}} \quad \text{and} \quad \|\overline{\Delta x}_l\|_2 \leqslant \mathrm{XTOL}$$

turn into

$$\|D_*^{-1}\Delta x_l\|_2 \leqslant \mathrm{XTOL}$$
$$\|D_*^{-1}\Delta x_l\|_2 \leqslant \sqrt{10 \cdot \mathrm{XTOL}} \quad \text{and} \quad \|D_*^{-1}\overline{\Delta x}_l\|_2 \leqslant \mathrm{XTOL}$$

giving us – apart from the applied threshold concept – estimates of a *componentwise relative* error instead of estimates for the absolute error. Hence, we do not need to adjust the value of XTOL to the order of magnitude of the current problem. We may choose a *problem independent* small value, e.g. 10^{-6}, for it. Unfortunately, D_* is usually not available. But we may construct approximations for it iteratively via the the next scaling concept.

Adaptive scaling. To approximate D_* from the above paragraph at iteration step l we may exploit our latest iterate x_l assuming this is our best approximation of a solution so far. Since the predictor step sizes in the above stated step size controls rely on quantities from the previous step we do not only take x_l but also x_{l-1} into account. We opt for the 'Solomonic' choice from [26] defining for $l > 0$,

$$D_l := \mathrm{diag}(d_1, \ldots, d_n) \quad \text{where} \quad d_i := \max\Big[\tfrac{1}{2}\big(|x_{l-1,(i)}| + |x_{l,(i)}|\big), \mathrm{thrsh}\Big]. \tag{3.119a}$$

If $l = 0$ there is no previous step so we use

$$D_0 := \mathrm{diag}(d_1, \ldots, d_n) \quad \text{where} \quad d_i := \max(|x_{0,(i)}|, \mathrm{thrsh}). \tag{3.119b}$$

Note that adaptive scaling introduces the need of proper *rescaling* in the course of determining predictor step sizes. This is due to the fact that from the previous step quantities like $D_{l-1}^{-1}\Delta x_{l-1}$ are at hand but to fit into the current scaling scheme $D_l^{-1}\Delta x_{l-1}$ must be available. This means that additional computational effort arises. Hence, we consider a reevaluation of the predictors from Subsection 3.4.1 in terms of floating point operations. In the course of this we assume that $D_l^{-1}\Delta x_l$ and its norm as well as a properly scaled version of w_l^T from (3.95), i.e.,

$$(w_l^{sc})^T := (D_l^{-1}\Delta x_l)^T D_l^{-1} F'(x_l)^{-1} \tag{3.120}$$

(where necessary) are already available—see the next paragraph for details about how to obtain these quantities.

- *simple predictor:*
  The predictor (3.87) becomes

  $$\lambda_{l,0_1}^{sc} := \min\left(1, \frac{1}{[\omega]_{l-1}(\lambda_{l-1})\|D_l^{-1}\Delta x_l\|_2}\right).$$

  So it is still available with a complexity of order $\mathcal{O}(1)$.

- *Deuflhard predictors:*
  For the predictor (3.91) we obtain

$$\lambda_{l,0_2}^{sc} := \min(1, \hat{\lambda}_{l,0_2}^{sc}), \quad \hat{\lambda}_{l,0_2}^{sc} = \frac{\|D_l^{-1}\Delta x_{l-1}\|_2 \cdot \|D_l^{-1}\overline{\Delta x}_l\|_2}{|(D_l^{-1}\Delta x_l)^T(D_l^{-1}\overline{\Delta x}_l - D_l^{-1}\Delta x_l)|} \cdot \lambda_{l-1}$$

  and for (3.92),

$$\widetilde{\lambda}_{l,0_2}^{sc} := \min\left[1, \frac{\|D_l^{-1}\Delta x_{l-1}\|_2}{\|D_l^{-1}\overline{\Delta x}_l - D_l^{-1}\Delta x_l\|_2} \cdot \frac{\|D_l^{-1}\overline{\Delta x}_l\|_2}{\|D_l^{-1}\Delta x_l\|_2} \cdot \lambda_{l-1}\right].$$

  We are in need of the rescaling matrix $D_l^R := D_{l-1}^{-1}D_l$ which is available in $\mathcal{O}(n)$. Rescaling applies to Δx_{l-1} and $\overline{\Delta x}_l$. Hence, the respective norms need to be reevaluated. For $\lambda_{l,0_2}^{sc}$ also $(D_l^{-1}\Delta x_l)^T D_l^{-1}\overline{\Delta x}_l$ is not at hand. Considering $\widetilde{\lambda}_{l,0_2}^{sc}$ the norm $\|D_l^{-1}\overline{\Delta x}_l - D_l^{-1}\Delta x_l\|_2$ must be computed instead. This gives a complexity of $\mathcal{O}(9n)$ for $\lambda_{l,0_2}^{sc}$ and a complexity of $\mathcal{O}(10n)$ for $\widetilde{\lambda}_{l,0_2}^{sc}$.

- *Projected nonlinearity bound predictor:*
  In this case the adaption is given by

$$\lambda_{l,0_3}^{sc} := \min(1, \hat{\lambda}_{l,0_3}^{sc}), \quad \hat{\lambda}_{l,0_3}^{sc} = \frac{1}{2} \cdot \frac{\lambda_{l-1}^2\|D_l^{-1}\Delta x_{l-1}\|_2^2}{|(w_l^{sc})^T F_{l-1} + \|D_l^{-1}\Delta x_l\|_2^2 + \lambda_{l-1}(D_l^{-1}\Delta x_l)^T D_l^{-1}\Delta x_{l-1}|}.$$

  Considerable computational effort arises by determining D_l^R, applying it to $D_{l-1}^{-1}\Delta x_{l-1}$, computing the respective norm and by calculating $(w_l^{sc})^T F_{l-1}$ and $(D_l^{-1}\Delta x_l)^T D_l^{-1}\Delta x_{l-1}$. Therefore, a complexity of order $\mathcal{O}(8n)$ arises.

As a summary we provide Table 3.2.

# of floating point operations in $\mathcal{O}(\cdot)$			
simple	proj Dflh	Dflh (NLF)	proj nonlin
1	9n	10n	8n

Table 3.2: Computational effort to determine predictor step sizes in the context of adaptive scaling. Costs for evaluating $D_l^{-1}\Delta x_l$, its norm and w_l^{sc} are excluded.

Scaling of the linear systems. To solve the system

$$J_l\Delta x_l = -F_l, \qquad F_l := F(x_l),\ J_l := F'(x_l),$$

we assume that direct methods are applicable, i.e., the Jacobian J_l or a scaled version of it, respectively, is decomposed in a reasonable way (QR, LU). In order to ensure scaling invariance of the linear system solution, cf. (3.118), the above system is to be read as

$$\left(J_l D_{\hat{x}}\right)\left(D_{\hat{x}}^{-1}\Delta x_l\right) = -F_l.$$

Following an idea from [26], see also [11], we introduce a diagonal scaling matrix $D_{\hat{F}}$ from the left to achieve scaling invariance of the linear system solver w.r.t. componentwise (re-)scaling of F. So with

$$J_l^{sc} := D_{\hat{F}} J_l D_{\hat{x}}$$

the fully scaled system reads as follows

$$J_l^{sc}\big(D_{\hat{x}}^{-1}\Delta x_l\big) = -D_{\hat{F}}F_l. \tag{3.121}$$

A proper choice of $D_{\hat{F}}$ may be given by the user. An *adaptive* choice is provided in [26]: Let m_{ij}, $i,j = 1,\ldots,n$, be the elements of the matrix $J_l D_{\hat{x}}$. Then we substitute D_{F_l} for $D_{\hat{F}}$ with

$$D_{F_l} := \operatorname{diag}(d_1,\ldots,d_n) \quad \text{where} \quad d_i := \left[\max_{1\leqslant j\leqslant n}|m_{ij}|\right]^{-1}. \tag{3.122}$$

It is readily seen that this way scaling invariance w.r.t. componentwise scaling in the range of F is ensured. Also,

$$\|J_l^{sc}\|_1 \leqslant n.$$

Hence, this bound is also true for the Euclidean norm of J_l^{sc}. This may be exploited to cheaply estimate the condition of J_l^{sc} via

$$\operatorname{cond}_2^{est}(J_l^{sc}) := n \cdot \|D_{\hat{x}}^{-1}\Delta x_l\|_2 / \|D_{\hat{F}}F_l\|_2. \tag{3.123}$$

Recall that we do not want to proceed with the iteration in case of ill-conditioned Jacobians. So we may abort the the whole algorithm if $\operatorname{cond}_2^{est}(J_l^{sc}) > \text{CONDTOL}$ for some prescribed bound CONDTOL.

Remark 3.46 Let J_l^{sc} be nonsingular but possibly ill-conditioned and let a QR-decomposition of J_l^{sc} with column pivoting via $J_l^{sc}P = QR$ be given. Let the permutation matrix P be chosen such that $|r_{11}| \geqslant \ldots, \geqslant |r_{nn}|$ where r_{ii}, $i = 1,\ldots,n$, are the diagonal elements of the upper triangular matrix R. It is an easy task to show that for the so-called *subcondition* number $\operatorname{sc}(J_l^{sc}) := |r_{11}|/|r_{nn}|$ it holds that $\operatorname{sc}(J_l^{sc}) \leqslant \operatorname{cond}_2(J_l^{sc})$. Therefore,

$$\operatorname{sc}(J_l^{sc}) > \text{CONDTOL} \quad \Longrightarrow \quad \operatorname{cond}_2(J_l^{sc}) > \text{CONDTOL}.$$

Such an implication cannot be derived for the estimator (3.123). So if there is a QR-decomposition of J_l^{sc} of the above described type at hand the concept of subcondition numbers to monitor ill-conditioning of J_l^{sc} is to be preferred. □

Via (3.121) the vector $D_{\hat{x}}^{-1}\Delta x_l$ is computed. Since this correction is already scaled we can directly exploit it for the computation of the predictor and the termination criterion. However, recall that $F^{rel}(x^{rel}) = F(x)$. So we need to undo the scaling of Δx_l to compute $F(x_l + \lambda_{l,j}\Delta x_l)$ which is either needed for the system

$$J_l^{sc} D_{\hat{x}}^{-1}\overline{\Delta x}_{l_j+} = -D_{\hat{F}}F(x_l + \lambda_{l,j}\Delta x_l) \tag{3.124}$$

or for a scaled version of the w_l-strategy (3.95) and (3.96). We generalize the definition of w_l^{sc} from (3.120) to $w_l^{sc} := (D_{\hat{x}}^{-1}J_l^{-1})^T D_{\hat{x}}^{-1}\Delta x_l$ in order to cover all three types of domain space scaling discussed above. To obtain w_l^{sc} by means of J_l^{sc} the following steps are necessary:

$$\text{I. solve} \quad (J_l^{sc})^T(D_{\hat{F}}^{-1}w_l^{sc}) = D_{\hat{x}}^{-1}\Delta x_l \qquad \text{II. undo the scaling} \quad D_{\hat{F}}(D_{\hat{F}}^{-1}w_l^{sc}) = w_l^{sc}.$$

The scaled version of the product (3.96) is easily available via substituting w_l^{sc} for w_l. No scaling of $F(x_l + \lambda_{l,j}\Delta x_l)$ is necessary.

Remark 3.47 Every time the system (3.124) is solved, and this is at least once per step, $F(x_l + \lambda_{l,j}\Delta x_l)$ needs to be scaled. On the contrary, the unscaling step in the above evaluation of w_l^{sc} is *always* only necessary once each step. □

Chapter 4

Approximate Projected Natural Level Function

In this chapter we will transport the elaborated concept of the projected natural level function (PNLF) to a context where the Jacobian (and a decomposition of it) is not directly at hand. Instead we assume that products of the form $w^T \cdot F'(x)$ and $F'(x) \cdot d$ are available. These products are efficiently computable by *Automatic Differentiation*-techniques—see [15] for details about this concept. By means of an approximation to the current Jacobian we will define the *approximate projected natural level function* (APNLF) and a correction δx which is a direction of descent w.r.t. the APNLF. We will also provide sufficient conditions which ensure that the APNLF behaves like the PNLF in the direction of δx. It is due to the projectional aspect of the two level functions that we will be able to state these conditions in terms of angles between δx and the transposed gradient of the APNLF and between δx and the Newton correction. For the latter angle we will provide a computable estimate. In this way we will obtain an easy to handle monitor for the quality of the current approximation to the Jacobian.

We will give shape to our approximation idea via employing quasi-Newton techniques: We will introduce particular rank-1 matrices to improve the quality of the current Jacobian approximation which we will call *purifying updates*. Additionally, we will provide an update whose associated correction fulfills the descent property w.r.t. to the APNLF. We will call it *descent update*. By means of the descent update local superlinear convergence of a sequence of associated iterates to a solution of $F(x)$ will be shown.

We will adapt the step size controls from Section 3.4 and combine them with the above stated updates and the aforementioned angle monitors to construct a damped quasi-Newton iteration. By means of the step size controls from Section 3.4 the globalization approach based on the PNLF is affine covariant. This will also be true for the quasi-Newton approach if the initial Jacobian approximation A_0 is *affine covariant compatible*. We call a matrix A *affine covariant compatible* if $F \to MF$ for nonsingular M implies that $A \to MA$.

Our affine covariant globalization approach in the context of quasi-Newton methods is an alternative to Schlenkrich's globalization approach from [28]. His quasi-Newton updates are related to affine contravariance and to determine step sizes he employs the classical affine contravariant level function $T(x|I) = \frac{1}{2}\|F(x)\|_2^2$.

4.1 Basic Approximation Idea

We consider a function F which fulfills Assumption 2.1 and an $x \in \mathcal{D}$ with $F(x) \neq 0$ and $F'(x)$ nonsingular. Let $\overline{H} \in \mathbb{R}^{n\times n}$ be an approximation for $F'(x)^{-1}$ such that

$$\overline{\delta x} := -\overline{H}F(x) \tag{4.1}$$

is available. Also, assume that we can compute

$$\overline{\delta x}^T\overline{H} \qquad \text{and} \qquad \overline{H}F'(x)\overline{\delta x}. \tag{4.2}$$

In the following we will define an approximation to the PNLF in terms of such a matrix $\overline{H}$ and provide a corresponding direction of descent δx. As stated in the introduction of this chapter we will call this new level function *approximate projected natural level function* which we will abbreviate by the term *APNLF*.

We will see that the approximation quality of the APNLF and of δx w.r.t. the PNLF and the Newton correction Δx at x, respectively, can be monitored by means of the angle between δx and the transposed negative gradient at x of the APNLF and the angle between δx and Δx. Without knowledge of Δx the angle between δx and Δx is usually unknown. But we will provide an estimate for it.

Two concepts are of crucial importance for our approach to be viable and computationally efficient.

- *Projection:*
  Dropping the indices, recall from (3.70) that for the relative change of the PNLF we have

  $$\frac{T(x+\lambda\Delta x|P_NF'(x)^{-1})}{T(x|P_NF'(x)^{-1})} = \big(1-\lambda+\mu(\lambda)\big)^2 \tag{4.3}$$

  with

  $$\mu(\lambda) = -\frac{\Delta x^T}{\|\Delta x\|_2^2}F'(x)^{-1}\big(F(x+\lambda\Delta x) - F(x) - \lambda F'(x)\Delta x\big). \tag{4.4}$$

  One will see from Theorem 4.5 below that the relative change of the APNLF in the direction of δx is a *structural* analogy to (4.3). This analogy will turn out to be fundamental to justify the aforementioned angles as an approximation monitor—see Theorem 4.7 and Corollary 4.8.

- *Automatic Differentiation (AD):*
  As we will see in the course of providing the above mentioned angles or estimates of angles, respectively, we will need *adjoint* and *direct tangent* evaluations, i.e.,

  $$w^T \cdot F'(x) \qquad \text{and} \qquad F'(x)\cdot d, \qquad w,\, d \in \mathbb{R}^n.$$

  Applying AD-techniques is an efficient way to obtain these values without calculating the whole Jacobian. Adjoint terms are computable via the *reverse mode* of AD and direct tangents via the *forward mode* of AD, [15]. So we assume in the following that these modes of AD are available and hence the above products.

Remark 4.1 We would like to emphasize that the concept of the APNLF with angle monitors can be combined with any techniques which aim to provide a better approximation $\overline{H}$ to $F'(x)^{-1}$ as long as adjoint and direct tangent evaluations as well as the products (4.1) and (4.2) are available. It is not mandatory to employ quasi-Newton techniques like we will do it in this work to provide better approximations to the Jacobian. □

In terms of the approximation $\overline{H}$ to the inverse of the Jacobian $F'(x)$ and in terms of $\overline{\delta x}$ from (4.1) we define the APNLF in the following way.

Definition 4.2 (Approximate projected natural level function) *Suppose F fulfills Assumption 2.1. Let $x_l \in \mathcal{D}$ with $F(x_l) \neq 0$. Furthermore, let $\overline{H}_l \in \mathbb{R}^{n\times n}$ and assume that $\overline{\delta x}_l$ defined according to* (4.1) *is nonzero. Then for*

$$P_l := \frac{\overline{\delta x}_l \overline{\delta x}_l^T}{\overline{\delta x}_l^T \overline{\delta x}_l}$$

we call

$$T(x|P_l\overline{H}_l) = \frac{1}{2}\|P_l\overline{H}_l F(x)\|_2^2$$

the approximate projected natural level function *(at x_l) or abbreviated* APNLF.

According to (2.12) we know that the Newton correction at x_l is a direction of descent for the APNLF. More precisely,

$$\frac{d}{d\lambda}T(x_l + \lambda\Delta x_l|P_l\overline{H}_l)_{|\lambda=0} = -2T(x_l|P_l\overline{H}_l). \tag{4.5}$$

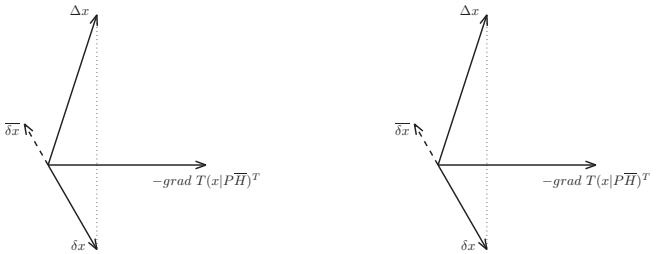
This behavior is not necessarily mimicked by the basic approximation $\overline{\delta x}_l$. In fact, it is not even guaranteed that $\overline{\delta x}_l$ is a direction of descent for the APNLF at x_l. To overcome this nuisance we define the *descent approximation*

$$\delta x_l := \frac{1}{1-\alpha_l}\overline{\delta x}_l \quad \text{where} \quad \alpha_l := \frac{\overline{\delta x}_l^T\big(I - \overline{H}_l F'(x_l)\big)\overline{\delta x}_l}{\overline{\delta x}_l^T \overline{\delta x}_l} \neq 1 \tag{4.6}$$

is assumed. Since the descent approximation is based on $\overline{\delta x}_l$ we denote the correction $\overline{\delta x}_l$ by the term *basic approximation*.

Remark 4.3 The scalar α_l may either be determined by employing the direct tangent evaluation $F'(x_l)\overline{\delta x}_l$ (forward mode of AD) or the adjoint tangent evaluation $\overline{\delta x}_l^T\overline{H}_l F'(x_l)$ (reverse mode of AD). Though the second one is slightly more expensive than the first one in terms of floating point operations, see [15], we opt for the second evaluation. From a short calculation or (4.16) it is seen that the adjoint tangent evaluation equals the negative gradient of the APNLF at x. Since one of our monitors is given by the angle between δx and the transposed negative gradient of the APNLF at x we need this evaluation anyway. □

The notation *descent approximation* for δx is indeed justifiable as the following proposition shows.



(a) Descent ensured and good approximation to Δx (b) Though descent holds approximation quality of δx is bad

Figure 4.1: Visualization of the descent property of δx. The Newton correction is denoted by Δx.

Proposition 4.4 (Descent property) *Let the descent approximation* (4.6) *be well defined and nonzero. Furthermore, let Δx_l be the Newton correction at x_l. Then the descent property*

$$\begin{aligned}\frac{d}{d\lambda}T(x_l + \lambda\delta x_l|P_l\overline{H}_l)_{|\lambda=0} &= -2T(x_l|P_l\overline{H}_l)\\ &= \frac{d}{d\lambda}T(x_l + \lambda\Delta x_l|P_l\overline{H}_l)_{|\lambda=0}\end{aligned}$$

holds.

Proof. Differentiation yields

$$\begin{aligned}\frac{d}{d\lambda}T(x_l + \lambda\delta x_l|P_l\overline{H}_l)_{|\lambda=0} &= \big(P_l\overline{H}_lF(x_l)\big)^TP_l\overline{H}_lF'(x_l)\delta x_l\\ &= \big(\overline{H}_lF(x_l)\big)^T\overline{H}_lF'(x_l)\delta x_l.\end{aligned}$$

Since

$$\overline{\delta x}_l = -\overline{H}_lF(x_l), \quad \frac{1}{1-\alpha_l} = \frac{\overline{\delta x}_l^T\overline{\delta x}_l}{\overline{\delta x}_l^T\overline{H}_lF'(x_l)\overline{\delta x}_l} \quad \text{and} \quad \overline{\delta x}_l^T\overline{\delta x}_l = 2T(x_l|P_l\overline{H}_l)$$

we obtain

$$\big(\overline{H}_lF(x_l)\big)^T\overline{H}_lF'(x_l)\delta x_l = -\overline{\delta x}_l^T\overline{H}_lF'(x_l)\frac{1}{1-\alpha_l}\overline{\delta x}_l = -\overline{\delta x}_l^T\overline{\delta x}_l = -2T(x_l|P_l\overline{H}_l).$$

The relation w.r.t. to the Newton direction is just (4.5) which follows from (2.12). ■

For the following discussion we will drop the iteration index and abbreviate $J := F'(x)$.

The update of $\overline{\delta x}$ to obtain δx has a nice geometric interpretation: The above descent property of δx is accomplished by ensuring that the orthogonal projection of δx and the Newton correction Δx onto the transposed gradient of the APNLF at the current iterate coincide—see also Figure 4.1. Certainly, such a property cannot be true if the basic approximation $\overline{\delta x}$ and $-\operatorname{grad} T(x|P\overline{H})^T = \big(\overline{\delta x}^T\overline{H}J\big)^T$ are perpendicular since the factor $1/(1-\alpha)$ is only capable of changing the length and orientation of a vector. This is directly reflected by the value of α since $\alpha = 1$ if and only if $\overline{\delta x}$

and $-\operatorname{grad} T(x|P\overline{H})^T = \left(\overline{\delta x}^T \overline{H} J\right)^T$ are perpendicular. If this is true a better approximation to the inverse of J is required. In the next section we will provide techniques to obtain such better approximations which can also be used to avoid a situation like the one depicted in Figure 4.1(b).

For δx from (4.6) the following result about the relative change of the APNLF is true.

Theorem 4.5 *Let F fulfill Assumption 2.1 and let $x \in \mathcal{D}$ such that $F(x) \neq 0$. Let $J := F'(x)$. For given $\overline{H} \in \mathbb{R}^{n\times n}$ suppose that $\overline{\delta x}$ from* (4.1) *is nonzero. Furthermore, let δx be given according to* (4.6)*. Then for all $\lambda \in \Lambda$ with*

$$\Lambda := \{\lambda \in (0,1] \mid x + \lambda\delta x \in \mathcal{D}\} \tag{4.7}$$

we obtain for the APNLF

$$\frac{T(x+\lambda\delta x|P\overline{H})}{T(x|P\overline{H})} = \left(1-\lambda+\overline{\mu}(\lambda)\right)^2 \tag{4.8}$$

with $\overline{\mu}(\lambda)$ given as

$$\begin{aligned}\overline{\mu}(\lambda) &:= -\frac{\overline{\delta x}^T}{\overline{\delta x}^T\overline{\delta x}}\overline{H}\big(F(x+\lambda\delta x) - F(x) - \lambda J\delta x\big)\\ &= \frac{\overline{\delta x}^T\overline{\delta x}_+}{\overline{\delta x}^T\overline{\delta x}} - (1-\lambda)\end{aligned} \tag{4.9}$$

where $\overline{\delta x}_+ := -\overline{H}F(x+\lambda\delta x)$.

Proof. We abbreviate $F := F(x)$. It holds that

$$P\overline{H}F(x+\lambda\delta x) = P\overline{H}F + \lambda P\overline{H}J\delta x + P\overline{\chi}(\lambda)$$

where

$$\overline{\chi}(\lambda) := \overline{H}\big(F(x+\lambda\delta x) - F - \lambda J\delta x\big).$$

Since

$$\frac{1}{1-\alpha} = \frac{\overline{\delta x}^T\overline{\delta x}}{\overline{\delta x}^T\overline{H}J\overline{\delta x}} \tag{4.10}$$

and due to the definition of δx we have

$$\frac{\overline{\delta x}^T\overline{H}J\delta x}{\overline{\delta x}^T\overline{\delta x}} = 1. \tag{4.11}$$

Hence,

$$P\overline{H}J\delta x = \overline{\delta x} = -\overline{H}F = -P\overline{H}F.$$

By the definition of $\overline{\mu}(\lambda)$ it is clear that

$$P\overline{\chi}(\lambda) = \overline{\mu}(\lambda)\overline{H}F$$

and therefore $P\overline{\chi}(\lambda) = \overline{\mu}(\lambda)P\overline{H}F$. Putting things together, we obtain

$$P\overline{H}F(x+\lambda\delta x) = \big(1-\lambda+\overline{\mu}(\lambda)\big)P\overline{H}F$$

which implies (4.8). From the definition of $\overline{\delta x}_+$ and by means of (4.11) it directly follows that

$$\overline{\mu}(\lambda) = \frac{\overline{\delta x}^T \overline{\delta x}_+}{\overline{\delta x}^T \overline{\delta x}} - (1 - \lambda).$$

■

Remark 4.6 The relation (4.8) provides the basis for the globalization approach based on quasi-Newton updates which we will discuss in Section 4.4. Due to the structural analogy of $\overline{\mu}(\lambda)$ to $\mu(\lambda)$ from (4.4) it is straightforward to adapt the step size strategies from Section 3.4—see Subsection 4.4.6 for details. □

Due to the projectional aspect of the APNLF and PNLF and the descent property of δx we obtain by mans of (4.8) an expression for the relative change of the APNLF which is structurally analogous to the one for the PNLF, cf. (4.3). However, this *does not* necessarily imply equal *behavior*. Due to a poor approximation quality of $\overline{H}$ the quantity $\overline{\mu}(\lambda)$ may produce values which are far off from the ones given by $\mu(\lambda)$ of (4.4). We ask for sufficient conditions such that $\overline{\mu}(\lambda) = \mu(\lambda)\ \forall \lambda \in \Lambda$. An answer is provided by the next theorem.

Theorem 4.7 *Let F fulfill Assumption 2.1 and let $x \in \mathcal{D}$ such that $F(x) \neq 0$. Let $J := F'(x)$ be nonsingular and Δx be the Newton correction at x. Define for given $\overline{H} \in \mathbb{R}^{n \times n}$ the approximation $\overline{\delta x}$ via* (4.1) *and let δx be given according to* (4.6)*. Furthermore, let $\overline{\mu}(\lambda)$ be as in* (4.9) *and $\mu(\lambda)$ defined via* (4.4)*. Then,*

$$\overline{\delta x} = \Delta x \quad \textit{and} \quad \overline{\delta x}^T \overline{H} J = \overline{\delta x}^T \tag{4.12a}$$

implies that

$$\delta x = \Delta x \quad \textit{and} \quad \frac{\overline{\delta x}^T \overline{H} J}{\overline{\delta x}^T \overline{\delta x}} = \frac{\delta x^T}{\delta x^T \delta x} \tag{4.12b}$$

which in turn implies that

$$\overline{\mu}(\lambda) = \mu(\lambda) \quad \forall \lambda \in \Lambda \tag{4.12c}$$

with Λ from (4.7)*.*

Proof. We abbreviate $F := F(x)$. If $\overline{\delta x} = \Delta x$ then $\overline{\delta x} \neq 0$ since $\Delta x \neq 0$. Also,

$$\overline{H} J \overline{\delta x} = \overline{\delta x}$$

which implies that

$$\alpha = \frac{\overline{\delta x}^T (I - \overline{H} J) \overline{\delta x}}{\overline{\delta x}^T \overline{\delta x}} = 0$$

and therefore $\overline{\delta x} = \delta x$ which by the assumptions means that the first part of (4.12b), i.e. $\delta x = \Delta x$, is true. From $\overline{\delta x} = \delta x$ and $\overline{\delta x}^T \overline{H} J = \overline{\delta x}$ it directly follows that the second part of (4.12b) holds. Now assume (4.12b) to be valid. Since δx is well defined we have $\overline{\delta x} \neq 0$. Employing the definition

of $\overline{\delta x}_+$ and (4.9) we obtain

$$\begin{aligned}\overline{\mu}(\lambda) &= -\frac{\overline{\delta x}^T \overline{H} J J^{-1} F(x+\lambda\delta x)}{\overline{\delta x}^T \overline{\delta x}} - (1-\lambda) = -\frac{\delta x^T J^{-1} F(x+\lambda\delta x)}{\delta x^T \delta x} - (1-\lambda) \\ &= -\frac{\Delta x^T J^{-1} F(x+\lambda\Delta x)}{\Delta x^T \Delta x} - (1-\lambda) \\ &= -\frac{\Delta x^T}{\Delta x^T \Delta x} J^{-1}\big(F(x+\lambda\Delta x) - F - \lambda J \Delta x\big) = \mu(\lambda).\end{aligned} \tag{4.13}$$

This concludes the proof.

■

The condition (4.12b) has a nice geometrical interpretation. We define for $y, z \in \mathbb{R}^n \setminus \{0\}$ the angle between these vectors via

$$\angle(y,z) := \arccos\left[\frac{y^T z}{\|y\|_2 \cdot \|z\|_2}\right] \in [0,\pi].$$

With this definition at hand we can state

Corollary 4.8 *Let the assumptions and definitions from Theorem 4.7 be given and additionally P according to Definition 4.2. Then the condition* (4.12b) *is equivalent to*

$$\angle(\delta x, \Delta x) = 0 \quad \textit{and} \quad \angle\big(\delta x, -\operatorname{grad} T(x|P\overline{H})^T\big) = 0. \tag{4.14}$$

Proof. We abbreviate again $F := F(x)$.

First, let (4.12b) be true. From $\delta x = \Delta x$ it directly follows that $\angle(\delta x, \Delta x) = 0$. Furthermore,

$$\frac{\delta x^T}{\delta x^T \delta x} = (1-\alpha)^2 \frac{\delta x^T}{\overline{\delta x}^T \overline{\delta x}}. \tag{4.15}$$

Hence, the second part of (4.12b) implies that

$$\frac{1}{(1-\alpha)^2} \overline{\delta x}^T \overline{H} J = \delta x.$$

With

$$-\operatorname{grad} T(x|P\overline{H}) = \overline{\delta x}^T \overline{H} J \tag{4.16}$$

and the fact that $(1-\alpha)^2 > 0$ we verify that the second part of (4.14) holds true.

Now suppose that the condition (4.14) is valid. Then, from $\angle(\delta x, \Delta x) = 0$ it follows that $\delta x = \beta \Delta x$ for some $\beta > 0$. By means of Proposition 4.4 we obtain

$$\operatorname{grad} T(x|P\overline{H})\delta x = \operatorname{grad} T(x|P\overline{H})\Delta x.$$

Hence, the scalar β must be equal to one which simply means that $\delta x = \Delta x$. By (4.16) and the second part of (4.14) we know that there is a $\gamma > 0$ such that

$$\delta x^T = \gamma \overline{\delta x}^T \overline{H} J.$$

Multiplying by δx from the right and exploiting (4.10) yields

$$\gamma = \frac{\delta x^T \delta x}{\overline{\delta x}^T \overline{\delta x}} = \frac{1}{(1-\alpha)^2}.$$

So it holds that

$$(1-\alpha)^2 \frac{\delta x^T}{\overline{\delta x}^T \overline{\delta x}} = \frac{\overline{\delta x}^T \overline{H} J}{\overline{\delta x}^T \overline{\delta x}}$$

which by (4.15) means

$$\frac{\delta x^T}{\delta x^T \delta x} = \frac{\overline{\delta x}^T \overline{H} J}{\overline{\delta x}^T \overline{\delta x}}.$$

This is just the second part of (4.12b).

■

If $F(x) = 0$ is a scalar problem, i.e. $n = 1$, then according to (4.10) the factor $1/(1-\alpha)$ simplifies to $(\overline{H}J)^{-1}$ which by definition of δx implies that $\delta x = \Delta x$. Keeping that in mind, a short calculation shows that also $\angle\big(\delta x, -\operatorname{grad} T(x|P\overline{H})^T\big) = 0$ holds. However, for $n \geqslant 2$ the descent property of δx does not necessarily imply (4.14). Since $\angle(\delta x, \Delta x)$ is practically not available we will provide a computable estimate for it. This estimate will be derived from the following upper bound which takes the descent property of δx into account.

Theorem 4.9 *For $n \geqslant 2$ suppose that F fulfills Assumption 2.1 and let $x \in \mathcal{D}$ such that $F(x) \neq 0$ and $F'(x)$ is nonsingular. The Newton correction at x is denoted by Δx. For given $\overline{H} \in \mathbb{R}^{n\times n}$ let $\overline{\delta x}$ be defined via* (4.1) *and different from zero. Assume that δx is given according to* (4.6)*. Let*

$$\beta := \angle\big(\delta x, -\operatorname{grad} T(x|P\overline{H})^T\big) \tag{4.17}$$

and suppose that

$$r_{rel} := \frac{\|\delta x - \Delta x\|_2}{\|\delta x\|_2} < 1. \tag{4.18}$$

Then with

$$\angle_b(\delta x, \Delta x)$$

defined via

$$\begin{aligned} \angle_b(\delta x, \Delta x) := \arccos\left[\big(1 - r_{rel}\cdot\sin(\beta)\big)\cdot\left\|\begin{pmatrix} r_{rel}\cdot\sin(\beta) - 1 \\ r_{rel}\cdot\cos(\beta)\end{pmatrix}\right\|_2^{-1}\right] \\ \textit{if}\ \ (n=2)\ \ \textit{or}\ \ (\beta = 0)\ \ \textit{or}\ \ \left(\beta > 0 \quad \textit{and}\quad \frac{r_{rel}}{\sin(\beta)} > 1\right) \\ \angle_b(\delta x, \Delta x) := \arcsin(r_{rel}) \\ \textit{otherwise} \end{aligned} \tag{4.19}$$

it holds that

$$\angle(\delta x, \Delta x) \leqslant \angle_b(\delta x, \Delta x). \tag{4.20}$$

Proof. If $r := \|\delta x - \Delta x\|_2 = 0$ then $\angle(\delta x, \Delta x) = 0$. Such is also true for $\angle_b(\delta x, \Delta x)$. Hence, (4.20) is valid in this case. For the remainder of the proof let $r > 0$.

The following lines of the proof are based on geometrical arguments. Therefore, we will provide several figures as part of the proof. For the sake of convenience we identify the symbols for given

points in a figure like, say, p_1 and p_2 with their respective position vectors. Hence, $\|p_1\|_2$, $\|p_1 - p_2\|_2$ and $\angle(p_1, p_2)$ are valid vector operations.

We split the remainder of the proof into two parts. First, the case $n = 2$ is considered, then the case $n \geqslant 3$. Note that in any case it holds by the descent property of δx that

$$0 \leqslant \beta < \tfrac{\pi}{2}. \tag{4.21}$$

$n = 2$: W.l.o.g. we may choose an orthonormal basis $\{v_1, v_2\}$ of $\mathbb{R}^2$ such that with $V = \begin{pmatrix} v_1 & v_2 \end{pmatrix}$ we can write

$$-\operatorname{grad} T(x|P\overline{H})^T = Vg \quad \text{where} \quad g := \big(\|\operatorname{grad} T(x|P\overline{H})\|_2, 0\big)^T,$$
$$\delta x = Vm \quad \text{where} \quad m := (m_{(1)}, m_{(2)})^T \quad \text{with} \quad m_{(1)} > 0 \text{ and } m_{(2)} \geqslant 0.$$

The scalar $m_{(1)}$ is positive due to the descent property of δx.

Note that for arbitrary $w \in \mathbb{R}^2$ there is a unique $\xi_w \in \mathbb{R}^2$ such that

$$w = V\xi_w, \quad \|w\|_2 = \|\xi_w\|_2. \tag{4.22a}$$

Also, with an additional $z \in \mathbb{R}^2$ it holds that

$$\angle(w, z) = \angle(\xi_w, \xi_z). \tag{4.22b}$$

The descent property of δx implies that $\Delta x = m_{(1)} v_1 + \tau v_2$ for some $\tau \in \mathbb{R}$. Hence,

$$\Delta x \in \ell \cap B$$
$$\text{where} \quad \ell := \{y \in \mathbb{R}^2 \mid y = m + s \cdot (0, 1)^T,\ s \in \mathbb{R}\}$$
$$\text{and} \quad B := \{y \in \mathbb{R}^2 \mid \|y - m\|_2 \leqslant r\}.$$

In order to obtain an upper bound for $\angle(\delta x, \Delta x)$ we consider $\ell \cap \partial B$ where ∂B denotes the boundary of B. This intersection contains only two points p_1, p_2, see Figure 4.2. Clearly, $\angle(\delta x, \Delta x) \leqslant \max_{i=1,2} \angle(m, p_i)$. Let $\alpha' := \angle(m, p_1)$ and $\alpha := \angle(m, p_2)$. Since

$$r / \|m\|_2 = r_{rel} \tag{4.23}$$

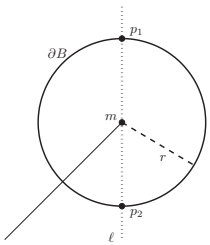
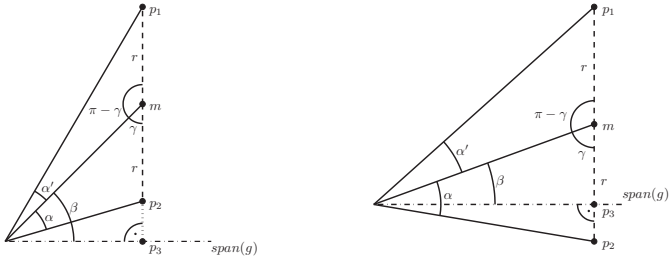


Figure 4.2: Intersection of ∂B and ℓ consists of two points



(a) $m_{(2)} > r$ (b) $m_{(2)} \leqslant r$

Figure 4.3: By the choice of the basis $\{v_1, v_2\}$ we either have $m_{(2)} > r$ or $0 \leqslant m_{(2)} \leqslant r$.

and $0 < r_{rel} < 1$ it holds that

$$0 < \alpha', \alpha < \tfrac{\pi}{2}. \tag{4.24}$$

We show now that $\alpha' \leqslant \alpha$: Either we have $m_{(2)} > r$ or $0 \leqslant m_{(2)} \leqslant r$, see Figure 4.3. In both cases it holds that $\|p_1\|_2 \geqslant \|p_2\|_2$ since either

$$\|p_1\|_2^2 = \|p_3\|_2^2 + \underbrace{\|p_3 - p_1\|_2^2}_{>\|p_3-m\|_2^2}, \qquad \|p_2\|_2^2 = \|p_3\|_2^2 + \underbrace{\|p_3 - p_2\|_2^2}_{<\|p_3-m\|_2^2}, \qquad (m_{(2)} > r)$$

or

$$\|p_1\|_2^2 = \|p_3\|_2^2 + \underbrace{\|p_3 - p_1\|_2^2}_{\geqslant\|p_3-p_2\|_2^2}, \qquad \|p_2\|_2^2 = \|p_3\|_2^2 + \|p_3 - p_2\|_2^2 \qquad (m_{(2)} \leqslant r)$$

where p_3 is given as in Figure 4.3. From the law of sines it follows that

$$\frac{\|p_1\|_2}{\sin(\pi-\gamma)} = \frac{r}{\sin(\alpha')} \qquad \text{and} \qquad \frac{\|p_2\|_2}{\sin(\gamma)} = \frac{r}{\sin(\alpha)}.$$

Since $\sin(\pi - \gamma) = \sin(\gamma)$ and $\|p_1\|_2 \geqslant \|p_2\|_2$ we obtain

$$1 \leqslant \frac{\sin(\alpha)}{\sin(\alpha')}$$

which by (4.24) implies that

$$\alpha' \leqslant \alpha.$$

To compute α we exploit that $\|m\|_2$ can be written as

$$\|p_2\|_2 \cdot \cos(\alpha) + r \cdot \cos(\gamma) = \|m\|_2.$$

Thus,

$$\cos(\alpha) = \frac{\|m\|_2 - r \cdot \cos(\gamma)}{\|p_2\|_2}.$$

It holds that

$$\|p_2\|_2 = \left\| r \cdot \begin{pmatrix} \cos(\gamma) \\ \sin(\gamma) \end{pmatrix} - \begin{pmatrix} \|m\|_2 \\ 0 \end{pmatrix} \right\|_2.$$

Also, $\gamma = \pi/2 - \beta$ which implies that $\cos(\gamma) = \sin(\beta)$ and $\sin(\gamma) = \cos(\beta)$. Hence, by means of (4.23) and (4.24) we obtain

$$\alpha = \arccos\left[\left(1 - r_{rel}\cdot\sin(\beta)\right)\cdot\left\|\begin{pmatrix} r_{rel}\cdot\sin(\beta) - 1 \\ r_{rel}\cdot\cos(\beta)\end{pmatrix}\right\|_2^{-1}\right].$$

This is just what is stated in (4.19).

$n \geqslant 3$: To obtain an upper bound for $\angle(\delta x, \Delta x)$ which takes the descent property of δx into account only a three dimensional (sub-)space of $\mathbb{R}^n$ which contains $\operatorname{grad} T(x|P\overline{H})^T$, δx and Δx needs to be considered. In analogy to the case $n = 2$ w.l.o.g. we may choose a basis $\{v_1, v_2, v_3\}$ of the above (sub-)space such that with $V = \begin{pmatrix} v_1 & v_2 & v_3\end{pmatrix}$ it holds that $V^TV = I$ and

$$\begin{gathered} -\operatorname{grad} T(x|P\overline{H})^T = Vg \quad \text{where} \quad g := \left(\|\operatorname{grad} T(x|P\overline{H})\|_2, 0, 0\right)^T, \\ \delta x = Vm \quad \text{where} \quad m := (m_{(1)}, m_{(2)}, 0)^T \quad \text{with} \quad m_{(1)} > 0 \text{ and } m_{(2)} \geqslant 0, \\ \Delta x = V\nu \quad \text{for some } \nu \in \mathbb{R}^3. \end{gathered}$$

Certainly, properties like (4.22) are also true. Let

$$B := \{y \in \mathbb{R}^3 \mid \|y - m\|_2 \leqslant r\}$$

and let ∂B be the boundary of B. Furthermore, let

$$C := \{t \in \mathbb{R}^3 \mid \exists!\sigma > 0 \quad \text{s.t.} \quad \sigma\cdot t \in \partial B\}$$

be the cone of tangent vectors w.r.t. B. From geometrical evidence and $r_{rel} < 1$, cf. Figure 4.4, it follows that

$$\angle(t, m) = const. =: \alpha_t < \tfrac{\pi}{2} \quad \forall t \in C$$

and

$$\angle(m, y) < \alpha_t \quad \forall y \in B \setminus (C \cap \partial B). \tag{4.25}$$

Thus,

$$\angle(m, \nu) \leqslant \alpha_t.$$

Since $\angle(\delta x, \Delta x) = \angle(m, \nu)$ a first upper bound is provided by α_t. By means of Figure 4.4 it is easy to see that α_t can be calculated via

$$\alpha_t = \arcsin\left(\frac{r}{\|m\|_2}\right) = \arcsin(r_{rel}). \tag{4.26}$$

In the following we will derive to what extent a refinement of the bound α_t is possible by taking the descent property of δx into account.

The intersection $C \cap \partial B$ uniquely defines a plane E_t such that $C \cap \partial B \subset E_t$. Furthermore, let

$$E_d := \{y \in \mathbb{R}^3 \mid g^Ty = 2T(x|P\overline{H})\}.$$

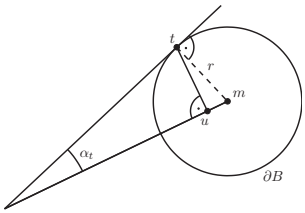


Figure 4.4: $\angle(t, m) =: \alpha_t$ provides an upper bound for $\angle(\delta x, \Delta x)$

It holds that $m, \nu \in E_d$. By the definition of E_t and E_d the intersection $E_t \cap E_d =: \ell_{td}$ is a straight line in case $\beta > 0$ or the empty set if $\beta = 0$. In addition, with $E_{12} := \text{span}\big((1,0,0)^T, (0,1,0)^T\big)$ the intersection $\ell_{td} \cap E_{12} =: A$ either consists of a single point a if $\beta > 0$ or is empty in case of $\beta = 0$, cf. Figure 4.5. Let

$$\text{diff}_{a,m} := \begin{cases} \|a - m\|_2 & \text{if } \beta > 0 \\ \infty & \text{if } \beta = 0. \end{cases}$$

Then,

$$\begin{aligned} \exists \hat{y} \in (C \cap \partial B \cap E_d) &\Leftrightarrow \ell_{td} \cap \partial B \neq \emptyset \\ &\Leftrightarrow A \subset (B \cap E_{12}) \setminus \emptyset \\ &\Leftrightarrow \text{diff}_{a,m} \leqslant r. \end{aligned} \tag{4.27}$$

Refer again to Figure 4.5 for aid to verify this. So if $\text{diff}_{a,m} \leqslant r$ we cannot exclude that $\hat{y} = \nu$. Hence, $\angle_b(\delta x, \Delta x)$ shall be equal to α_t in this case. We show now that this is true. From $\text{diff}_{a,m} \leqslant r$ it follows that $\beta > 0$. By means of Figure 4.5 we obtain

$$\text{diff}_{a,m} = \|a - m\|_2 = \frac{\|m\|_2 - \|u\|_2}{\sin(\beta)}.$$

From Figure 4.4 it follows that $\|u\|_2 = \|m\|_2 \cdot \cos^2(\alpha_t)$ and $r = \|m\|_2 \cdot \sin(\alpha_t)$. Hence,

$$\text{diff}_{a,m} = \frac{\|m\|_2 \cdot \big(1 - \cos^2(\alpha_t)\big)}{\sin(\beta)} = r \cdot \frac{\sin(\alpha_t)}{\sin(\beta)}$$

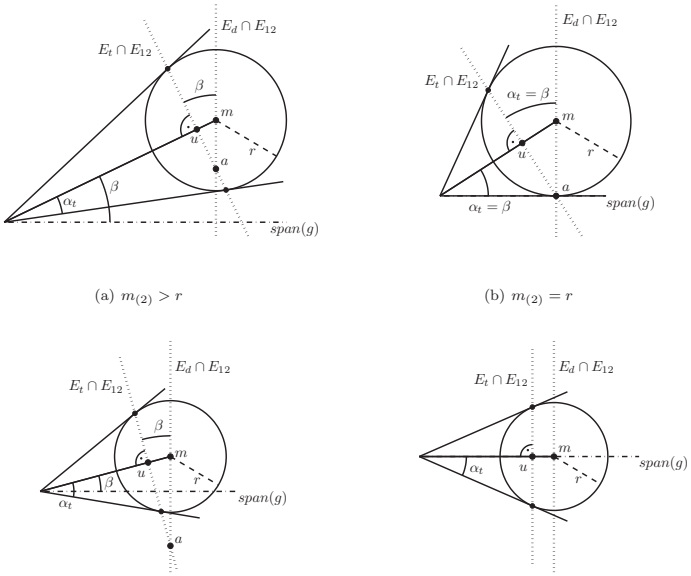
which means that

$$\text{diff}_{a,m} \leqslant r \quad \Leftrightarrow \quad \frac{\sin(\alpha_t)}{\sin(\beta)} \leqslant 1 \quad \overset{(4.26)}{\Leftrightarrow} \quad \frac{r_{rel}}{\sin(\beta)} \leqslant 1.$$

Therefore, by the definition of $\angle_b(\delta x, \Delta x)$ in (4.19) it indeed holds that $\angle_b(\delta x, \Delta x) = \alpha_t$.

Let $\text{diff}_{a,m} > r$, i.e., one of the cases depicted in Figure 4.5(c) or 4.5(d), respectively, is true. Then, $\nu \notin E_t$ but still $\nu \in (E_d \cap B)$. Hence,

$$\angle(m, \nu) \leqslant \max_{y \in (E_d \cap \partial B)} \angle(m, y). \tag{4.28}$$



(a) $m_{(2)} > r$

(b) $m_{(2)} = r$

(c) $m_{(2)} < r$

(d) $0 = m_{(2)} < r$, i.e., $\beta = 0$

Figure 4.5: Influence of the relation between $m_{(2)}$ and r on α_t, β and a

By (4.25) we have

$$\max_{y \in (E_d \cap \partial B)} \angle(m, y) < \alpha_t.$$

In order to determine the maximum in (4.28) we exploit that

$$y \in (E_d \cap \partial B) \qquad \Leftrightarrow \qquad \exists \varphi \in [0, 2\pi) \quad \text{s.t.} \quad y = y(\varphi) := m + r \cdot \begin{pmatrix} 0 \\ \cos(\varphi) \\ \sin(\varphi) \end{pmatrix}.$$

We define the function $\psi : [0, 2\pi) \to [0, \pi]$ via

$$\psi(\varphi) := \angle\big(m, y(\varphi)\big).$$

This function is well defined since $m_{(1)} > 0$ and therefore $y(\varphi) \neq 0 \, \forall \varphi \in [0, 2\pi)$. First, let $\beta > 0$. By means of a computer algebra system it is easily seen that

$$\psi'(\varphi) = \frac{r^2 \cdot \sin(\varphi) \cdot m_{(2)} \cdot \big(m_{(2)} \cdot \cos(\varphi) + r\big)}{\tilde{\psi}(\varphi)} \quad \text{where} \quad \tilde{\psi}(\varphi) > 0 \quad \forall \varphi \in [0, 2\pi).$$

The assumption $\mathrm{diff}_{a,m} > r$ is equivalent to $m_{(2)} < r$, see Figure 4.5 for evidence. Thus, extrema of ψ may only be assumed for $\varphi = 0$ and $\varphi = \pi$. A discussion of ψ'' shows that $\varphi = \pi$ is the unique maximizer of ψ. Therefore,

$$\psi(\pi) = \max_{y \in (E_d \cap \partial B)} \angle(m, y).$$

It holds that

$$y(\pi) = m + r \cdot \begin{pmatrix} 0 \\ -1 \\ 0 \end{pmatrix} \in E_{12}.$$

Thus, this quantity can be identified with p_2 from the discussion of the case $n = 2$ leading to the same expression for $\angle_b(\delta x, \Delta x)$ we already established there. If $\beta = 0$ then $\psi(\varphi) = const.$ and any value of φ will yield the maximum in (4.28), especially $\varphi = \pi$ leading to the same result as for $\beta > 0$.

■

Recall that we assume the product $\overline{\delta x}^T \overline{H}$ and adjoint evaluations $w^T J$, $w \in \mathbb{R}^n$ to be available. Therefore, β from (4.17) is computable. Hence, to provide an estimate for $\angle_b(\delta x, \Delta x)$ we have to provide an estimate for r_{rel} from (4.18). To do so, the following auxiliary result is given.

Proposition 4.10 *Suppose that F fulfills Assumption 2.1 and for $x \in \mathcal{D}$ let $F(x) \neq 0$. Furthermore, let the associated Newton correction Δx be well defined, i.e., assume that $J := F'(x)$ is nonsingular. Define for $\overline{H} \in \mathbb{R}^{n \times n}$ the basic approximation $\overline{\delta x}$ according to* (4.1)*. Consider α and δx defined according to* (4.6)*. Furthermore, suppose that*

$$\left\| I - \tfrac{1}{1-\alpha} \overline{H} J \right\|_2 < 1. \tag{4.29}$$

Then, $\overline{H}$ is nonsingular and one has

$$\frac{\|\delta x - \Delta x\|_2}{\|\delta x\|_2} \leqslant \frac{\left\| \left(I - \tfrac{1}{1-\alpha} \overline{H} J\right) \overline{\delta x} \right\|_2}{\left(1 - \left\| I - \tfrac{1}{1-\alpha} \overline{H} J \right\|_2\right) \|\overline{\delta x}\|_2}. \tag{4.30}$$

The above bound is tight in the sense that $\delta x = \Delta x$ implies that the bound is equal to zero.

Proof. We abbreviate $F := F(x)$ and $H := \frac{1}{1-\alpha} \overline{H}$. From the Perturbation Lemma and (4.29) it follows that HJ is nonsingular. Hence, H is nonsingular and we obtain

$$\begin{aligned} \delta x - \Delta x &= J^{-1} F - HF \\ &= (HJ)^{-1}(I - HJ)HF = \big(I - (I - HJ)\big)^{-1}(I - HJ)HF. \end{aligned}$$

Thus,

$$\|\delta x - \Delta x\|_2 \leqslant \|\big(I - (I - HJ)\big)^{-1}\|_2 \cdot \|(I - HJ)\delta x\|_2$$

and therefore

$$\|\delta x - \Delta x\|_2 \leqslant \frac{\|\big(I - HJ\big)\delta x\|_2}{1 - \|I - HJ\|_2}. \tag{4.31}$$

Since

$$\delta x = \frac{1}{1-\alpha} \overline{\delta x}$$

dividing (4.31) by $\|\delta x\|$ yields (4.30). If $\delta x = \Delta x$ then

$$HJ\overline{\delta x} = \overline{H}J \cdot \tfrac{1}{1-\alpha}\overline{\delta x} = \overline{H}J\delta x = \overline{H}J\Delta x = -\overline{H}F = \overline{\delta x}.$$

Thus, the bound in (4.30) becomes zero.

■

For the bound (4.30) we need to compute $\overline{HJ\overline{\delta x}}$. This is one of the products from (4.2) which is assumed to be available. Note that $J\overline{\delta x}$ is efficiently computable via the forward mode of AD. Unfortunately, the bound (4.30) also depends on an operator norm, namely $\left\|\left(I - \frac{1}{1-\alpha}\overline{H}J\right)\right\|_2$, which is usually computationally not available. So we will estimate it. Such an estimate should introduce only minor computational effort. Recall from Remark 4.3 that the product $\overline{\delta x}^T\overline{H}J$ was used to determine α and therefore is available. Furthermore, the bound (4.30) itself provides the quantity $\left\|\left(I - \frac{1}{1-\alpha}\overline{H}J\right)\overline{\delta x}\right\|_2 / \|\overline{\delta x}\|_2$. Hence, we define

$$\operatorname{opn}_{est} := \max\left[\frac{\left\|\left(I - \frac{1}{1-\alpha}\overline{H}J\right)\overline{\delta x}\right\|_2}{\|\overline{\delta x}\|_2}, \frac{\left\|\overline{\delta x}^T\left(I - \frac{1}{1-\alpha}\overline{H}J\right)\right\|_2}{\|\overline{\delta x}\|_2}\right] \leqslant \left\|\left(I - \tfrac{1}{1-\alpha}\overline{H}J\right)\right\|_2. \tag{4.32}$$

To determine opn_{est} only a computational effort of $\mathcal{O}(n)$ floating point operations arises if we do not take the costs for $\overline{HJ\overline{\delta x}}$ and $\overline{\delta x}^T\overline{H}J$ into account. Assume that $\operatorname{opn}_{est} < 1$. Then according to the above proposition we define our estimate r_{rel}^{est} for r_{rel} from (4.18) via

$$r_{rel}^{est} := \frac{1}{1 - \operatorname{opn}_{est}} \cdot \frac{\left\|\left(I - \frac{1}{1-\alpha}\overline{H}J\right)\overline{\delta x}\right\|_2}{\|\overline{\delta x}\|_2}. \tag{4.33}$$

Note that analogously to the bound (4.30) the estimate r_{rel}^{est} is tight in the sense that $\delta x = \Delta x$ implies that $r_{rel}^{est} = 0$. If $r_{rel}^{est} < 1$ we substitute it for r_{rel} in (4.19) to finally obtain our estimate $\angle_{est}(\delta x, \Delta x)$ for $\angle(\delta x, \Delta x)$. For an algorithmic summary of computing r_{rel}^{est} and $\angle_{est}(\delta x, \Delta x)$ refer to Algorithm 4.1 and 4.2.

Motivated by Theorem 4.7 and Corollary 4.8 we use the quantities $\beta = \angle\big(\delta x, -\operatorname{grad} T(x|P\overline{H})^T\big)$ and $\angle_{est}(\delta x, \Delta x)$ to monitor the approximation quality of δx and the APNLF. Basically, we accept the descent approximation and the APNLF as a substitute for Δx and the PNLF, respectively, if

$$\beta \leqslant \phi \qquad \text{and} \qquad \angle_{est}(\delta x, \Delta x) \leqslant \psi \tag{4.34}$$

for predefined $0 \leqslant \phi, \psi < \frac{\pi}{2}$. One of these checks may fail to pass due to a bad approximation quality of $\overline{H}$. Even worth, such an insufficiency of $\overline{H}$ may lead to $\operatorname{opn}_{est} > 1$ or $r_{rel}^{est} > 1$ making our estimate $\angle_{est}(\delta x, \Delta x)$ unavailable. Therefore, we need a concept to improve the quality of $\overline{H}$. Our approach is based on quasi-Newton techniques which we will discuss in the next section.

Algorithm 4.1 (Calculating r_{rel}^{est})

---

given: $\overline{H}$, $F := F(x) \neq 0$, $J := F'(x)$ nonsingular
determine $\overline{\delta x} = -\overline{H}F$ ▷ cf. (4.1)
if $\overline{\delta x} = 0$ **then**
    increase approximation quality of $\overline{H}$ and reinvoke this algorithm
end if
determine $\hat{g} = \overline{\delta x}^T \overline{H}$
determine $g = \hat{g}J$ ▷ i.e. $-\operatorname{grad} T\big(x|P\overline{H}\big)$, done by reverse mode of AD
determine $s = \frac{\overline{\delta x}^T \overline{\delta x}}{\overline{\delta x}^T \overline{H} J \overline{\delta x}}$ ▷ i.e. $\frac{1}{1-\alpha}$, cf. (4.10)
determine $u = J\overline{\delta x}$ ▷ done by forward mode of AD
determine $\tilde{r} = \overline{\delta x} - s \cdot \overline{H}u$ ▷ i.e. $(I - \frac{1}{1-\alpha}\overline{H}J)\overline{\delta x}$
set $\operatorname{opn}_{est} = \max\Big(\|\tilde{r}\|_2/\|\overline{\delta x}\|_2,\ \|\overline{\delta x}^T - s \cdot g\|_2/\|\overline{\delta x}\|_2\Big)$ ▷ cf. (4.32)
if $\operatorname{opn}_{est} \geqslant 1$ **then**
    increase approximation quality of $\overline{H}$ and reinvoke this algorithm
end if
set $r_{rel}^{est} = \frac{1}{1-\operatorname{opn}_{est}} \cdot \frac{\|\tilde{r}\|_2}{\|\overline{\delta x}\|_2}$ ▷ cf. (4.33)
if $r_{rel}^{est} \geqslant 1$ **then**
    increase approximation quality of $\overline{H}$ and reinvoke this algorithm
end if

Algorithm 4.2 (Calculating $\angle_{est}(\delta x, \Delta x)$)

---

given: g, $\overline{\delta x}$ and $r_{rel}^{est} < 1$ from Algorithm 4.1
determine $\beta = \arccos\left[\frac{g \cdot \overline{\delta x}}{\|g\|_2 \cdot \|\overline{\delta x}\|_2}\right]$ ▷ cf. (4.17)
if $n = 2$ || $\beta = 0$ || $\big(\beta > 0$ && $r_{rel}^{est}/\sin(\beta) > 1\big)$ **then** ▷ cf. (4.19)
    set $\angle_{est}(\delta x, \Delta x) = \arccos\left[\big(1 - r_{rel}^{est} \cdot \sin(\beta)\big) \cdot \left\| \begin{pmatrix} r_{rel}^{est} \cdot \sin(\beta) - 1 \\ r_{rel}^{est} \cdot \cos(\beta) \end{pmatrix} \right\|_2^{-1}\right]$
else
    set $\angle_{est}(\delta x, \Delta x) = \arcsin(r_{rel}^{est})$
end if

4.2 Purifying Updates

In the previous section we introduced the APNLF, an approximation for the PNLF, and an approximation δx for the Newton correction Δx at $x \in \mathcal{D}$ in terms of a given approximation $\overline{H}$ for $F'(x)^{-1}$. We also introduced the angle checks (4.34) to monitor the quality of this approximations, i.e.,

$$\angle\big(\delta x, -\operatorname{grad} T(x|P\overline{H})^T\big) \leqslant \phi \qquad \text{and} \qquad \angle_{est}(\delta x, \Delta x) \leqslant \psi \tag{4.35}$$

for predefined $0 \leqslant \phi, \psi < \frac{\pi}{2}$. If both angles are zero then by Theorem 4.7 and Corollary 4.8 it holds that $\delta x = \Delta x$ and the APNLF behaves equal to the PNLF in the direction of δx. In this section we will provide techniques to polish up the approximation quality of $\overline{H}$ if one of the checks fails to pass or if one of the quantities opn_{est} and r_{rel}^{est} from (4.32) and (4.33), respectively, is bigger than one, which makes the angle estimate $\angle_{est}(\delta x, \Delta x)$ unavailable.

Inspired by the work of Schlenkrich, [28], we will employ specific rank-1 updates to improve the quality of $\overline{H}$ which we call *purifying updates*. We will show that if $F'(x)$ is nonsingular an iterative application of these purifying updates eventually leads to an approximation $\overline{H}$ such that (4.12a) is fulfilled, hence (4.14) and therefore the angle checks in (4.35) are passed.

We will formulate the purifying updates in such a way that they are affine covariant compatible if the initial Jacobian is affine covariant compatible. Therefore, the inverse of $\overline{H}$ is affine covariant compatible as well and hence δx and $\overline{\mu}(\lambda)$ are affine covariant. This means, the basis for an affine covariant globalization approach is given.

Remark 4.11 Recall from the introduction of this chapter that Schlenkrich's approach is based on the classical level function $T(x|I) = \frac{1}{2}\|F(x)\|_2^2$. The gradient of this level function does not depend on the approximation $\overline{H}$. So the purpose of Schlenkrich's purifying update is to provide a better approximation w.r.t to Δx. Though Δx is a direction of descent to $T(x|I)$ the angle between Δx and $-\operatorname{grad} T(x|I)^T$ by no means need to be close to zero. E.g., consider the example from Subsection 3.2.7. There, for $a = 50$ and at $x_0 = (50, 1)^T$ we have $\Delta x = -(50, 1)^T$ and $-\operatorname{grad} T(x_0|I) = -50 \cdot (1, 50)$ which results in an angle of almost $\pi/2$. It is an inherent weakness of this approach that usually the correction δx cannot be a good approximation to the Newton correction *and* the transposed negative gradient of $T(x|I)$ simultaneously. Often, this leads to unnecessary small step sizes. Such a drawback is not existent in the context of the APNLF as it will be seen from Corollary 4.18 below. □

In the context of improving the approximation quality of $\overline{H}$ we consider approximations $A_k \in \mathbb{R}^{n\times n}$ to the Jacobian $J := F'(x)$ and formulate the purifying updates as corrections to these approximations. More precisely, starting with a matrix A_0 we will construct a sequence of matrices $\{A_k\}$ such that for

$$\mathcal{W}_k := \ker\big((A_k - J)^T\big) = \{u \in \mathbb{R}^n \mid u^T A_k = u^T J\}$$

and

$$\mathcal{T}_k := \ker\big((A_k - J)\big) = \{y \in \mathbb{R}^n \mid A_k y = Jy\} \tag{4.36}$$

and with

$$\nu_k := \dim(\mathcal{T}_k) \quad \big(= \dim(\mathcal{W}_k)\big) \tag{4.37}$$

it holds that

$$\nu_k \geqslant k. \tag{4.38}$$

This means for nonsingular J that in a finite number of purifying steps the conditions (4.12a) are satisfied—cf. Corollary 4.18.

Remark 4.12 As we will see, for the construction of the sequence $\{A_k\}$ it is of no importance that the matrix J is the evaluation of F' at an $x \in \mathcal{D}$. The process is simply a procedure to construct to a given fixed matrix J a sequence of approximations such that (4.38) is fulfilled. Therefore, for following statements we will drop the relation between J and F' whenever J just needs to be such a fixed matrix. □

We will introduce in the next subsection three types of purifying updates. All three updates are of the same basic structure: Let $A_k \in \mathbb{R}^{n\times n}$ be the current approximation to $J \in \mathbb{R}^{n\times n}$. Assume that $A_k \neq J$. Then the next approximation A_{k+1} is given via

$$A_{k+1} = A_k - \frac{(A_k - J)d_k w_k^T (A_k - J)}{w_k^T (A_k - J)d_k}. \tag{4.39}$$

We call the above update *the basic purifying update*. The choice of d_k and $w_k \in \mathbb{R}^n$ depends on which specific purifying update is considered. But in any case we assume that $w_k^T(A_k - J)d_k \neq 0$. It is readily seen that such vectors always exist if $A_k \neq J$.

Remark 4.13 Our basic purifying update is the two-sided-rank-one (TR1) update which was originally introduced in [14] in the context of constrained optimization. Also in [29] it is applied to stiff ODEs. In the following we stick to the notation *basic purifying update* to emphasize its purpose in our context. □

In the following we will consider the basic purifying update (4.39) and will exploit its properties to show that (4.38) holds true and that for nonsingular J eventually a nonsingular approximation $A_{\bar{k}}$ exists such that $\overline{H} = A_{\bar{k}}^{-1}$ fulfills the conditions (4.12a). Afterwards we will discuss our three specific choices of purifying updates.

The essential properties of the basic purifying update (4.39) are as follows.

Proposition 4.14 *Let A_k, $J \in \mathbb{R}^{n\times n}$ be given such that $A_k \neq J$. Assume for the vectors w_k, $d_k \in \mathbb{R}^n$ that $w_k^T(A_k - J)d_k \neq 0$ holds and let A_{k+1} be given according to* (4.39).

I) With $\mathcal{W}_k$ and $\mathcal{T}_k$ as defined in (4.36) *it holds that*

$$\mathcal{W}_k + \operatorname{span}(w_k) = \mathcal{W}_{k+1}, \qquad \mathcal{T}_k + \operatorname{span}(d_k) = \mathcal{T}_{k+1} \tag{4.40a}$$

and also for ν_k from (4.37),

$$\nu_{k+1} = \nu_k + 1. \tag{4.40b}$$

II) Let $f \in \mathbb{R}^n \setminus \{0\}$.

(a) If A_{k+1} and J are nonsingular then

$$A_{k+1}^{-1} f \in \mathcal{T}_{k+1} \quad \Leftrightarrow \quad A_{k+1}^{-1} f = J^{-1} f. \tag{4.41}$$

(b) If A_{k+1} is nonsingular then

$$\left(A_{k+1}^{-1} f\right)^T A_{k+1}^{-1} \in \mathcal{W}_{k+1} \quad \Leftrightarrow \quad \left(A_{k+1}^{-1} f\right)^T = \left(A_{k+1}^{-1} f\right)^T A_{k+1}^{-1} J.$$

III) If A_{k+1} is singular and $\ker(A_{k+1}) \cap \mathcal{T}_{k+1} \neq \{0\}$ or $\ker(A_{k+1}^T) \cap \mathcal{W}_{k+1} \neq \{0\}$ then J is singular.

Proof.

I) Regarding the statements in (4.40a) we only prove the one related to $\mathcal{T}$. The relation w.r.t. $\mathcal{W}$ is verified in an analogous way.

For $z \in \mathbb{R}^n$ we abbreviate $\xi = \xi(z) := w_k^T(A_k - J)z \,/\, w_k^T(A_k - J)d_k$. Then we have by the definition of ξ and of A_{k+1} in (4.39),

$$\begin{aligned} z \in \mathcal{T}_{k+1} &\Leftrightarrow A_{k+1}z = Jz \Leftrightarrow A_k(z - \xi d_k) = J(z - \xi d_k) \Leftrightarrow z - \xi d_k \in \mathcal{T}_k \\ &\Leftrightarrow z \in \operatorname{span}(d_k) + \mathcal{T}_k. \end{aligned} \tag{4.42}$$

The last of the above equivalences may be verified in the following way: The validity of the implication $z - \xi d_k \in \mathcal{T}_k \Rightarrow z \in \operatorname{span}(d_k) + \mathcal{T}_k$ is evident. On the other hand, if $z \in \operatorname{span}(d_k) + \mathcal{T}_k$ is true then there is a $\zeta \in \mathbb{R}$ and a $\tau_z \in \mathcal{T}_k$ such that $z = \zeta d_k + \tau_z$. Hence, $\xi(z) = \zeta$ which implies that $z - \xi d_k \in \mathcal{T}_k$.

Since (4.42) holds we obtain $\mathcal{T}_k + \operatorname{span}(d_k) = \mathcal{T}_{k+1}$ which is just the relation from (4.40a).

Considering (4.40b) it is sufficient to show that $\dim(\mathcal{T}_{k+1}) = \dim(\mathcal{T}_k) + 1$:

The assumption $w_k^T(A_k - J)d_k \neq 0$ implies that $(A_k - J)d_k \neq 0$. Hence, $d_k \notin \mathcal{T}_k$. This means that

$$\dim\big(\mathcal{T}_k + \operatorname{span}(d_k)\big) = \dim(\mathcal{T}_k) + 1$$

and by (4.40a) we have $\dim(\mathcal{T}_{k+1}) = \dim(\mathcal{T}_k) + 1$.

II) (a) With the assumed nonsingularity of A_{k+1} and J it holds that

$$\begin{aligned} A_{k+1}^{-1}f \in \mathcal{T}_{k+1} &\Leftrightarrow A_{k+1}A_{k+1}^{-1}f = JA_{k+1}^{-1}f \\ &\Leftrightarrow f = JA_{k+1}^{-1}f \\ &\Leftrightarrow J^{-1}f = A_{k+1}^{-1}f. \end{aligned}$$

(b) For nonsingular A_{k+1} we obtain

$$\begin{aligned} \big(A_{k+1}^{-1}f\big)^T A_{k+1}^{-1} \in \mathcal{W}_{k+1} &\Leftrightarrow \big(A_{k+1}^{-1}f\big)^T A_{k+1}^{-1}A_{k+1} = \big(A_{k+1}^{-1}f\big)^T A_{k+1}^{-1}J \\ &\Leftrightarrow \big(A_{k+1}^{-1}f\big)^T = \big(A_{k+1}^{-1}f\big)^T A_{k+1}^{-1}J. \end{aligned}$$

III) Assume that $M := \ker(A_{k+1}) \cap \mathcal{T}_{k+1} \neq \{0\}$. Then there is a $d \in M \setminus \{0\}$ such that

$$0 = A_{k+1}d = Jd.$$

Hence, J is singular. Analogously, if $N := \ker(A_{k+1}^T) \cap \mathcal{W}_{k+1} \neq \{0\}$ then there is a $w \in N \setminus \{0\}$ with

$$0 = w^T A_{k+1} = w^T J$$

which implies that J is singular.

■

Remark 4.15 The relations $\mathcal{W}_k \subset \mathcal{W}_{k+1}$ and $\mathcal{T}_k \subset \mathcal{T}_{k+1}$ describe the basic approximation concept: Preserve the directions for which the approximation behaves like J and add new ones. In [28] such an aspect of preserving is denoted by the term *heredity*. We shall adopt this denotation as well. Note that paragraph I) is a refinement of Lemma 4.1.2 in [28] since here we prove the equalities (4.40a) instead of just the inclusions $\mathcal{W}_k + \operatorname{span}(w_k) \subseteq \mathcal{W}_{k+1}$ and $\mathcal{T}_k + \operatorname{span}(d_k) \subseteq \mathcal{T}_{k+1}$. □

An iterative application of the basic purifying update results in the following algorithm:

Algorithm 4.3 (Basic purifying process)

1: given: A_0, $J \in \mathbb{R}^{n\times n}$
2: set $k = 0$
3: **while** $A_k \neq J$ **do**
4: determine w_k, $d_k \in \mathbb{R}^n$ such that $w_k^T(A_k - J)d_k \neq 0$ is true
5: determine A_{k+1} from A_k, J, w_k and d_k via the basic purifying update (4.39)
6: set $k = k + 1$
7: **end while**

This algorithm terminates after a finite number of steps delivering at its end an A_K with $A_K = J$. This is the contents of the following proposition.

Proposition 4.16 *Let $S \coloneqq \{A_k\}$ be the sequence of matrices constructed by Algorithm 4.3 and let ν_0 be defined according to* (4.37).

I) For $K = n - \nu_0 \leqslant n$ one has $A_k = J$, such that S is finite.

II) If J and $A_i \in S$, $i < K$, are nonsingular and if for given $f \in \mathbb{R}^n \setminus \{0\}$ we have $A_i^{-1}f = J^{-1}f$ then for every nonsingular matrix $A_k \in S$ with $k > i$ it also holds that $A_k^{-1}f = J^{-1}f$.

Proof.

I) As long as $A_k \neq J$ it is evident that line 4 of Algorithm 4.3 always yields a w_k and d_k with the property $w_k^T(A_k - J)D_k \neq 0$. Since S is constructed by employing the update (4.39) and by means of (4.40b) from Proposition 4.14 an induction argument shows that it takes $K = n - \nu_0$ iteration steps to obtain

$$\nu_K = n$$

which is equivalent to $A_K = J$.

II) By (4.41) of Proposition 4.14 we have

$$A_i^{-1}f = J^{-1}f \quad \Leftrightarrow \quad A_i^{-1}f \in \mathcal{T}_i.$$

By induction it follows from (4.40a) that $\mathcal{T}_i \subseteq \mathcal{T}_k$ for $K \geqslant k > i$. Hence, $J^{-1}f \in \mathcal{T}_k$. If A_k is nonsingular then this implies that

$$A_k J^{-1}f = JJ^{-1}f \quad \Leftrightarrow \quad A_k^{-1}f = J^{-1}f.$$

■

Remark 4.17 If we identify J with $F'(x)$ and f with $-F(x)$ in the above proposition then paragraph II) shows that once the approximate correction $\overline{\delta x}_i = -A_i^{-1}F(x)$ equals the Newton correction that this is also true for the corrections related to subsequent nonsingular approximations A_k. This is due to the exploited heredity concept.

Note that if

$$\overline{\delta x}_i^T = \overline{\delta x}_i^T A_i^{-1} F'(x) \tag{4.43}$$

holds, i.e., the approximate correction $\overline{\delta x}_i$ and the transposed negative gradient of the APNLF for $\overline{H} = A_i^{-1}$ coincide, this does not necessarily imply that (4.43) is also true for *all* subsequent indices $k > i$. However, as the first paragraph of the above proposition shows eventually there will be a subsequent index such that (4.43) holds. □

By means of the results of Proposition 4.14 and 4.16 we obtain

Corollary 4.18 *Let F fulfill Assumption 2.1 and for $x \in \mathcal{D}$ assume that $F(x) \neq 0$ and that $J := F'(x)$ is nonsingular. Let Δx be the Newton correction at x. For given $A_0 \in \mathbb{R}^{n\times n}$ define ν_0 according to* (4.37) *and consider the sequence of matrices $\{A_k\}$ constructed by Algorithm 4.3. Then there is an index*

$$\hat{k} \leqslant n - \nu_0 \tag{4.44}$$

such that $A_{\hat{k}}$ is nonsingular and for $\overline{H} := A_{\hat{k}}^{-1}$ and $\overline{\delta x} := -\overline{H}F(x)$ it holds that

$$\overline{\delta x} = \Delta x \quad \textit{and} \quad \overline{\delta x}^T \overline{H} J = \overline{\delta x}^T. \tag{4.45}$$

Therefore, there is a second index $\bar{k}$ with $\bar{k} \leqslant \hat{k}$ such that $A_{\bar{k}}$ is nonsingular and for given ϕ, $\psi \geqslant 0$ the angle checks (4.35) *are passed for the choice $\overline{H} := A_{\bar{k}}^{-1}$.*

Proof. By means of Proposition 4.16 I) we know that $A_K = J$ for $K = n - \nu_0$. Hence, there is an index $\hat{k} \leqslant K$ such that $A_{\hat{k}}$ is nonsingular and

$$A_{\hat{k}}^{-1}\big(-F(x)\big) \in \mathcal{T}_{\hat{k}} \quad \text{and} \quad \Big[A_{\hat{k}}^{-1}\big(-F(x)\big)\Big]^T A_{\hat{k}}^{-1} \in \mathcal{W}_{\hat{k}}$$

with $\mathcal{T}_{\hat{k}}$ and $\mathcal{W}_{\hat{k}}$ defined according to (4.36). By Proposition 4.14 II) and the above definition of $\overline{H}$ and $\overline{\delta x}$ this is equivalent to (4.45). That there is an index $\bar{k}$ with the stated properties follows directly from the fact that (4.45) implies (4.12b) which by Corollary 4.8 means that the angle checks (4.35) are fulfilled for $\phi = \psi = 0$ if $\overline{H} = A_{\hat{k}}^{-1}$.

■

Recall from Theorem 4.7 that (4.45) is the sufficient condition (4.12a) from Theorem 4.7 which ensures that the APNLF behaves like the PNLF for all $\lambda \in \Lambda$ with Λ from (4.7).

4.2.1 Three specific purifying updates

The above results are in terms of the basic purifying update (4.39) which depends on the vector quantities w_k and d_k. So far the choice of w_k and d_k is arbitrary as long as $w_k^T(A_k - J)d_k \neq 0$ is fulfilled. Next, we will introduce three specific shapes of the basic purifying update. Our choices are made taking the following concepts into account:

- *maintaining affine covariance compatibility*
  If A_k is affine covariant compatible we have to ensure that a purifying update produces an A_{k+1} which is also affine covariant compatible in order to realize an affine covariant globalization approach.

- *delayed approximation*
  From Proposition 4.14 we know that if for

$$\overline{w}_k^T := \left[A_k^{-1}\big(-F(x)\big)\right]^T A_k^{-1} \quad \text{and} \quad \overline{d}_k := A_k^{-1}\big(-F(x)\big)$$

  and nonsingular $F'(x)$ it holds that

$$A_k\overline{d}_k = F'(x)\overline{d}_k \quad \text{and} \quad \overline{w}_k^T A_k = \overline{w}_k^T F'(x) \tag{4.46}$$

  then the choice $\overline{H} = A_k^{-1}$ implies (4.45). Note that the first of the above relations means that we can express the Newton correction via A_k, i.e., $-A_k^{-1}F(x) = -F'(x)^{-1}F(x)$. The second one simply translates to $\operatorname{grad} T(x|P\overline{H}) = \big(A_k^{-1}F(x)\big)^T$ for $\overline{H} = A_k^{-1}$.

  If (4.46) is not true we construct A_{k+1} by ensuring that at least one of the two above properties is true for this next approximation, i.e., A_{k+1} fulfills

$$A_{k+1}\overline{d}_k = F'(x)\overline{d}_k \quad \text{and/or} \quad \overline{w}_k^T A_{k+1} = \overline{w}_k^T F'(x). \tag{4.47}$$

  By means of the above interpretation of (4.46) the first relation of (4.47) may be interpreted in a way that A_{k+1} is hoped to provide a better approximation in terms of the Newton correction. Regarding the second relation such hope refers to the above given gradient statement.

Note that we cannot provide a result which characterizes our below stated choices for w_k and d_k as optimal among all possible choices of w_k and d_k such that (4.45) or (4.35), respectively, are fulfilled for a minimum number of iteration steps. However, our numerical tests confirm that our choices are reasonable. Further details about the application of the upcoming purifying updates in the context of a damped quasi-Newton iteration where step sizes are determined by means of the APNLF are discussed in Section 4.4.

We assume that $F(x) \neq 0$ and abbreviate $F := F(x)$ and $J := F'(x)$. Let A_k be given and affine covariant compatible. Additionally, if A_k is nonsingular let $\overline{\delta x}_k := -A_k^{-1}F$. The special case of singular A_k will be discussed in Paragraph 4.2.1.3.

4.2.1.1 The duophilic update

Assume A_k to be nonsingular. Choose

$$w_k^T := \overline{w}_k^T, \quad \text{i.e.,} \quad w_k^T = \overline{\delta x}_k^T A_k^{-1} \qquad \text{and} \qquad d_k := \overline{d}_k, \quad \text{i.e.,} \quad d_k = \overline{\delta x}_k$$

in the definition of the basic purifying update (4.39) and assume that

$$\overline{\delta x}_k^T (I - A_k^{-1}J)\overline{\delta x}_k \neq 0. \tag{4.48}$$

Then the *duophilic update*

$$A_{k+1} = A_k - \frac{(A_k - J)\overline{\delta x}_k \overline{\delta x}_k^T (I - A_k^{-1}J)}{\overline{\delta x}_k^T (I - A_k^{-1}J)\overline{\delta x}_k} \tag{4.49}$$

is well defined. It is readily seen that A_{k+1} is affine covariant compatible. The name of the update is motivated by the fact that both properties from (4.47) are fulfilled. Note that the adjoint tangent evaluation $\overline{\delta x}_k A_k^{-1} J$ is already at hand due to the calculation of α from (4.6). If r_{rel}^{est} from Algorithm 4.1 was already considered also the direct tangent evaluation $J\overline{\delta x}$ is available, cf. line 9 of Algorithm 4.1. This makes this update cheap to evaluate. However, it is not well defined if (4.48) is not true. Three cases may occur:

I)
$$(I - A_k^{-1}J)\overline{\delta x}_k = 0. \tag{4.50}$$
In this case it is safe to assume that $\overline{\delta x}_k^T (I - A_k^{-1}J) \neq 0$. Otherwise, the checks (4.35) would have been passed for $\overline{H} = A_k^{-1}$ and no purifying would have been considered. To proceed with the purifying process we then employ the update which we will introduce in Paragraph 4.2.1.2.

II)
$$\overline{\delta x}_k^T (I - A_k^{-1}J) = 0.$$
Analogously to the first case, it is safe to assume that $(I - A_k^{-1}J)\overline{\delta x}_k \neq 0$. Otherwise no purifying would have been initiated. In this case we proceed with the update we will introduce in Paragraph 4.2.1.3.

III)
$$\overline{\delta x}_k^T (I - A_k^{-1}J)\overline{\delta x}_k = 0$$
but
$$(I - A_k^{-1}J)\overline{\delta x}_k \neq 0 \quad \text{and} \quad \overline{\delta x}_k^T (I - A_k^{-1}J) \neq 0.$$
In this case both updates from the next two paragraphs will be applicable. In Section 4.4 we will present a purifying strategy which determines which of the following two updates will be used in this case.

4.2.1.2 The gradientphilic update

Assume A_k to be nonsingular. Choose

$$w_k^T := \overline{w}_k^T, \quad \text{i.e.,} \quad w_k^T = \overline{\delta x}_k^T A_k^{-1} \qquad \text{and} \qquad d_k := (I - A_k^{-1}J)^T \overline{\delta x}_k$$

in the definition of the basic purifying update (4.39) and assume that $d_k \neq 0$. Then the *gradientphilic update*

$$A_{k+1} = A_k - \frac{(A_k - J)(I - A_k^{-1}J)^T \overline{\delta x}_k \overline{\delta x}_k^T (I - A_k^{-1}J)}{\|\overline{\delta x}_k^T (I - A_k^{-1}J)\|_2^2} \tag{4.51}$$

is well defined. It is directly verified that A_{k+1} is affine covariant compatible. This update is guaranteed to fulfill only the second relation of (4.47), instead of $A_{k+1}\overline{d}_k = J\overline{d}_k$ we have $A_{k+1}d_k = Jd_k$.

Furthermore, the direct tangent evaluation Jd_k needs to be computed. This is a quantity which is necessary to be available solely for this update. Hence, we will consider this update only for some of the cases where the duophilic update is not well defined—see Section 4.4 for details.

4.2.1.3 The Newton-philic update

Let A_k be given and a nonsingular $A_w \in \mathbb{R}^{n\times n}$. If A_k is nonsingular we set $A_w = A_k$. Let $\overline{\delta x}_k \neq 0$ be determined according to

$$A_k\overline{\delta x}_k = \begin{cases} -F & \text{if } A_k \text{ is nonsingular} \\ 0 & \text{if } A_k \text{ is singular} \end{cases} \tag{4.52}$$

and let

$$w_k^T := \left(A_w^{-1}(A_k - J)\overline{\delta x}_k\right)^T A_w^{-1} \qquad \text{and} \qquad d_k := \overline{\delta x}_k.$$

Assume that

$$\|A_w^{-1}(A_k - J)\overline{\delta x}_k\|_2 \neq 0. \tag{4.53}$$

Then the *Newton-philic update*

$$A_{k+1} = A_k - \frac{(A_k - J)\overline{\delta x}_k \left(A_w^{-1}(A_k - J)\overline{\delta x}_k\right)^T A_w^{-1}(A_k - J)}{\|A_w^{-1}(A_k - J)\overline{\delta x}_k\|_2^2} \tag{4.54}$$

is well defined. If A_w is affine covariant compatible this is also true for A_{k+1}. The Newton-philic update fulfills the first relation of (4.47) if A_k is nonsingular. The second one is not met since instead of $\overline{w}_k^T A_{k+1} = \overline{w}_k^T J$ we have $w_k^T A_{k+1} = w_k^T J$. Note that the adjoint tangent evaluation $w_k^T J$ is required, also the direct tangent evaluation $J\overline{\delta x}$ is definitely not at hand if A_k is singular. We will employ this update if A_k is singular or in some of the cases where the duophilic update is not well defined due to (4.48) not being true, refer to Section 4.4 for more information.

If A_k is singular repeated application of this update may eventually lead to a nonsingular approximation. If this is not the case, i.e., all subsequent approximations stay singular then due to heredity there will be an index k_s such that A_{k_s} is singular, i.e., $A_{k_s}\overline{\delta x}_{k_s} = 0$ and (4.53) is no longer true. But this directly implies that also $J\overline{\delta x}_{k_s} = 0$ which means that J is singular, cf. paragraph III) of Proposition 4.14. So there is an opportunity to detect singularity of J via our approximations.

Remark 4.19 Note that the duophilic and gradientphilic updates may be adapted such that a singular A_k can be handled as well. Simply define $\overline{\delta x}_k$ via (4.52) in these cases too and substitute $A_w^{-1}(A_k - J)$ for $(I - A_k^{-1}J)$. However, (part of) the purpose of these updates is to deal with gradient information of the APNLF and such is not well defined for singular A_k. So we refrain from introducing such adaptations. □

4.3 The Descent Update

For nonsingular $J := F'(x)$, $x \in \mathcal{D}$, and for $F(x) \neq 0$ the purifying updates discussed in the previous section ensure that there is a nonsingular approximation $\overline{H}$ for J^{-1} available such that δx from (4.6) is well defined and the checks (4.35) are passed or eventually the conditions (4.12a) hold, respectively.

Assuming that no purifying updates are necessary to pass (4.35) there is at least the approximation step (4.6) with

$$\delta x = \frac{1}{1-\alpha}\overline{\delta x}$$

and

$$\overline{\delta x} = -\overline{H}F(x), \qquad \alpha = \frac{\overline{\delta x}^T(I - \overline{H}J)\overline{\delta x}}{\overline{\delta x}^T\overline{\delta x}} \neq 1,$$

executed to meet the descent property from Proposition 4.4.

We introduce the update

$$H := \left[I + \frac{1}{1-\alpha} \cdot \frac{\overline{\delta x}\overline{\delta x}^T}{\overline{\delta x}^T\overline{\delta x}}(I - \overline{H}J)\right]\overline{H}. \tag{4.55}$$

It is well defined since $\alpha \neq 1$ and it holds that

$$\begin{aligned} -HF = H\big(-F(x)\big) &= \left[I + \frac{1}{1-\alpha} \cdot \frac{\overline{\delta x}\overline{\delta x}^T}{\overline{\delta x}^T\overline{\delta x}}(I - \overline{H}J)\right]\overline{\delta x} \\ &= \overline{\delta x} + \frac{\alpha}{1-\alpha}\overline{\delta x} = \frac{1}{1-\alpha}\overline{\delta x} \\ &= \delta x. \end{aligned} \tag{4.56}$$

Hence, the descent approximation δx is available via this update too. Therefore, we call the above update *descent update*.

Remark 4.20 The descent update is based on an adjoint tangent evaluation, namely

$$\overline{\delta x}^T\overline{H}J.$$

This evaluation is efficiently available via the reverse mode of AD. □

For given $H_0 \in \mathbb{R}^{n\times n}$ and $x_0 \in \mathcal{D}$ let us consider an iterative application of the descent update,

$$\begin{aligned} \overline{H} \rightarrowtail H_{l-1}, \quad H \rightarrowtail H_l, \\ x \rightarrowtail x_l, \quad F(x) \rightarrowtail F_l := F(x_l), \quad J \rightarrowtail J_l := F'(x_l) \\ \overline{\delta x} \rightarrowtail \overline{\delta x}_l = -H_{l-1}F_l, \quad \alpha \rightarrowtail \alpha_l = \frac{\overline{\delta x}_l^T(I - H_{l-1}J_l)\overline{\delta x}_l}{\overline{\delta x}_l^T\overline{\delta x}_l}, \end{aligned} \tag{4.57}$$

i.e.,

$$H_l = \left[I + \frac{1}{1-\alpha_l} \cdot \frac{\overline{\delta x}_l\overline{\delta x}_l^T}{\overline{\delta x}_l^T\overline{\delta x}_l}(I - H_{l-1}J_l)\right]H_{l-1}, \quad l \geqslant 1, \tag{4.58}$$

with an associated sequence of iterates $\{x_l\}$ constructed via

$$x_{l+1} = x_l + \delta x_l, \quad \delta x_l = -H_lF_l, \quad l \geqslant 0. \tag{4.59}$$

In this section we will show that under appropriate conditions on the function F, the matrix H_0 and the initial guess x_0 the sequence of iterates $\{x_l\}$ converges locally q-superlinearly to a solution x_* of $F(x) = 0$. Additionally, we will show that asymptotically the conditions (4.12a), i.e.,

$$\lim_{l\to\infty} \overline{\delta x}_l - \Delta x_l = 0 \qquad \text{and} \qquad \lim_{l\to\infty} \overline{\delta x}_l^T H_{l-1} J_l - \overline{\delta x}_l^T = 0 \tag{4.60}$$

are fulfilled which by Theorem 4.7 and Corollary 4.8 implies that

$$\lim_{l\to\infty} \angle(\delta x_l, \Delta x_l) = 0 \qquad \text{and} \qquad \lim_{l\to\infty} \angle\big(\delta x_l, -\operatorname{grad} T(x_l|P_l H_{l-1})^T\big) = 0$$

hold where Δx_l is the Newton correction at x_l and P_l given via $P_l := \frac{\overline{\delta x}_l \overline{\delta x}_l^T}{\overline{\delta x}_l^T \overline{\delta x}_l}$.

Due to these properties the descent update is well-suited to be combined with a globalization approach based on the APNLF. Such an approach will be addressed in the next section.

Remark 4.21 An intuitive approach how to obtain the descent update (4.58) is given by the following considerations. Let H_{l-1} be nonsingular. Since $\alpha_l \neq 1$ is assumed we obtain by means of the Matrix Determinant Lemma, see e.g. [7],

$$\det(H_l) = (1-\alpha_l)^{-1} \cdot \det(H_{l-1}) \neq 0.$$

Hence, H_l is also nonsingular. With $A_l := H_l^{-1}$ the Sherman-Morrison-Woodbury formula, see e.g. [9], yields

$$A_l = A_{l-1}\left[I - \frac{\overline{\delta x}_l \overline{\delta x}_l^T}{\overline{\delta x}_l^T \overline{\delta x}_l}(I - A_{l-1}^{-1} J_l)\right]. \tag{4.61}$$

By construction the matrix A_l fulfills the following adjoint based property

$$\overline{\delta x}_l^T A_{l-1}^{-1} A_l = \overline{\delta x}_l^T A_{l-1}^{-1} J_l. \tag{4.62}$$

In Definition 4.2 (APNLF), let $\overline{H}_l = A_{l-1}^{-1}$. Then,

$$\begin{aligned} \frac{d}{d\lambda} T(x_l + \lambda \delta x_l | P_l A_{l-1}^{-1})_{|\lambda=0} &= -\overline{\delta x}_l^T A_{l-1} J_l \delta x_l \\ &\overset{(4.62)}{=} \overline{\delta x}_l^T A_{l-1}^{-1} A_l \cdot A_l^{-1} F_l \\ &= -\overline{\delta x}_l^T \overline{\delta x}_l = -\|\overline{\delta x}_l\|_2^2 < 0. \end{aligned} \tag{4.63}$$

Hence, the descent property of Proposition 4.4 is fulfilled. □

Remark 4.22 From the update formula (4.61) it is directly seen that A_l is affine covariant compatible if this property holds for A_{l-1}. Hence, for an affine covariant compatible choice of A_0 a globalization approach based on the APNLF combined with purifying updates and the descent update is affine covariant if also the step size control is held in affine covariant terms. This will be the case for our approach as it is seen from Subsection 4.4.6. □

For the upcoming local convergence results we will consider an iteration of the form (4.59) where the approximations H_l are defined via

$$H_{l+1} := \left[I + \frac{1}{1-\alpha_{l+1}} \cdot \frac{v_l v_l^T}{v_l^T v_l}(I - H_l J_{l+1})\right] H_l, \quad v_l \in \mathbb{R}^n \setminus \{0\}, \tag{4.64}$$

with

$$\alpha_{l+1} := \frac{v_l^T(I - H_l J_{l+1})v_l}{v_l^T v_l}.$$

Note that for $v_l = \overline{\delta x}_{l+1}$ we obtain the update (4.58). Inspired by the analysis in [8] our proof of convergence will be performed in two steps. First we will consider q-linear convergence then q-superlinear convergence. We will show q-linear convergence for all sequences of iterates which are based on updates like (4.64). Like in [8] this will be done by means of a so-called *bounded deterioration* property of the approximations H_l. For q-superlinear convergence we will show that an affine covariant version of the Dennis-Moré property, namely,

$$\lim_{l\to\infty} \frac{\|(I - H_l J_*)\delta x_l\|_2}{\|\delta x_l\|_2} = 0, \quad J_* := F'(x_*) \text{ where } F(x_*) = 0, \tag{4.65}$$

holds if the sequence $\{v_l\}$ satisfies an adapted version of the *residual property* which was originally introduced by Schlenkrich in [28]. As we will see this property is given for the choice $v_l = \overline{\delta x}_{l+1}$.

All following statements are held in affine covariant terms provided H_0 is affine covariant compatible and the sequence $\{v_l\}$ is affine covariant. This helps to verify that (4.60) holds.

Remark 4.23 In the following, every time we characterize quantities which depend on the approximations H_l by the terms *affine covariance* or *affine covariant* we mean that there is an affine covariant compatible choice of H_0 and an affine covariant choice of v_l such that these quantities feature affine covariance. □

In the course of the upcoming analysis we will suppose that the following assumption regarding the nonlinearity of F is valid.

Assumption 4.24 *In addition to Assumption 2.1 it holds that $F(x_*) = 0$ and $F'(x_*)$ is nonsingular for some $x_* \in \mathcal{D}$. Furthermore, there exists a nonnegative constant $\omega < \infty$ such that the affine covariant Lipschitz condition*

$$\|F'(x_*)^{-1}\big(F'(x) - F'(x_*)\big)\|_2 \leqslant \omega\|x - x_*\|_2 \tag{4.66}$$

holds for all $x \in \mathcal{D}$.

This means that we will rely on a Lipschitz condition for the proof of local convergence. This does not fit into our overall concept of describing the nonlinearity of F in terms of nonlinearity bounds. But we believe that there is no appropriate nonlinearity bound that serves the purpose of providing linear convergence of the sequence $\{x_l\}$ if updates of the form (4.64) are considered.

4.3.1 Bounded deterioration and linear convergence

For given F which fulfills Assumption 4.24 we consider an iteration of the form

$$x_{l+1} = x_l + \delta x_l, \quad \delta x_l = -H_l F_l, \tag{4.67}$$

with $F_l := F(x_l)$ and where the sequence of inverse Jacobian approximations $\{H_l\}$ is recursively defined via (4.64). With $J_* := F'(x_*)$ and for the Frobenius norm $\|\cdot\|_F$ we will show that under appropriate conditions the estimate

$$\|(J_*^{-1} - H_{l+1})J_*\|_F \leqslant \|(J_*^{-1} - H_l)J_*\|_F + C\|x_{l+1} - x_*\|_2 \tag{4.68}$$

holds for some constant $C > 0$. This means that the deterioration w.r.t. J_*^{-1} of the next approximation H_{l+1} can be bounded by the current deterioration plus a quantity which is proportional to the norm of the error at the next iterate. To have such an estimate at hand is crucial for the techniques we will adapt from [8] to provide a result about local q-linear convergence of the sequence $\{x_l\}$ generated by (4.67).

In order to verify the above estimate we need the following auxiliary lemma.

Lemma 4.25 *Let $a, b \in \mathbb{R}^n$ and $E \in \mathbb{R}^{n\times n}$. Then,*

$$\|(I - ab^T)E\|_F^2 = \|E\|_F^2 - 2b^T EE^T a + \|a\|_2^2 \cdot \|b^T E\|_2^2.$$

Proof. With the trace $\operatorname{tr}(A) := \sum_{i=1}^n a_{ii}$ of a matrix $A = \{a_{ij}\}_{i,j=1,\dots,n} \in \mathbb{R}^{n\times n}$ one calculates

$$\begin{aligned}\|(I - ab^T)E\|_F^2 &= \operatorname{tr}\big[E^T(I - ba^T)(I - ab^T)E\big]\\ &= \operatorname{tr}(E^TE) - \operatorname{tr}(E^Tba^TE) - \operatorname{tr}(E^Tab^TE) + \operatorname{tr}(E^Tba^Tab^TE)\\ &= \|E\|_F^2 - 2\operatorname{tr}(E^Tba^TE) + \|a\|_2^2 \cdot \|b^TE\|_2^2\\ &= \|E\|_F^2 - 2b^TEE^Ta + \|a\|_2^2 \cdot \|b^TE\|_2^2.\end{aligned}$$

Hence the stated equality is true. ∎

Now we can prove (4.68).

Theorem 4.26 (Bounded deterioration) *Let F fulfill Assumption 4.24. For x_* and $x_{l+1} \in \mathcal{D}$ let $J_* := F'(x_*)$ and $J_{l+1} := F'(x_{l+1})$. For this x_{l+1} and for $H_l \in \mathbb{R}^{n\times n}$ assume that there exist positive constants δ and ζ such that*

$$\|E_l\|_2 := \|I - H_lJ_*\|_2 \leqslant \delta < \frac{1}{3}, \tag{4.69}$$

$$\|x_{l+1} - x_*\|_2 \leqslant \zeta \tag{4.70}$$

and

$$\beta := 1 - (\delta + (1+\delta)\omega\zeta) > 0. \tag{4.71}$$

Then for each $v_l \in \mathbb{R}^n \setminus \{0\}$ the matrix

$$H_{l+1} := \left[I + \frac{1}{1-\alpha_{l+1}} \cdot \frac{v_lv_l^T}{v_l^Tv_l}\big(I - H_lJ_{l+1}\big)\right] H_l$$

with

$$\alpha_{l+1} := \frac{v_l^T\big(I - H_lJ_{l+1}\big)v_l}{v_l^Tv_l}$$

is well defined and

$$\|E_{l+1}\|_F \leqslant \|E_l\|_F + \frac{(1+\delta)^2}{(1-\delta)\beta} \cdot \omega\|x_{l+1} - x_*\|_2. \tag{4.72}$$

Proof. The proof is technical. In order not to overload it with clumsy notation we abbreviate

$$\alpha_{l,*} := \frac{v_l^T\big(I - H_lJ_*\big)v_l}{v_l^Tv_l}, \quad e_l := x_l - x_*.$$

Then,

$$\begin{aligned}
E_{l+1} &= I - H_{l+1}J_* = I - \left[I + \frac{1}{(1-\alpha_{l+1})}\frac{v_l v_l^T}{v_l^T v_l}\big(I - H_l J_{l+1}\big)\right] H_l J_* \\
&= E_l - \frac{1}{(1-\alpha_{l,*})}\frac{v_l v_l^T}{v_l^T v_l} H_l J_* (I - H_l J_*) \\
&\quad + \frac{\alpha_{l,*} - \alpha_{l+1}}{(1-\alpha_{l,*})(1-\alpha_{l+1})}\frac{v_l v_l^T}{v_l^T v_l} H_l J_* (I - H_l J_*) \\
&\quad + \frac{1}{(1-\alpha_{l+1})}\frac{v_l v_l^T}{v_l^T v_l} H_l (J_{l+1} - J_*) H_l J_* \\
&= \left[I - \frac{1}{(1-\alpha_{l,*})}\frac{v_l v_l^T}{v_l^T v_l} H_l J_*\right] E_l \\
&\quad + \frac{1}{(1-\alpha_{l,*})(1-\alpha_{l+1})}\frac{v_l^T H_l J_* J_*^{-1}(J_{l+1} - J_*) v_l}{v_l^T v_l}\frac{v_l v_l^T}{v_l^T v_l} H_l J_* E_l \\
&\quad + \frac{1}{(1-\alpha_{l+1})}\frac{v_l v_l^T}{v_l^T v_l} H_l J_* J_*^{-1}(J_{l+1} - J_*) H_l J_*.
\end{aligned} \tag{4.73}$$

By the Lipschitz condition (4.66) of Assumption 3.16 and the bounds given in (4.69) and (4.70) we obtain for $\alpha_{l,*}$ and α_{l+1} the bounds

$$\alpha_{l,*} \leqslant \|E_l\|_2 \leqslant \delta,$$

and

$$\alpha_{l+1} \leqslant \|E_l\|_2 + (1 + \|E_l\|_2)\omega\|e_{l+1}\|_2 \leqslant \delta + (1+\delta)\omega\zeta,$$

respectively. Thus,

$$1 - \alpha_{l,*} \geqslant 1 - \delta > 0$$

and with the definition (4.71) of β one has

$$1 - \alpha_{l+1} \geqslant 1 - (\delta + (1+\delta)\omega\zeta) = \beta > 0. \tag{4.74}$$

Applying norms to (4.73) and using the assumed and derived bounds along with exploiting the Lipschitz condition (4.66) yields

$$\begin{aligned}
\|E_{l+1}\|_F &\leqslant \left\|\left[I - \frac{1}{(1-\alpha_{l,*})}\frac{v_l v_l^T}{v_l^T v_l} H_l J_*\right] E_l\right\|_F \\
&\quad + \frac{(1+\delta)^2\delta}{(1-\delta)\beta}\cdot\omega\|e_{l+1}\|_2 + \frac{(1+\delta)^2}{\beta}\cdot\omega\|e_{l+1}\|_2 \\
&= \left\|\left[I - \frac{1}{(1-\alpha_{l,*})}\frac{v_l v_l^T}{v_l^T v_l} H_l J_*\right] E_l\right\|_F + \frac{(1+\delta)^2}{(1-\delta)\beta}\cdot\omega\|e_{l+1}\|_2.
\end{aligned} \tag{4.75}$$

To prove (4.72), it only remains to show that

$$\left\|\left[I - \frac{1}{(1-\alpha_{l,*})}\frac{v_l v_l^T}{v_l^T v_l} H_l J_*\right] E_l\right\|_F \leqslant \|E_l\|_F. \tag{4.76}$$

Applying Lemma 4.25 to the square of the left hand side of the above claimed inequality leads to

$$\left\|\left[I - \frac{1}{(1-\alpha_{l,*})}\frac{v_l v_l^T}{v_l^T v_l} H_l J_*\right] E_l\right\|_F^2 = \|E_l\|_F^2 - 2\frac{v_l^T H_l J_* E_l E_l^T v_l}{(1-\alpha_{l,*})v_l^T v_l} + \frac{v_l^T H_l J_* E_l E_l^T J_*^T H_l^T v_l}{(1-\alpha_{l,*})^2 v_l^T v_l}. \tag{4.77}$$

A closer look at the last two terms on the right hand side reveals that

$$\begin{aligned}
-2\frac{v_l^T H_l J_* E_l E_l^T v_l}{(1-\alpha_{l,*})v_l^T v_l} &+ \frac{v_l^T H_l J_* E_l E_l^T J_*^T H_l^T v_l}{(1-\alpha_{l,*})^2 v_l^T v_l} \\
&= -2\frac{v_l^T E_l E_l^T v_l}{v_l^T v_l} + 2\frac{v_l^T (I - \frac{1}{1-\alpha_{l,*}} H_l J_*) E_l E_l^T v_l}{v_l^T v_l} \\
&\quad + \frac{v_l^T E_l E_l^T v_l}{v_l^T v_l} - \frac{v_l^T (I - \frac{1}{1-\alpha_{l,*}} H_l J_*) E_l E_l^T v_l}{v_l^T v_l} \\
&\quad - \frac{\frac{1}{1-\alpha_{l,*}} v_l^T H_l J_* E_l E_l^T (I - \frac{1}{1-\alpha_{l,*}} J_*^T H_l^T) v_l}{v_l^T v_l} \\
&= -\frac{v_l^T E_l E_l^T v_l}{v_l^T v_l} + \frac{v_l^T (I - \frac{1}{1-\alpha_{l,*}} H_l J_*) E_l E_l^T v_l}{v_l^T v_l} \\
&\quad - \frac{\frac{1}{1-\alpha_{l,*}} v_l^T H_l J_* E_l E_l^T (I - \frac{1}{1-\alpha_{l,*}} J_*^T H_l^T) v_l}{v_l^T v_l} \\
&= -\frac{v_l^T E_l E_l^T v_l}{v_l^T v_l} + \frac{v_l^T (I - \frac{1}{1-\alpha_{l,*}} H_l J_*) E_l E_l^T v_l}{v_l^T v_l} \\
&\quad - \frac{v_l^T E_l E_l^T (I - \frac{1}{1-\alpha_{l,*}} J_*^T H_l^T) v_l}{v_l^T v_l} \\
&\quad + \frac{v_l^T (I - \frac{1}{1-\alpha_{l,*}} H_l J_*) E_l E_l^T (I - \frac{1}{1-\alpha_{l,*}} J_*^T H_l^T) v_l}{v_l^T v_l} \\
&= -\frac{v_l^T E_l E_l^T v_l}{v_l^T v_l} \\
&\quad + \frac{v_l^T (I - \frac{1}{1-\alpha_{l,*}} H_l J_*) E_l E_l^T (I - \frac{1}{1-\alpha_{l,*}} J_*^T H_l^T) v_l}{v_l^T v_l} \\
&= -\frac{\|v_l^T E_l\|_2^2}{\|v_l\|_2^2} + \frac{\|v_l^T (I - \frac{1}{1-\alpha_{l,*}} H_l J_*) E_l\|_2^2}{\|v_l\|_2^2}.
\end{aligned}$$

Applying the transformation

$$\left(I - \frac{1}{1-\alpha_{l,*}} H_l J_*\right) E_l = \frac{1}{1-\alpha_{l,*}} E_l (E_l - \alpha_{l,*} I)$$

to the second term yields

$$\begin{aligned}
\frac{\|v_l^T (I - \frac{1}{1-\alpha_{l,*}} H J_*) E_l\|_2^2}{\|v_l\|_2^2} &= \frac{\|\frac{1}{1-\alpha_{l,*}} v_l^T E_l (E_l - \alpha_{l,*} I)\|_2^2}{\|v_l\|_2^2} \\
&\leqslant \frac{1}{(1-\alpha_{l,*})^2} \cdot \frac{\|v_l^T E_l\|_2^2}{\|v_l\|_2^2} \cdot \|E_l - \alpha_{l,*} I\|_2^2 \\
&\leqslant \left(\frac{\alpha_{l,*} + \|E_l\|_2}{1-\alpha_{l,*}}\right)^2 \cdot \frac{\|v_l^T E_l\|_2^2}{\|v_l\|_2^2} \\
&\leqslant \frac{4\delta^2}{(1-\delta)^2} \cdot \frac{\|v_l^T E_l\|_2^2}{\|v_l\|_2^2}.
\end{aligned} \tag{4.78}$$

The bound (4.69) implies that $4\delta^2/(1-\delta)^2 < 1$. Inserting the derived results into (4.77) gives

$$\begin{aligned}\left\| \left[I - \frac{1}{1-\alpha_{l,*}} \frac{v_l v_l^T}{v_l^T v_l} H_l J_* \right] E_l \right\|_F^2 &= \|E_l\|_F^2 - \frac{\|v_l^T E_l\|_2^2}{\|v_l\|_2^2} + \frac{\|v_l^T (I - \frac{1}{1-\alpha_{l,*}} H_l J_*) E_l\|_2^2}{\|v_l\|_2^2} \\ &\leqslant \|E_l\|_F^2 - \left(1 - \frac{4\delta^2}{(1-\delta)^2}\right) \cdot \frac{\|v_l^T E_l\|_2^2}{\|v_l\|_2^2} \\ &\leqslant \|E_l\|_F^2 \end{aligned} \tag{4.79}$$

providing evidence of (4.76) and thus concluding the proof.

■

Next, we will state our linear convergence result. It is an affine covariant adaption of Theorem 3.2 in [8] tailored to the update (4.64) and the associated bounded deterioration property (4.72). Though not explicitly stated in [8] the basic idea of the proof is to consider one step of the iteration (4.67) at x_l as a perturbed version of

$$x_{l+1} = x_l - J_*^{-1} F_l. \tag{4.80}$$

It is well-known that an iteration based on this recursion converges locally q-quadratically to x_*. By means of the bounded deterioration property at least q-linear convergence of the perturbed iteration (4.67) can be guaranteed.

Theorem 4.27 (Linear convergence) *Suppose Assumption 4.24 holds for F. Let $J_* := F'(x_*)$. For $H_0 \in \mathbb{R}^{n\times n}$ and $x_0 \in \mathcal{D}$ consider the iteration*

$$x_{l+1} = x_l + \delta x_l, \quad \delta x_l = -H_l F_l, \quad F_l := F(x_l), \tag{4.81}$$

where

$$H_{l+1} := \left[I + \frac{1}{1-\alpha_{l+1}} \cdot \frac{v_l v_l^T}{v_l^T v_l} (I - H_l J_{l+1}) \right] H_l, \quad J_{l+1} := F'(x_{l+1}), \tag{4.82a}$$

with $v_l \in \mathbb{R}^n \setminus \{0\}$ and

$$\alpha_{l+1} := \frac{v_l^T (I - H_l J_{l+1}) v_l}{v_l^T v_l}. \tag{4.82b}$$

Then for $r \in (0,1)$ there exist positive constants $\varepsilon(r)$ and $\rho(r)$ such that for

$$\begin{aligned} \|x_0 - x_*\|_2 &\leqslant \varepsilon(r), \\ \|E_0\|_F := \|I - H_0 J_*\|_F &\leqslant \rho(r) < \frac{1}{6} \end{aligned} \tag{4.83}$$

the sequence $\{x_l\}$ is well defined, converges q-linearly to x_* with

$$\|x_{l+1} - x_*\|_2 \leqslant r \|x_l - x_*\|_2$$

for $l \geqslant 0$, and $\{\|H_l J_*\|_2\}$, $\{\|J_*^{-1} H_l^{-1}\|_2\}$ are uniformly bounded.

Proof. Let $r \in (0,1)$ be given and abbreviate $\varepsilon = \varepsilon(r)$. Define with the Lipschitz constant ω introduced in (4.66) the quantities

$$\begin{aligned} \delta = \delta(r) &:= 2\rho(r) \\ \beta &:= 1 - (\delta + (1+\delta) \cdot \omega r \varepsilon). \end{aligned}$$

Choose δ and ε such that the following inequalities hold

$$\delta + (1+\delta)\cdot\frac{\omega}{2}\varepsilon \leqslant r, \tag{4.84}$$

$$\beta > 0, \tag{4.85}$$

$$\frac{(1+\delta)^2}{(1-\delta)\beta}\cdot\frac{r}{1-r}\cdot\omega\varepsilon \leqslant \frac{\delta}{2}. \tag{4.86}$$

If necessary further restrict ε so that the Ball $B_* := \{x \in \mathbb{R}^n \mid \|x - x_*\|_2 \leqslant \varepsilon\}$ is contained in $\mathcal{D}$. For ease of writing we use $e_l := x_l - x_*$. In accordance with E_0 we let $E_l := I - H_l J_*$, $l \geqslant 1$. Since $\|E_0\|_F \leqslant \rho(r)$ we have $\|E_0\|_2 \leqslant \|E_0\|_F \leqslant \rho(r) < \delta$. From (4.83) and the definition of δ it follows that $\delta < \frac{1}{3}$. Hence,

$$\|H_0 J_*\|_2 \leqslant 1 + \delta < \frac{4}{3}.$$

Furthermore, by the Perturbation Lemma we obtain

$$\|J_*^{-1} H_0^{-1}\|_2 \leqslant \frac{1}{1-\delta} < \frac{3}{2}.$$

As already stated we consider one step of the iteration (4.81) as a perturbation of (4.80). Thus, the first step can be written as

$$x_1 = x_0 + \delta x_0 = x_0 - J_*^{-1}F_0 + (J_*^{-1} - H_0)F_0$$

and accordingly for the error e_1 we get

$$\begin{aligned} e_1 &= e_0 - J_*^{-1}F_0 + \big(J_*^{-1} - H_0\big)F_0 \\ &= -\int_0^1 J_*^{-1}\big(F'(x_* + se_0) - J_*\big)e_0 ds \\ &\quad + \big(I - H_0 J_*\big)e_0 + \big(I - H_0 J_*\big)\int_0^1 J_*^{-1}\big(F'(x_* + se_0) - J_*\big)e_0 ds \\ &= \big(I - H_0 J_*\big)e_0 - H_0 J_* \int_0^1 J_*^{-1}\big(F'(x_* + se_0) - J_*\big)e_0 ds. \end{aligned} \tag{4.87}$$

With the Lipschitz condition (4.66) and the upper bound (4.84) we obtain

$$\begin{aligned} \|e_1\|_2 &\leqslant \Big[\|E_0\|_2 + (1 + \|E_0\|_2)\frac{\omega}{2}\|e_0\|_2\Big]\,\|e_0\|_2 \\ &\leqslant \Big[\delta + (1+\delta)\cdot\frac{\omega}{2}\varepsilon\Big]\,\|e_0\|_2 \\ &\leqslant r\|e_0\|_2 \leqslant r\varepsilon < \varepsilon. \end{aligned} \tag{4.88}$$

Thus, $x_1 \in B_*$ which implies that $x_1 \in \mathcal{D}$. The remaining part of the proof is done via an induction argument. Assume that $\|E_m\|_F \leqslant \delta < \frac{1}{3}$ and $\|e_{m+1}\|_2 \leqslant r\|e_m\|_2 \leqslant r\varepsilon$ for $m = 0, \ldots, l-1$, $l \geqslant 1$. Hence, by (4.85) Theorem 4.26 is applicable for each m with the choice $\zeta := r\varepsilon$. Therefore,

$$\|E_{m+1}\|_F - \|E_m\|_F \leqslant \frac{(1+\delta)^2}{(1-\delta)\beta}\cdot\omega\|e_{m+1}\|_2 \leqslant \frac{(1+\delta)^2}{(1-\delta)\beta}\cdot r^{m+1}\omega\varepsilon \quad \forall m.$$

Summing over the indices m leads to

$$\|E_l\|_F \leqslant \|E_0\|_F + \frac{(1+\delta)^2}{(1-\delta)\beta}\cdot\frac{r}{1-r}\cdot\omega\varepsilon.$$

And by virtue of (4.83) and (4.86) we have

$$\|E_l\|_F \leqslant \frac{\delta}{2} + \frac{\delta}{2} = \delta.$$

In analogy to the case $l = 0$ it directly follows that

$$\|H_l J_*\|_2 \leqslant 1 + \delta < \frac{4}{3}. \tag{4.89}$$

Also, $\delta < \frac{1}{3}$ in conjunction with the Perturbation Lemma yields

$$\|J_*^{-1} H_l^{-1}\|_2 \leqslant \frac{1}{1-\delta} < \frac{3}{2}. \tag{4.90}$$

Now proceeding as in (4.87) and (4.88), replacing 0 and 1 by l and $l+1$, leads to

$$\|e_{l+1}\|_2 \leqslant r\|e_l\|_2.$$

This shows q-linear convergence.

■

4.3.2 Superlinear convergence

In the previous section we showed for F fulfilling Assumption 4.24 and by means of the bounded deterioration property (4.72) from Theorem 4.26 that the iteration (4.81) with the update (4.82) converges q-linearly to a solution x_* of $F(x) = 0$. In order to prove superlinear convergence we will adapt techniques from [28] and [9] in such a way that they fit into our affine covariant framework. The key to superlinear convergence is to show that with $J_* := F'(x_*)$ the relation

$$\lim_{l\to\infty} \frac{\|(I - H_l J_*)\delta x_l\|_2}{\|\delta x_l\|_2} = 0 \tag{4.91}$$

holds which is an affine covariant version of the standard Dennis-Moré property from [9]. Our proof to show that (4.91) is true is based on the assumption that the vectors v_l in the update formula (4.82) fulfill the residual property (4.102). We will show that the sequence $\{v_l\}$ with $v_l := -H_l F_{l+1} = \overline{\delta x}_{l+1}$ satisfies this property. Subsequently, we will discuss the r-order of convergence for all superlinear convergent sequences $\{x_l\}$ generated by (4.81) where the sequence $\{v_l\}$ fulfills the residual property.

By means of (4.91) we will show that for the descent update (4.58) asymptotically the correction $\overline{\delta x}_l$ and the Newton correction Δx_l coincide which is just the left statement of (4.60). To prove the second statement in (4.60), i.e.,

$$\lim_{l\to\infty} \overline{\delta x}_l^T H_{l-1} J_l - \overline{\delta x}_l^T = 0$$

we will exploit an affine covariant adaption of Schlenkrich's *transposed Dennis-Moré property*, i.e.,

$$\lim_{l\to\infty} \frac{\|v_l^T (I - H_l J_*)\|_2}{\|v_l\|_2} = 0. \tag{4.92}$$

The original property was introduced in [28] and does not feature any invariance property. We make use of the above affine covariant property also for our proof of (4.91).

The following theorem shows that (4.92) is indeed true under appropriate conditions.

Theorem 4.28 (Transposed Dennis-Moré series and property) *Suppose Assumption 4.24 holds for F and let $J_* := F'(x_*)$. Consider the iteration (4.81) with the update (4.82). If $\{x_l\}$ is well defined and converges to x_* with $\sum_{l=0}^{\infty}\|x_l - x_*\|_2 < \infty$ and if positive constants δ and ζ exist such that for all l the bounds (4.69) and (4.70) hold so that (4.71) is true, then the transposed Dennis-Moré series is bounded, i.e.,*

$$\sum_{l=0}^{\infty} \frac{\|v_l^T\big(I - H_l J_*\big)\|_2}{\|v_l\|_2} < \infty \tag{4.93}$$

and consequently the transposed Dennis-Moré property

$$\lim_{l\to\infty} \frac{\|v_l^T\big(I - H_l J_*\big)\|_2}{\|v_l\|_2} = 0 \tag{4.94}$$

holds.

Remark 4.29 Note that if the conditions of Theorem 4.27 hold then also the conditions of the above theorem are fulfilled. □

Proof. We use the notation as introduced in Theorem 4.26. By the estimate (4.75) we have for all l,

$$\|E_{l+1}\|_F \leqslant \left\| \left[I - \frac{1}{(1-\alpha_{l,*})} \cdot \frac{v_l v_l^T}{v_l^T v_l} H_l J_* \right] E_l \right\|_F + \frac{(1+\delta)^2}{(1-\delta)\beta} \cdot \omega \|e_{l+1}\|_2. \tag{4.95}$$

Recalling (4.79) we know that

$$\left\| \left[I - \frac{1}{(1-\alpha_{l,*})} \cdot \frac{v_l v_l^T}{v_l^T v_l} H_l J_* \right] E_l \right\|_F^2 \leqslant \|E_l\|_F^2 - \left(1 - \frac{4\delta^2}{(1-\delta)^2}\right) \cdot \frac{\|v_l^T E_l\|_2^2}{\|v_l\|_2^2} \tag{4.96a}$$

and

$$\left\| \left[I - \frac{1}{(1-\alpha_{l,*})} \cdot \frac{v_l v_l^T}{v_l^T v_l} H_l J_* \right] E_l \right\|_F \leqslant \|E_l\|_F. \tag{4.96b}$$

For arbitrary $M \in \mathbb{R}^{n\times n}$ it holds that $\|M\|_F \leqslant \sqrt{n}\|M\|_2$. Hence, from (4.69) it follows that

$$\|E_l\|_F \leqslant \sqrt{n}\delta =: \hat{\delta}. \tag{4.96c}$$

Squaring (4.95) and applying the bounds from (4.96) we get

$$\begin{aligned} \|E_{l+1}\|_F^2 \leqslant \|E_l\|_F^2 &- \left(1 - \frac{4\delta^2}{(1-\delta)^2}\right) \cdot \frac{\|v_l^T E_l\|_2^2}{\|v_l\|_2^2} \\ &+ 2\hat{\delta}\frac{(1+\delta)^2}{(1-\delta)\beta} \cdot \omega \|e_{l+1}\|_2 + \left[\frac{(1+\delta)^2}{(1-\delta)\beta} \cdot \omega\right]^2 \cdot \|e_{l+1}\|_2^2. \end{aligned} \tag{4.97}$$

For convenience we define the following constants

$$C_1 := 1 - \frac{4\delta^2}{(1-\delta)^2}, \quad C_2 := 2\hat{\delta}\frac{(1+\delta)^2}{(1-\delta)\beta} \cdot \omega, \quad C_3 := \left[\frac{(1+\delta)^2}{(1-\delta)\beta} \cdot \omega\right]^2.$$

Note that by the assumptions we have $\delta < \frac{1}{3}$ and hence $C_1 > 0$. With these constants rearranging of (4.97) gives

$$C_1 \frac{\|v_l^T E_l\|_2^2}{\|v_l\|_2^2} \leqslant \|E_l\|_F^2 - \|E_{l+1}\|_F^2 + C_2\|e_{l+1}\|_2 + C_3\|e_{l+1}\|_2^2.$$

Summing over the indices l leads to

$$\sum_{l=0}^{k} \frac{\|v_l^T E_l\|_2^2}{\|v_l\|_2^2} \leqslant \frac{1}{C_1}\left(\|E_0\|_F^2 - \|E_{k+1}\|_F^2\right) + \frac{C_2}{C_1}\sum_{l=0}^{k}\|e_{l+1}\|_2 + \frac{C_3}{C_2}\sum_{l=0}^{k}\|e_{l+1}\|_2^2. \tag{4.98}$$

By (4.96c) we have $\|E_0\|_F^2 \leqslant \hat{\delta}^2$. Dropping the negative term on the right hand side of (4.98) and for $k \to \infty$ we obtain

$$\sum_{l=0}^{\infty} \frac{\|v_l^T E_l\|_2^2}{\|v_l\|_2^2} \leqslant \frac{\hat{\delta}^2}{C_1} + \frac{C_2}{C_1}\sum_{l=0}^{\infty}\|e_{l+1}\|_2 + \frac{C_3}{C_2}\sum_{l=0}^{\infty}\|e_{l+1}\|_2^2. \tag{4.99}$$

Since the right hand side is assumed to be bounded this implies

$$\sum_{l=0}^{\infty} \frac{\|v_l^T E_l\|_2}{\|v_l\|_2} < \infty$$

and therefore

$$\lim_{l\to\infty} \frac{\|v_l^T E_l\|_2}{\|v_l\|_2} = 0.$$

This concludes the proof.

■

As a direct consequence of this result we obtain for the descent update:

Corollary 4.30 *Under the assumptions and with the notation of Theorem 4.28 assume that $x_l \neq x_* \ \forall l$. Suppose that for the update* (4.82) *the vectors v_l are chosen via $v_l := \overline{\delta x}_{l+1} = -H_l F_{l+1}$, $F_{l+1} := F(x_{l+1})$, i.e., the descent update* (4.58) *is employed for the iteration* (4.81)*. Consider the APNLF from Definition 4.2 for $\overline{H}_l := H_{l-1}$ and let $g_l := -\operatorname{grad} T(x_l|P_l H_{l-1})^T$, i.e., $g_l = \left(\overline{\delta x}_l^T H_{l-1} J_l\right)^T$ where $J_l := F'(x_l)$.*

Then,

$$\lim_{l\to\infty}\left|1 - \frac{\|g_l\|_2}{\|\overline{\delta x}_l\|_2}\right| = 0 \qquad \textit{and} \qquad \lim_{l\to\infty}\left\|\frac{g_l}{\|g_l\|_2} - \frac{\overline{\delta x}_l}{\|\overline{\delta x}_l\|_2}\right\|_2 = 0. \tag{4.100}$$

The above result shows that asymptotically $\overline{\delta x}_l$ is equal to the transposed negative gradient of the APNLF in length and direction. Hence, the second statement of (4.60) is fulfilled.

Proof. We first show that

$$\lim_{l\to\infty} \frac{\|v_l^T(I - H_l J_{l+1})\|_2}{\|v_l\|_2} = 0 \tag{4.101}$$

is true. By the assumptions (4.69) holds and hence $\|H_l J_*\|_2 \leqslant 1 + \delta < \frac{4}{3}\ \forall l$. Thus,

$$\begin{aligned}
\frac{\|v_l^T(I - H_l J_{l+1})\|_2}{\|v_l\|_2} &\leqslant \frac{\|v_l^T(I - H_l J_*)\|_2}{\|v_l\|_2} + \frac{\|v_l^T H_l J_*(I - J_*^{-1} J_{l+1})\|_2}{\|v_l\|_2} \\
&\leqslant \frac{\|v_l^T(I - H_l J_*)\|_2}{\|v_l\|_2} + \|H_l J_*\|_2 \cdot \|I - J_*^{-1} J_{l+1}\|_2 \\
&\leqslant \frac{\|v_l^T(I - H_l J_*)\|_2}{\|v_l\|_2} + \frac{4}{3}\cdot\|I - J_*^{-1} J_{l+1}\|_2.
\end{aligned}$$

Since $\{x_l\} \to x_*$ and the transposed Dennis-Moré property (4.94) holds, (4.101) is verified. Now we can prove the first statement of (4.100).

$$\left|1 - \frac{\|g_l\|_2}{\|\overline{\delta x}_l\|_2}\right| = \frac{|\|\overline{\delta x}_l\|_2 - \|g_l\|_2|}{\|\overline{\delta x}_l\|_2} \leqslant \frac{\|\overline{\delta x}_l - g_l\|_2}{\|\overline{\delta x}_l\|_2} = \frac{\|\overline{\delta x}_l^T(I - H_{l-1}J_l)\|_2}{\|\overline{\delta x}_l\|_2}.$$

Since $\overline{\delta x}_l = v_{l-1}$ we obtain by means of (4.101) the desired result

$$\lim_{l\to\infty} \left| 1 - \frac{\|g_l\|_2}{\|\overline{\delta x}_l\|_2} \right| = 0.$$

Regarding the second statement of (4.100) we use the fact that

$$\begin{aligned}
\left\| \frac{g_l}{\|g_l\|_2} - \frac{\overline{\delta x}_l}{\|\overline{\delta x}_l\|_2} \right\|_2 &= \left\| \left(\frac{\|\overline{\delta x}_l\|_2}{\|g_l\|_2} - 1 + 1 \right) \cdot \frac{g_l}{\|\overline{\delta x}_l\|_2} - \frac{\overline{\delta x}_l}{\|\overline{\delta x}_l\|_2} \right\|_2 \\
&\leqslant \left| \frac{\|\overline{\delta x}_l\|_2}{\|g_l\|_2} - 1 \right| \cdot \frac{\|g_l\|_2}{\|\overline{\delta x}_l\|_2} + \frac{\|\overline{\delta x}_l - g_l\|_2}{\|\overline{\delta x}_l\|_2} \\
&= \left| 1 - \frac{\|g_l\|_2}{\|\overline{\delta x}_l\|_2} \right| + \frac{\|\overline{\delta x}_l^T (I - H_{l-1} J_l)\|_2}{\|\overline{\delta x}_l\|_2} \\
&\leqslant 2 \cdot \frac{\|\overline{\delta x}_l^T (I - H_{l-1} J_l)\|_2}{\|\overline{\delta x}_l\|_2}.
\end{aligned}$$

Thus, (4.101) shows that also the second statement of (4.100) is true.

■

We will also employ the transposed Dennis-Moré property to show that the affine covariant Dennis-Moré property (4.91) holds. Therefore, we require the sequence $\{v_l\}$ to satisfy the following assumption:

Assumption 4.31 (Affine covariant residual property) *For the vectors $\{v_l\}$ in the update (4.82) there is a sequence $\{\xi_l\} \subset \mathbb{R} \setminus \{0\}$ such that with $J_* := F'(x_*)$ and the corrections δx_l given in (4.81) one has*

$$\lim_{l\to\infty} \frac{\|\xi_l v_l - \big(I - H_l J_*\big)\delta x_l\|_2}{\|\delta x_l\|_2} = 0 \tag{4.102a}$$

or equivalently

$$\begin{aligned}
&\xi_l v_l = \big(I - H_l J_*\big)\delta x_l + r_l \\
&\text{with } \|r_l\|_2 \leqslant c_l \|\delta x_l\|_2 \text{ and } \lim_{l\to\infty} c_l = 0.
\end{aligned} \tag{4.102b}$$

Furthermore, for our proof of superlinear convergence we will need an affine covariant modification of Lemma 4.1.16 in [9].

Lemma 4.32 *Suppose Assumption 4.24 holds for F and let $J_* := F'(x_*)$. Then there exist constants $\gamma > 0$, $0 < \kappa_1 < \kappa_2$, such that*

$$\kappa_1 \|y - x\|_2 \leqslant \|J_*^{-1}\big(F(y) - F(x)\big)\|_2 \leqslant \kappa_2 \|y - x\|_2 \tag{4.103}$$

for all x, $y \in \mathcal{D}$ for which $\max\{\|x - x_*\|_2, \|y - x_*\|_2\} \leqslant \gamma$.

Proof. The proof is basically along the lines of the proof of Lemma 4.1.16 in [9]. Note that by the Lipschitz condition (4.66) it holds that

$$\|J_*^{-1}\big(F(y) - F(x) - J_*(y - x)\big)\|_2 \leqslant \frac{\omega}{2}(\|x - x_*\|_2 + \|y - x_*\|_2)\|y - x\|_2 \tag{4.104}$$

for all x, $y \in \mathcal{D}$. Hence,

$$\begin{aligned}\|J_*^{-1}\big(F(y) - F(x)\big)\|_2 &\leqslant \|y - x\|_2 + \|J_*^{-1}\big(F(y) - F(x) - J_*(y - x)\big)\|_2 \\ &\leqslant \big[1 + \frac{\omega}{2}(\|x - x_*\|_2 + \|y - x_*\|_2)\big] \cdot \|y - x\|_2 \\ &\leqslant [1 + \omega\gamma] \cdot \|y - x\|_2\end{aligned}$$

and therefore κ_2 in (4.103) can be chosen as $\kappa_2 = 1 + \omega\gamma$. Furthermore,

$$\begin{aligned}\|J_*^{-1}\big(F(y) - F(x)\big)\|_2 &\geqslant \|y - x\|_2 - \|J_*^{-1}\big(F(y) - F(x) - J_*(y - x)\big)\|_2 \\ &\geqslant \big[1 - \frac{\omega}{2}(\|x - x_*\|_2 + \|y - x_*\|_2)\big] \cdot \|y - x\|_2 \\ &\geqslant [1 - \omega\gamma] \cdot \|y - x\|_2.\end{aligned}$$

Thus, if $\gamma < \frac{1}{\omega}$ the first part of (4.103) holds true with $\kappa_1 = 1 - \omega\gamma > 0$.

■

Now we can prove superlinear convergence.

Theorem 4.33 (Superlinear convergence) *Suppose Assumption 4.24 holds for F and let $J_* := F'(x_*)$. Let the sequence of iterates $\{x_l\}$ generated by (4.81) with the update (4.82) be well defined and satisfy $\lim_{l\to\infty} x_l = x_*$ with $x_l \neq x_*$ for all l. Assume that the transposed Dennis-Moré property (4.94) is valid and that the sequence $\{v_l\}$ has the affine covariant residual property 4.31. Additionally, suppose that $\{\|H_l J_*\|_2\}$ and $\{\|J_*^{-1} H_l^{-1}\|_2\}$ are well defined and uniformly bounded. Then the affine covariant Dennis-Moré property holds, i.e.,*

$$\lim_{l\to\infty} \frac{\|\big(I - H_l J_*\big)\delta x_l\|_2}{\|\delta x_l\|_2} = 0 \tag{4.105}$$

which is equivalent to q-superlinear convergence of the iterates $\{x_l\}$ to x_*.

Proof. We split the proof into two parts I) and II). In the first part we will show that under the assumptions the transposed Dennis-Moré property (4.94) implies (4.105). To establish this result we will apply the techniques introduced in [28]. Let $e_l := x_l - x_*$. In the second part we will show that under the given assumptions it holds that

$$\lim_{l\to\infty} \frac{\|e_{l+1}\|_2}{\|e_l\|_2} = 0 \Leftrightarrow \lim_{l\to\infty} \frac{\|J_*^{-1} F(x_{l+1})\|_2}{\|\delta x_l\|_2} = 0 \Leftrightarrow \lim_{l\to\infty} \frac{\|(I - H_l J_*)\delta x_l\|_2}{\|\delta x_l\|_2} = 0.$$

The proof of the latter is completely held in affine covariant terms and is in part inspired by the proof of Theorem 8.2.4 in [9].

I) For $l \in \mathbb{N}$ we have by the affine covariant residual property (4.102b),

$$\begin{aligned}\|\xi_l v_l\|_2^2 &= \xi_l v_l^T\big((I - H_l J_*)\delta x_l + r_l\big) \\ &\leqslant |\xi_l| \cdot \|v_l^T(I - H_l J_*)\|_2 \|\delta x_l\|_2 + |\xi_l| c_l \|\delta x_l\|_2 \|v_l\|_2.\end{aligned}$$

Division by $|\xi_l| \cdot \|v_l\|_2 \|\delta x_l\|_2$ yields

$$|\xi_l| \frac{\|v_l\|_2}{\|\delta x_l\|_2} \leqslant \frac{\|v_l^T(I - H_l J_*)\|_2}{\|v_l\|_2} + c_l.$$

And again by (4.102b),

$$\frac{\|(I - H_l J_*)\delta x_l\|_2}{\|\delta x_l\|_2} \leqslant |\xi_l| \frac{\|v_l\|_2}{\|\delta x_l\|_2} + c_l.$$

Thus, together

$$\frac{\|(I - H_l J_*)\delta x_l\|_2}{\|\delta x_l\|_2} \leqslant \frac{\|v_l^T (I - H_l J_*)\|_2}{\|v_l\|_2} + 2c_l.$$

Hence, the transposed Dennis-Moré property (4.94) implies that

$$\lim_{l\to\infty} \frac{\|(I - H_l J_*)\delta x_l\|_2}{\|\delta x_l\|_2} = 0.$$

II) First, we will show that

$$\lim_{l\to\infty} \frac{\|J_*^{-1} F(x_{l+1})\|_2}{\|\delta x_l\|_2} = 0 \Leftrightarrow \lim_{l\to\infty} \frac{\|(I - H_l J_*)\delta x_l\|_2}{\|\delta x_l\|_2} = 0.$$

From (4.81) we obtain

$$\delta x_l + H_l F(x_l) = 0$$

and hence,

$$\begin{aligned} -H_l J_* J_*^{-1} F(x_{l+1}) = (I - H_l J_*)\delta x_l \\ + H_l J_* J_*^{-1}(F(x_l) - F(x_{l+1}) + J_* \delta x_l). \end{aligned} \tag{4.106}$$

Suppose $\lim_{l\to\infty} \|J_*^{-1} F(x_{l+1})\|_2 / \|\delta x_l\|_2 = 0$ is true. Rearrangement of (4.106) leads to

$$(I - H_l J_*)\delta x_l = -H_l J_* \big[J_*^{-1} F(x_{l+1}) + J_*^{-1}(F(x_l) - F(x_{l+1}) + J_* \delta x_l)\big].$$

By the assumptions there exist positive constants ψ and $\hat{\psi}$ such that for all l,

$$\|H_l J_*\|_2 \leqslant \psi \tag{4.107a}$$

$$\|J_*^{-1} H_l^{-1}\|_2 \leqslant \hat{\psi}. \tag{4.107b}$$

Introducing norms and applying the bounds (4.104) and (4.107a) yields

$$\frac{\|(I - H_l J_*)\delta x_l\|_2}{\|\delta x_l\|_2} \leqslant \psi \cdot \left[\frac{\|J_*^{-1} F(x_{l+1})\|_2}{\|\delta x_l\|_2} + \frac{\omega}{2}(\|e_l\|_2 + \|e_{l+1}\|_2)\right].$$

And since convergence is assumed one has

$$\lim_{l\to\infty} \frac{\|(I - H_l J_*)\delta x_l\|_2}{\|\delta x_l\|_2} = 0.$$

Now suppose $\lim_{l\to\infty} \|(I - H_l J_*)\delta x_l\|_2 / \|\delta x_l\|_2 = 0$ holds true. Suitable transformation of (4.106) gives

$$J_*^{-1} F(x_{l+1}) = -J_*^{-1} H_l^{-1}(I - H_l J_*)\delta x_l + J_*^{-1}(F(x_{l+1}) - F(x_l) - J_* \delta x_l). \tag{4.108}$$

Hence, by (4.104) and (4.107b),

$$\frac{\|J_*^{-1} F(x_{l+1})\|_2}{\|\delta x_l\|_2} \leqslant \hat{\psi} \cdot \frac{\|(I - H_l J_*)\delta x_l\|_2}{\|\delta x_l\|_2} + \frac{\omega}{2}(\|e_l\|_2 + \|e_{l+1}\|_2).$$

Since $\lim_{l\to\infty}\|e_l\|_2 \to 0$, we obtain

$$\lim_{l\to\infty}\frac{\|J_*^{-1}F(x_{l+1})\|_2}{\|\delta x_l\|_2} = 0.$$

Now we prove

$$\lim_{l\to\infty}\frac{\|e_{l+1}\|_2}{\|e_l\|_2} = 0 \Leftrightarrow \lim_{l\to\infty}\frac{\|J_*^{-1}F(x_{l+1})\|_2}{\|\delta x_l\|_2} = 0.$$

Assume that superlinear convergence holds true. Then there is an index l_0 such that for $\gamma > 0$ it holds that $\|e_{l+1}\|_2 \leqslant \gamma\ \forall l \geqslant l_0$. From Lemma 4.32 it follows that there is a $\kappa_2 = \kappa_2(\gamma) > 0$ such that

$$\|J_*^{-1}\big(F(x_{l+1}) - F(x_*)\big)\|_2 \leqslant \kappa_2\|e_{l+1}\|_2 \quad \forall l \geqslant l_0.$$

Additionally, we can assume l_0 to be large enough that also

$$\frac{\|e_{l+1}\|_2}{\|e_l\|_2} < 1$$

holds for all $l \geqslant l_0$. Therefore, by $F(x_*) = 0$ and the triangle inequality we obtain

$$\frac{\|J_*^{-1}F(x_{l+1})\|}{\|\delta x_l\|} \leqslant \frac{\kappa_2\|e_{l+1}\|}{\|e_l\| - \|e_{l+1}\|} \leqslant \frac{\kappa_2\frac{\|e_{l+1}\|}{\|e_l\|}}{1 - \frac{\|e_{l+1}\|}{\|e_l\|}}$$

and hence,

$$\lim_{l\to\infty}\frac{\|J_*^{-1}F(x_{l+1})\|_2}{\|\delta x_l\|_2} = 0.$$

We now assume $\lim_{l\to\infty}\frac{\|J_*^{-1}F(x_{l+1})\|_2}{\|\delta x_l\|_2} = 0$ to hold. From the assumed convergence of the iterates x_l to x_* and from Lemma 4.32 it follows that there exist $\kappa_1 > 0$, $l_0 \geqslant 0$ such that

$$\|J_*^{-1}\big(F(x_{l+1}) - F(x_*)\big)\|_2 \geqslant \kappa_1\|e_{l+1}\|_2 \quad \forall l \geqslant l_0,$$

and thus by the triangle inequality,

$$\frac{\|J_*^{-1}F(x_{l+1})\|_2}{\|\delta x_l\|_2} \geqslant \frac{\kappa_1\|e_{l+1}\|_2}{\|e_{l+1}\|_2 + \|e_l\|_2} = \kappa_1\frac{\frac{\|e_{l+1}\|_2}{\|e_l\|_2}}{1 + \frac{\|e_{l+1}\|_2}{\|e_l\|_2}}$$

from which we conclude

$$\lim_{l\to\infty}\frac{\|e_{l+1}\|_2}{\|e_l\|_2} = 0.$$

This completes the proof.

■

It remains to show that for the choice $v_l = \overline{\delta x}_{l+1}$ which characterizes the descent update (4.58) the sequence $\{v_l\}$ has the residual property from Assumption 4.31.

Proposition 4.34 *Suppose Assumption 4.24 holds for F. Let $J_* := F'(x_*)$. Consider for $H_l \in \mathbb{R}^{n\times n}$ and $x_l \in \mathcal{D}$ one step of the iteration* (4.81)*. Assume that $x_{l+1} \in \mathcal{D}$ and that for some positive constant ψ it holds that*

$$\|H_l J_*\|_2 \leqslant \psi. \tag{4.109}$$

Then for $v_l := \overline{\delta x}_{l+1} = -H_l F(x_{l+1})$, $F_{l+1} := F(x_{l+1})$, we obtain

$$\|v_l - \big(I - H_l J_*\big)\delta x_l\|_2 \leqslant \psi\frac{\omega}{2}(\|x_l - x_*\|_2 + \|x_{l+1} - x_*\|_2)\|\delta x_l\|_2.$$

Proof. We verify that

$$\begin{aligned}\|v_l - (I - H_l J_*)\delta x_l\|_2 &= \|H_l F_{l+1} + (I - H_l J_*)\delta x_l\|_2 \\ &= \|H_l F_{l+1} - H_l F_l - H_l J_* \delta x_l\|_2 \\ &= \|H_l J_* J_*^{-1}(F_{l+1} - F_l - J_* \delta x_l)\|_2\end{aligned}$$

which by (4.104) and (4.109) means that

$$\|v_l - (I - H_l J_*)\delta x_l\|_2 \leqslant \psi \frac{\omega}{2}(\|x_l - x_*\|_2 + \|x_{l+1} - x_*\|_2)\|\delta x_l\|_2.$$

■

Thus, by the other assumptions of Theorem 4.33 the sequence $\{v_l\}$ with $v_l = \overline{\delta x}_{l+1}$ fulfills the residual property from Assumption 4.31 with $\xi_l = 1\ \forall l$. This means that the affine covariant Dennis Moré property (4.105) is valid and the sequence $\{x_l\}$ constructed via (4.81) and the descent update (4.58) converges superlinearly to x_*.

Now we can prove that for the descent update asymptotically the corrections $\overline{\delta x}_l$ and Δx_l are the same. This property is a direct consequence of the transposed and the affine covariant Dennis-Moré property.

Corollary 4.35 *Let the assumptions and the notation from Theorem 4.33 with the choice* $v_l := \overline{\delta x}_{l+1} = -H_l F(x_{l+1})$, $F_{l+1} := F(x_{l+1})$ *be given. Additionally, assume that for each iterate* x_l *the corresponding Newton correction* $\Delta x_l := -J_l^{-1} F_l$, $J_l := F'(x_l)$ *is well defined. Then,*

$$\lim_{l\to\infty}\left|1 - \frac{\|\Delta x_l\|_2}{\|\overline{\delta x}_l\|_2}\right| = 0 \qquad \textit{and} \qquad \lim_{l\to\infty}\left\|\frac{\Delta x_l}{\|\Delta x_l\|_2} - \frac{\overline{\delta x}_l}{\|\overline{\delta x}_l\|_2}\right\|_2 = 0. \tag{4.110}$$

Analogously to the result of Corollary 4.30 this means that asymptotically the length and direction of $\overline{\delta x}_l$ and Δx_l coincide and therefore the first statement of (4.60) is valid.

Proof. First, we will show that the relations

$$\lim_{l\to\infty}\left|1 - \frac{\|\Delta x_l\|_2}{\|\delta x_l\|_2}\right| = 0 \qquad \text{and} \qquad \lim_{l\to\infty}\left\|\frac{\Delta x_l}{\|\Delta x_l\|_2} - \frac{\delta x_l}{\|\delta x_l\|_2}\right\|_2 = 0 \tag{4.111}$$

hold. This will be done by the same techniques we used in the proof of Corollary 4.30. So we will omit some details here. From the affine covariant Dennis-Moré property, the assumption that $\{H_l J_*\}$ is uniformly bounded, the assumed convergence and from

$$\frac{\|\big(I - H_l J_l\big)\delta x_l\|_2}{\|\delta x_l\|_2} \leqslant \frac{\|\big(I - H_l J_*\big)\delta x_l\|_2}{\|\delta x_l\|_2} + \|H_l J_*\|_2 \cdot \|I - J_*^{-1} J_l\|_2$$

it follows that

$$\lim_{l\to\infty}\frac{\|\big(I - H_l J_l\big)\delta x_l\|_2}{\|\delta x_l\|_2} = 0. \tag{4.112}$$

Furthermore,

$$\left|1 - \frac{\|\Delta x_l\|_2}{\|\delta x_l\|_2}\right| \leqslant \frac{\|\delta x_l - \Delta x_l\|_2}{\|\delta x_l\|_2} \leqslant \|J_l^{-1} J_*\|_2 \cdot \|J_*^{-1} H_l^{-1}\|_2 \cdot \frac{\|(I - H_l J_l)\delta x_l\|_2}{\|\delta x_l\|_2}.$$

By the assumptions the sequence $\{\|J_*^{-1}H_l^{-1}\|_2\}$ is uniformly bounded. Also, convergence is assumed. Hence, from (4.112) it follows that the first statement of (4.111) is true. Now we turn to prove the second statement. It holds that

$$\left\| \frac{\Delta x_l}{\|\Delta x_l\|_2} - \frac{\delta x_l}{\|\delta x_l\|_2} \right\|_2 = \left| 1 - \frac{\|\Delta x_l\|_2}{\|\delta x_l\|_2} \right| + \frac{\|\delta x_l - \Delta x_l\|_2}{\|\delta x_l\|_2}$$
$$\leqslant 2 \cdot \|J_l^{-1} J_*\|_2 \cdot \|J_*^{-1} H_l^{-1}\|_2 \cdot \frac{\|(I - H_l J_l)\delta x_l\|_2}{\|\delta x_l\|_2}$$

and hence the second statement of (4.111).

Now we show that $\overline{\delta x_l}$ and δx_l are asymptotically the same. Recall from (4.56) and (4.57) that

$$\delta x_l = \frac{1}{1 - \alpha_l} \overline{\delta x}_l.$$

Also,

$$|\alpha_l| = \frac{|\overline{\delta x}_l^T (I - H_{l-1} J_l) \overline{\delta x}_l|}{\overline{\delta x}_l^T \overline{\delta x}_l} \leqslant \frac{\|\overline{\delta x}_l^T (I - H_{l-1} J_l)\|_2}{\|\overline{\delta x}_l\|_2}.$$

Since $\{H_l J_*\}$ is uniformly bounded and the transposed Dennis-Moré property holds we obtain in analogy to the proof of (4.101) the result

$$\lim_{l \to \infty} = \frac{\|\overline{\delta x}_l^T (I - H_{l-1} J_l)\|_2}{\|\overline{\delta x}_l\|_2} = 0.$$

Hence,

$$\lim_{l \to \infty} \alpha_l = 0.$$

This completes the proof.

■

4.3.2.1 R-order of convergence

From the previous analysis we know that under appropriate conditions the sequence $\{x_l\}$ generated by (4.81) with the update (4.82) converges q-superlinearly to x_*. Here we will concretize the rate of convergence by determining the corresponding r-order of convergence. The basis for our analysis is provided by Theorem 4.2.19 from [28]. This theorem describes the r-order of convergence of a quasi-Newton method where the sequence of generated approximations $\{A_l\}$ of the Jacobians $F'(x_l)$ satisfies for some vector norm $\|\cdot\|$ the so-called *nonlinear heredity* property, i.e.,

$$\frac{\|(A_l - J_*)\delta x_j\|}{\|\delta x_j\|} \leqslant C \sum_{k=j}^{l} \|x_k - x_*\|, \quad 0 \leqslant j < l,\ C > 0,\ J_* := F'(x_*). \tag{4.113}$$

The statement of this theorem reads as follows.

Theorem 4.36 *Let $\|\cdot\|$ be some vector norm on $\mathbb{R}^n$. Suppose $F : \mathbb{R}^n \to \mathbb{R}^n$ and $x_* \in \mathbb{R}^n$ with $F(x_*) = 0$. Consider for an $x_0 \in \mathbb{R}^n$ and a sequence $\{A_l\} \subset \mathbb{R}^{n \times n}$ of nonsingular matrices the quasi-Newton iteration*

$$\delta x_l = -A_l^{-1} F(x_l)$$
$$x_{l+1} = x_l + \delta x_l.$$

Furthermore assume that F is Lipschitz continuously differentiable at x_* and the Jacobian J_* at x_* is nonsingular. If the sequence $\{x_l\}$ converges q-linearly to x_* and the sequence $\{A_l\}$ satisfies the nonlinear heredity property (4.113) *then the rate of convergence is q-superlinear with an r-order of*

$$\rho = \liminf_{l\to\infty} \sqrt[l]{|\ln\|x_l - x_*\||} \geqslant \rho_n, \tag{4.114}$$

where ρ_n is the positive root of $\rho^n(\rho - 1) = 1$.

The proof of this result is long and technical. Therefore, we do not state it here and instead refer to [28]. For a discussion of the derived r-order of convergence we refer once more to [28]. Since the nonlinear heredity property is not given in affine covariant terms Theorem 4.36 lacks this property too. Fortunately, careful study of the proof in [28] shows that Theorem 4.36 may be modified such that it fits into our affine covariant framework:

Theorem 4.37 *Let F fulfill Assumption 4.24 (with $\|\cdot\|_2$ substituted by some arbitrary vector norm $\|\cdot\|$ and the respective induced matrix norm) and let $J_* := F'(x_*)$. Consider for an $x_0 \in \mathcal{D}$ and a sequence $\{H_l\} \subset \mathbb{R}^{n\times n}$ of nonsingular matrices the quasi-Newton iteration*

$$\begin{aligned} \delta x_l &= -H_l F(x_l) \\ x_{l+1} &= x_l + \delta x_l. \end{aligned} \tag{4.115}$$

If

- *the sequence $\{x_l\}$ converges q-linearly to x_*,*
- *the sequence $\{H_l\}$ satisfies the affine covariant nonlinear heredity property*

$$\frac{\|(I - H_l J_*)\delta x_j\|}{\|\delta x_j\|} \leqslant C \sum_{k=j}^{l} \|x_k - x_*\|, \quad 0 \leqslant j < l,\ C > 0, \tag{4.116}$$

- *and additionally there exists a positive constant $\hat{\psi}$ such that $\|J_*^{-1} H_l^{-1}\| \leqslant \hat{\psi}\ \forall l$,*

then the rate of convergence is q-superlinear with an r-order given by (4.114).

We do not state the adapted proof here. Except for one statement the adaption is straightforward. Simply substitute $\|(A_l - J_*)\delta x_j\|/\|\delta x_j\|$ by $\|(I - H_l J_*)\delta x_j\|/\|\delta x_j\|$ and apply (4.116) instead of (4.113). The argument in the original proof which requires some additional care relates the rate of convergence to the decrease of $\|(A_l - J_*)\delta x_l\|/\|\delta x_l\|$. It reads as follows: There exist an index l_0 and $\tilde{C}_1$, $\tilde{C}_2 > 0$ such that for all $l \geqslant l_0$ it holds that

$$\begin{aligned} \|x_{l+1} - x_*\| \leqslant \tilde{C}_1 \frac{\|(A_l - J_*)\delta x_i\|}{\|\delta x_l\|} \big(\|x_{l+1} - x_*\| + \|x_l - x_*\|\big) \\ + \tilde{C}_2 \big(\|x_{l+1} - x_*\| + \|x_l - x_*\|\big)^2. \end{aligned} \tag{4.117}$$

By means of the additional assumption in Theorem 4.37 that $\{\|J_*^{-1} H_l^{-1}\|\}$ is uniformly bounded we are able to establish an affine covariant counterpart of (4.117) based on the affine covariant term $\|(I - H_l J_*)\delta x_l\|/\|\delta x_l\|$.

Lemma 4.38 *Suppose Assumption 4.24 holds for F with $\|\cdot\|_2$ substituted by an arbitrary vector norm $\|\cdot\|$. Let $J_* := F'(x_*)$. For an $x_0 \in \mathcal{D}$ and a sequence $\{H_l\} \subset \mathbb{R}^{n\times n}$ of nonsingular matrices consider the iteration* (4.115)*. If the sequence $\{x_l\}$ converges to x_* with $x_l \neq x_*$ $\forall l$, and if a positive constant $\hat{\psi}$ exists such that $\|J_*^{-1}H_l^{-1}\| \leqslant \hat{\psi}$ $\forall l$, then there is a $\kappa > 0$ and an index L such that for all $l \geqslant L$ it holds that*

$$\begin{aligned}\|x_{l+1} - x_*\| \leqslant \frac{\|\big(I - H_l F'(x_*)\big)\delta x_l\|}{\|\delta x_l\|} \cdot \frac{\hat{\psi}}{\kappa}\big(\|x_{l+1} - x_*\| + \|x_l - x_*\|\big) \\ + \frac{\omega}{2\kappa}\big(\|x_{l+1} - x_*\| + \|x_l - x_*\|\big)^2.\end{aligned} \tag{4.118}$$

Proof. Let $e_l := x_l - x_*$. From equation (4.108) we obtain by the Lipschitz continuity of F', cf. (4.104),

$$\|J_*^{-1}F(x_{l+1})\| \leqslant \hat{\psi}\|(I - H_l J_*)\delta x_l\| + \frac{\omega}{2}(\|e_l\| + \|e_{l+1}\|)\|\delta x_l\|.$$

Note that $\delta x_l = e_{l+1} - e_l$ and therefore,

$$\|J_*^{-1}F(x_{l+1})\| \leqslant \hat{\psi} \cdot \frac{\|(I - H_l J_*)\delta x_l\|}{\|\delta x_l\|}(\|e_l\| + \|e_{l+1}\|) + \frac{\omega}{2}(\|e_l\| + \|e_{l+1}\|)^2. \tag{4.119}$$

Since convergence is assumed there exists an index L such that by Lemma 4.32 and the equivalence of norms in finite spaces it holds that

$$\kappa\|e_{l+1}\| \leqslant \|J_*^{-1}F(x_{l+1})\| \quad \forall l \geqslant L$$

with $\kappa > 0$ independent of l. Applying this result to (4.119) and rearranging leads to the claimed statement.

■

The additional assumption that $\{\|J_*^{-1}H_l^{-1}\|\}$ is uniformly bounded is not that much of a restriction if we consider the update (4.82). Because if the assumptions of Theorem 4.27 hold and therefore q-linear convergence is obtained it also holds that $\{\|J_*^{-1}H_l^{-1}\|\}$ is uniformly bounded.

It remains to show that under appropriate conditions the sequence $\{H_l\}$ of matrices generated by the update (4.82) satisfies the affine covariant nonlinear heredity property (4.116). The proof of this statement is long and technical and therefore can be found in Appendix I.

Theorem 4.39 (Heredity) *Suppose that F satisfies Assumption 4.24. Let $J_* := F'(x_*)$. Assume that the sequence of iterates generated by* (4.81) *with the update* (4.82) *is well defined and converges to x_* with $x_l \neq x_*$ $\forall l$. Furthermore assume that all H_l are nonsingular. Let the transposed Dennis-Moré series* (4.93) *be bounded and let the sequence $\{v_l\}$ fulfill the affine covariant residual property 4.31. Additionally, assume that there exists a positive constant ψ such that $\|H_l J_*\|_2 \leqslant \psi$ for all l. Then for some constant $C > 0$ the estimate*

$$\frac{\|(I - H_l J_*)\delta x_j\|_2}{\|\delta x_j\|_2} \leqslant C \sum_{k=j+1}^{l} \|x_k - x_*\|_2 + \tilde{c}_j \tag{4.120}$$

is valid for $0 \leqslant j < l$. Considering the quantities $\tilde{c}_j$ it holds that $\lim_{j\to\infty} \tilde{c}_j = 0$.

If v_j is chosen as $v_j = \overline{\delta x}_{j+1} = -H_j F(x_{j+1})$ then there exists a constant $\tilde{C} \geqslant C$ such that

$$\frac{\|(I - H_l J_*)\delta x_j\|}{\|\delta x_j\|} \leqslant \tilde{C}\sum_{k=j}^{l}\|x_k - x_*\|_2. \tag{4.121}$$

By means of this result the affine covariant nonlinear heredity property (4.116) holds for the descent update. Hence, from Theorem 4.27, Proposition 4.34 and Theorem 4.37 it follows that the sequence of iterates generated by the iteration (4.81) in conjunction with the descent update (4.55) is capable of q-superlinear convergence to a solution x_* of $F(x) = 0$ with an r-order of convergence given via (4.114).

4.4 A Damped quasi-Newton Iteration

To solve $F(x) = 0$ we presented and discussed in Chapter 3 a damped Newton iteration where the step sizes are monitored by the PNLF. In this section we will provide an approximation to the sequence of iterates which emerges from this damped Newton iteration. For this, we will combine the ideas and concepts from the previous sections of this chapter to define a damped quasi-Newton iteration, i.e.,

$$x_{l+1} = x_l + \lambda_l \delta x_l, \quad \delta x_l = -A_l^{-1} F(x_l), \quad \lambda_l \in (0, 1]. \tag{4.122}$$

Let $F(x_l) \neq 0$. To determine the step size λ_l we will employ the APNLF at x_l from Section 4.1, Definition 4.2. The correction δx_l will either be constructed by means of the descent update from Section 4.3 or it will be a descent approximation (4.6) such that the angle checks (4.35) are passed. In the latter case we will employ the purifying updates from Section 4.2, if necessary, to ensure that (4.35) holds. This means that for purifying purposes we will construct from a given nonsingular matrix $A_{l,0}$ intermediate matrices $A_{l,1}, \ldots, A_{l,\bar{k}_l}$ by means of the douphilic, gradientphilic or Newton-philic update such that for

$$\overline{\delta x}_{l,\bar{k}_l} := -A_{l,\bar{k}_l}^{-1} F(x_l) \qquad \text{and} \qquad A_l := (1 - \alpha_{l,\bar{k}_l}) A_{l,\bar{k}_l}, \quad \text{i.e.,} \quad \delta x_l = \frac{1}{1 - \alpha_{l,\bar{k}_l}} \overline{\delta x}_{l,\bar{k}_l}$$

with

$$\alpha_{l,\bar{k}_l} := \frac{\overline{\delta x}_{l,\bar{k}_l}^T (I - A_{l,\bar{k}_l}^{-1} J_l) \overline{\delta x}_{l,\bar{k}_l}}{\overline{\delta x}_{l,\bar{k}_l}^T \overline{\delta x}_{l,\bar{k}_l}}, \quad J_l := F'(x_l), \tag{4.123}$$

and with

$$P_{l,\bar{k}_l} := \frac{\overline{\delta x}_{l,\bar{k}_l} \overline{\delta x}_{l,\bar{k}_l}^T}{\overline{\delta x}_{l,\bar{k}_l}^T \overline{\delta x}_{l,\bar{k}_l}}$$

and for Δx_l being the Newton correction at x_l the angle conditions

$$\angle\big(\delta x_l, -\operatorname{grad} T(x_l | P_{l,\bar{k}_l} A_{l,\bar{k}_l}^{-1})^T\big) \leqslant \phi \qquad \text{and} \qquad \angle_{est}(\delta x_l, \Delta x_l) \leqslant \psi \tag{4.124}$$

for given $0 \leqslant \phi, \psi < \frac{\pi}{2}$ hold. A subsequent step size control will be based on δx_l and the APNLF $T(x | P_{l,\bar{k}_l} A_{l,\bar{k}_l}^{-1})$. For $l = 0$ the initial approximation $A_{l,0}$ to the current Jacobian is provided by the user and for $l > 0$ we will use information from the previous step. If we decide not to purify, i.e. $\bar{k}_l = 0$, but instead to employ a descent update based on $A_{l,0}$ and $\overline{\delta x}_{l,0}$, i.e.,

$$A_l := A_{l,0} \left[I - \frac{\overline{\delta x}_{l,0} \overline{\delta x}_{l,0}^T}{\overline{\delta x}_{l,0}^T \overline{\delta x}_{l,0}} (I - A_{l,0}^{-1} J_l) \right] \tag{4.125}$$

we will use

$$\delta x_l = \frac{1}{1 - \alpha_{l,0}} \overline{\delta x}_{l,0}$$

and the APNLF $T(x|P_{l,0}A_{l,0}^{-1})$ for the step size control. The correction δx_l results from an application of the descent update, however, the actual matrix update will be postponed. A motivation for this strategy is given in Paragraph 4.4.6.4. How to decide whether purifying shall be applied will be discussed in Subsection 4.4.2.

4.4.1 Main features and basic algorithmic outline

The main features of the above introduced quasi-Newton approach are as follows:

- *structural analogy to the PNLF*
  Recall from (3.70) and (3.71) that the relative change of the PNLF at x_l in the direction of the Newton correction Δx_l is described via

  $$\big(1-\lambda+\mu_l(\lambda)\big)^2$$

  with

  $$\mu_l(\lambda) = -\frac{\Delta x_l^T}{\|\Delta x_l\|_2^2} J_l^{-1}\big(F(x_l+\lambda\Delta x_l) - F(x_l) - \lambda J_l \Delta x_l\big)$$

  whereas by Theorem 4.5 the relative change of the APNLF at x_l in the direction of δx_l is given via

  $$\big(1-\lambda+\overline{\mu}_l(\lambda)\big)^2$$

  with

  $$\overline{\mu}_l(\lambda) = -\frac{\overline{\delta x}_{l,\bar{k}_l}^T}{\overline{\delta x}_{l,\bar{k}_l}^T \overline{\delta x}_{l,\bar{k}_l}} A_{l,\bar{k}_l}^{-1}\big(F(x_l+\lambda\delta x_l) - F(x_l) - \lambda J_l \delta x_l\big) \qquad \text{for some } \bar{k}_l \geqslant 0.$$

  This structural analogy is fundamental for a straightforward adaption of the step size controls from Section 3.4. We will discuss this adaption in Subsection 4.4.6.

- *no operations of complexity* $\mathcal{O}(n^3)$
  We will use rank-1 updates to construct our Jacobian approximations $A_{l,k}$. If there is a QR-decomposition of $A_{l,k}$ a rank-1 update can be incorporated within a complexity of $\mathcal{O}(13\cdot n^2)$ floating point operations, [3]. In [17] an algorithm is presented which incorporates a rank-1 update into an LU-decomposition including a specific pivoting strategy. The complexity is bounded by $\mathcal{O}(c\cdot n^2)$ with $c \in [\frac{5}{2}, \frac{9}{2}]$. The value of c depends on the number of times where pivoting is applied. We opt for the latter approach and will discuss it in Subsection 4.4.5.

  Note that for the first matrix $A_{0,0}$ a decomposition has to be computed from scratch. Generally, this takes $\mathcal{O}(n^3)$ operations. However, this has to be done only once per run of the iteration (4.122).

- *affine covariant globalization approach*
  For an affine covariant compatible choice of $A_{0,0}$, e.g. $A_{0,0} = J_0$, the sequence of iterates from (4.122) will feature affine covariance. This holds because the purifying updates and the descent update maintain affine covariance compatibility and the step size strategies are adaptations of the strategies from Section 3.4.

- *w_l-strategy and projected nonlinearity bound predictor*
  In order to check whether the current purifying index k fulfills the angle conditions (4.124), and hence may become $\bar{k}_l$, for both tests in (4.124) the term

  $$\overline{\delta x}_{l,k}^T A_{l,k}^{-1} J_l \tag{4.126}$$

  is required. For the first test this is clear since the negative gradient of $T(x|P_{l,k}A_{l,k}^{-1})$ at x_l must be calculated which is just the above stated vector. For the second test recall from Algorithm 4.2 that $\angle_{est}(\delta x_l, \Delta x_l)$ depends on the quantity r_{rel}^{est} which by Algorithm 4.1 also requires the evaluation of the gradient of the APNLF at x_l.

  If the descent update is considered there is also the need to compute (4.126) for $k = 0$, cf. (4.125). Note that in any case we can use (4.126) to compute $\alpha_{l,k}$ as well, cf. (4.123).

  The term (4.126) is an adjoint tangent evaluation which is available via the reverse mode of AD. Therefore, we have to calculate the product

  $$g_{l,k}^T := \overline{\delta x}_{l,k}^T A_{l,k}^{-1}. \tag{4.127}$$

  This is just the analogon to the quantity w_l^T from (3.95),

  $$w_l^T = \Delta x_l^T J_l^{-1}.$$

  Hence, we apply the strategy to evaluate a term of the form

  $$\overline{\delta x}_{l,k}^T A_{l,k}^{-1} F(x_l + \lambda_{l,j}\delta x_l), \quad \lambda_{l,j} \in (0,1],$$

  by calculating

  $$\hat{g}_{l,k}^T \cdot F(x_l + \lambda_{l,j}\delta x_l). \tag{4.128}$$

  Analogously to the Newton case, an $\mathcal{O}(n^2)$ operation, the computation of $A_{l,k}^{-1}F(x_l+\lambda_{l,j}\delta x_l)$, is saved. The product (4.128) will be used to evaluate the APNLF at a trial iterate or for the construction of a corrector step size, respectively. For details refer to the discussion in Paragraph 4.4.6.2 and 4.4.6.1. Since both quantities (4.126) and (4.127) are already computed an adaption of the projected nonlinearity bound predictor is directly available. How to determine the predictor will be discussed in Paragraph 4.4.6.3.

- *superlinear convergence possible*
  Provided the sequence of iterates generated by the damped iteration (4.122) converges to a solution x_* we can show under some additional conditions that the convergence is superlinear. However, these conditions are rather harsh since among other assumptions it is required that the sequence of all Jacobian approximations $\{A_{l,k}\}$ converges to a fixed nonsingular matrix A_* and only a finite number of purifying updates are performed. We will state the convergence result in Paragraph 4.4.6.6.

A basic outline of our approach is described by the following algorithm:

Algorithm 4.4 (Basic outline of the quasi-Newton approach at step l)

given: $A_{l,0} \in \mathbb{R}^{n\times n}$ nonsingular, $\overline{\delta x}_{l,0} := -A_{l,0}^{-1}F_l$ with $F_l := F(x_l) \neq 0$,
    ϕ, ψ with $0 \leqslant \phi, \psi < \frac{\pi}{2}$
Determine whether a descent update (4.125) from $A_{l,0}$ and $\overline{\delta x}_{l,0}$ is to be performed or if a purifying process has to be started
if use_descent_update **then**
    set $\bar{k}_l = 0$
else
    invoke a purifying process: employ the duophilic, gradientphilic and Newton-philic purifying
    updates from Section 4.2 iteratively to find an index $\bar{k}_l$ such that the angle conditions
    (4.124) are fulfilled for given ϕ and ψ
end if
use $T(x|P_{l,\bar{k}_l}A_{l,\bar{k}_l}^{-1})$ and $\delta x_l = (1-\alpha_{l,\bar{k}_l})^{-1}\overline{\delta x}_{l,\bar{k}_l}$ in an adaption of the step size controls from Section 3.4 to determine λ_l and x_{l+1}
if convergence criterion not met yet **then**
    if $\bar{k}_l = 0$ **then**
        apply the matrix descent update (4.125) to determine A_l
        set $A_{l+1,0} = A_l$
    else
        set $A_l = (1-\alpha_{l,\bar{k}_l})A_{l,\bar{k}_l}$
        set $A_{l+1,0} = A_{l,\bar{k}_l}$
    end if
    invoke this algorithm for $l+1$
else
    stop the iteration
    return x_{l+1} as approximation for a solution x_*
end if

In the following we will discuss several aspects of the above algorithm.

- We will explain our strategy under which conditions we will skip a purifying process and instead directly employ the descent update.
- Regarding the purifying process we will discuss when we will apply which specific purifying update.
- In analogy to the Newton case we will introduce proper scaling to achieve scaling invariance of the iteration (4.122).
- For the above defined matrices $A_{l,k}$ we will maintain an LU-decomposition. We will convey the basic principle of the algorithm proposed in [17] which incorporates a rank-1 update into a given LU-decomposition.

- For an adaption of the step size controls from Section 3.4 we will define new local (step size dependent) nonlinearity bounds. By means of these quantities we will see that the adaption is straightforward. We will also discuss an adaption of the predictors from Section 3.4. Additionally, we will present a strategy when to reconsider purifying. Furthermore, a termination criterion will be presented and also a result about superlinear convergence.

Remark 4.40 In Appendix II there can be found a detailed algorithmic representation of our strategy how to determine a Jacobian approximation $A_{l,k}$ and an associated correction $\delta x_l = (1-\alpha_{l,k})^{-1}\overline{\delta x}_{l,k}$ which are employed in the adaption of the step size strategies from Section 3.4. This representation also involves the computation of predictor step sizes. We advice the reader first to consider the upcoming related explanations in this section, namely

- Subsection 4.4.2
  A strategy to decide whether a descent update is preferred to a purifying process
- Subsection 4.4.3
  Algorithmic aspects of the purifying process
- Paragraph 4.4.6.3
  Adaption of predictors
- Paragraph 4.4.6.4
  Post-purifying

before studying the algorithmic representation in Appendix II. □

4.4.2 A strategy to decide whether a descent update is preferred to a purifying process

If the descent update is recursively applied in a full step iteration we know from the results of Section 4.3 that under appropriate conditions local superlinear convergence of the sequence of iterates to a solution x_* of $F(x) = 0$ is achieved. Furthermore, by means of Corollary 4.30 and 4.35 the angle conditions (4.124) are asymptotically fulfilled for arbitrary nonnegative ϕ and ψ.

For our superlinear convergence result in Theorem 4.33 to hold it is crucial that the transposed Dennis-Moré property (4.94) is valid. This property in turn is based on the estimate (4.79) which holds if bounded deterioration of the Jacobian approximations by means of Theorem 4.26 is ensured, i.e., if the conditions

$$\|I - A_l^{-1}J_*\|_2 \leqslant \delta < \frac{1}{3}$$
$$\|x_{l+1} - x_*\|_2 \leqslant \zeta$$

and

$$1 - \big(\delta + (1+\delta)\omega_{(4.66)}\zeta\big) > 0$$

hold for each index l, cf. (4.69)-(4.71). Recall that the constant $\omega_{(4.66)}$ is defined via the Lipschitz condition

$$\|F'(x_*)^{-1}\big(F'(x) - F'(x_*)\big)\|_2 \leqslant \omega\|x - x_*\|_2 \quad \forall x \in \mathcal{D}.$$

We do not think that especially for ω a proper and cheap estimate at an iterate l is available. Hence, to decide whether a descent update shall be applied we need different criteria. Our strategy is based on the angle conditions (4.124) and an estimate of the step size λ_l. We will explain it in the following and provide an algorithmic representation of it—see Algorithm 4.5 below.

To determine δx_l via the descent update (4.125) we have to compute $\alpha_{l,0}$ which is available if

$$g_{l,0}^T := -\operatorname{grad} T(x_l|P_{l,0}A_{l,0}^{-1})\big) = \hat{g}_{l,0}^T \cdot J_l \tag{4.129}$$

with $\hat{g}_{l,0}$ defined according to (4.127) is known. We first check if

$$|1-\alpha_{l,0}| < \varepsilon \tag{4.130}$$

for some predefined $\varepsilon \ll 1$. If this is the case then A_l defined via the descent update (4.125) may be close to singular since by the Matrix Determinant Lemma

$$\det(A_l) = (1-\alpha_{l,0}) \cdot \det(A_{l,0}). \tag{4.131}$$

We directly opt for purifying in this case. If (4.130) does not hold we compute by means of $\overline{\delta x}_{l,0}$, $\alpha_{l,0}$ and $g_{l,0}$ the left angle of (4.124), i.e.,

$$\angle\big(\delta x_l, -\operatorname{grad} T(x_l|P_{l,0}A_{l,0}^{-1})^T\big) = \angle\big((1-\alpha_{l,0})^{-1}\overline{\delta x}_{l,0}, g_{l,0}\big) \tag{4.132}$$

and demand that the angle condition

$$\angle\big((1-\alpha_{l,0})^{-1}\overline{\delta x}_{l,0}, g_{l,0}\big) \leqslant \phi$$

is fulfilled in accordance to (4.124).

The vector $g_{l,0}$ is also required to compute r_{est}^{rel} which in turn is needed to determine $\angle_{est}(\delta x_l, \Delta x_l)$, cf. Algorithm 4.1 and 4.2. More precisely, $g_{l,0}$ is employed to compute the estimate opn_{est} from (4.32). With the definition of $g_{l,0}$ and introducing indices and setting $\overline{H} = A_{l,0}^{-1}$ this estimate reads as follows

$$\operatorname{opn}_{est}^{l,0} := \max\left[\frac{\left\|\big(I - \frac{1}{1-\alpha_{l,0}}A_{l,0}^{-1}J_l\big)\overline{\delta x}_{l,0}\right\|_2}{\|\overline{\delta x}_{l,0}\|_2}, \frac{\left\|\overline{\delta x}_{l,0}^T - \frac{1}{1-\alpha_{l,0}}g_{l,0}^T\right\|_2}{\|\overline{\delta x}_{l,0}\|_2}\right].$$

Recall from Algorithm 4.1 that we require $\operatorname{opn}_{est}^{l,0} < 1$ in order to calculate r_{est}^{rel}. To determine $\operatorname{opn}_{est}^{l,0}$ the direct tangent evaluation $J_l\overline{\delta x}_{l,0}$ must be available. On the other hand, this evaluation is not necessary if A_l is chosen as a descent update of $A_{l,0}$. So instead of computing $J_l\overline{\delta x}_{l,0}$ and hereby $\operatorname{opn}_{est}^{l,0}$ and $\angle_{est}(\delta x_l, \Delta x_l)$, respectively, we decide to determine only

$$a_{l,0} := \frac{\left\|\overline{\delta x}_{l,0}^T - \frac{1}{1-\alpha_{l,0}}g_{l,0}^T\right\|_2}{\|\overline{\delta x}_{l,0}\|_2} \tag{4.133}$$

and check the necessary condition

$$a_{l,0} < 1.$$

Our local convergence analysis in Section 4.3 requires a full step iteration, i.e. $\lambda_l = 1\ \forall\, l$, in order to guarantee local superlinear convergence for a sequence of iterates $\{x_l\}$ where the corrections

δx_l are defined via a recursive application of the descent update. However, the step size λ_l is determined *after* we made the choice whether the descent update is applied or a purifying process is initiated. Fortunately, if $l \geqslant 1$ and by means of $\hat{g}_{l,0}$ and $g_{l,0}$ we are in the state to compute a straightforward adaption of the projected nonlinearity bound predictor $\lambda_{l,0_3}$ from (3.97) which we defined in Section 3.4 in the context of Newton's method. We postpone the definition of the adapted predictor $\lambda_{l,0}$ to (4.161) in Subsection 4.4.6 where we will also derive adaptations of the simple predictor (3.87) and the projected Deuflhard's predictor (3.91). For the time being assume that the adapted projected nonlinearity bound predictor is available. If this predictor is one and additionally λ_{l-1} is equal to one this gives good reason to hope that also λ_l will be one in case that δx_l is defined via the descent update. Hence, we demand that $\lambda_{l,0} = \lambda_{l-1} = 1$ to opt for the descent update. If $l = 0$ then no predictor information is available. Instead we check whether the user provided step size $\lambda_{0,0}$ is equal to one.

The above stated conditions to opt for the descent update are summarized in the following algorithm.

Algorithm 4.5 (Strategy to decide for a descent update or a purifying process at x_l)

---

given: $A_{l,0} \in \mathbb{R}^{n\times n}$ nonsingular, $\overline{\delta x}_{l,0} := -A_{l,0}^{-1}F_l$ with $F_l := F(x_l) \neq 0$,
  $J_l := F'(x_l)$ (not explicitly given but for adjoint tangent evaluations)
  $\varepsilon \ll 1$, ϕ from (4.124) with $\frac{\pi}{2} > \phi$,
  $\lambda_{0,0}$ if $l = 0$ or λ_{l-1} if $l > 0$
determine $\hat{g}_{l,0}^T = \overline{\delta x}_{l,0} A_{l,0}^{-1}$
determine $g_{l,0}^T = \hat{g}_{l,0}^T J_l$ ▷ i.e. $-\operatorname{grad} T(x_l|P_{l,0}A_{l,0}^{-1})$
determine $\alpha_{l,0} = 1 - g_{l,0}^T \overline{\delta x}_{l,0} / \|\overline{\delta x}_{l,0}\|_2^2$ ▷ i.e. $\frac{\overline{\delta x}_{l,0}^T (I - A_{l,0}^{-1}J_l)\overline{\delta x}_{l,0}}{\overline{\delta x}_{l,0}^T \overline{\delta x}_{l,0}}$
if $|1 - \alpha_{l,0}| < \varepsilon$ **then** ▷ A_l as a descent update of $A_{l,0}$ may be close to singular
  use_descent_update = false
else
  determine $a_{l,0}$ from (4.133)
  if $\big(\angle\big((1-\alpha_{l,0})^{-1}\overline{\delta x}_{l,0}, g_{l,0}\big) \leqslant \phi\big)$ && $(a_{l,0} < 1)$ **then** ▷ $\angle\big(\delta x_l, -\operatorname{grad} T(x_l|P_{l,0}A_{l,0}^{-1})^T\big) \leqslant \phi$
    if $l > 0$ **then**
      if $\lambda_{l-1} = 1$ **then**
        determine the adapted projected nonlinearity bound predictor $\lambda_{l,0}$ defined
        according to (4.160) and (4.161)
        if $\lambda_{l,0} = 1$ **then**
          use_descent_update = true
        else
          use_descent_update = false
        end if
      else
        use_descent_update = false
      end if

```
        else if λ_{0,0} = 1 then
            use_descent_update = true
        else
            use_descent_update = false
        end if
    else
        use_descent_update = false
    end if
end if
```

4.4.3 Algorithmic aspects of the purifying process

As long as not both angles in (4.124) are zero there is at least one of the three purifying updates from Subsection 4.2.1 well defined. So if the Newton correction at x_l is well defined, i.e. J_l is nonsingular, and nonzero then by Corollary 4.18 after a finite number of applications of the purifying updates the angle conditions (4.124) are fulfilled. If already one of the two angle conditions is violated a purifying process shall be initiated or continued, respectively. Due to the basic outline of our quasi-Newton approach, cf. Algorithm 4.4, we first check if we will go for a descent update instead of a purifying process. Recall from (4.129) that for this check we compute

$$g_{l,0}^T = -\operatorname{grad} T(x_l|P_{l,0}A_{l,0}^{-1}).$$

This means that $\angle\big(\delta x_l, -\operatorname{grad} T(x_l|P_{l,0}A_{l,0}^{-1})^T\big)$ is readily at hand, cf. (4.132). This is not true for the estimated angle between δx_l and the Newton correction Δx_l since the direct tangent evaluation $J_l\overline{\delta x}_{l,0}$ is not available yet. This gives rise to the following prioritization strategy:

I) Meet the gradient related angle condition $\angle\big(\delta x_l, -\operatorname{grad} T(x_l|P_{l,k}A_{l,k}^{-1})^T\big) \leqslant \phi$.

II) If the angle condition in I) is fulfilled then check for $\angle_{est}(\delta x_l, \Delta x_l) \leqslant \psi$.

This strategy is further backed up by the fact that for computing $\angle_{est}(\delta x_l, \Delta x_l)$ it is anyway necessary that the vector $g_{l,k}$ defined according to (4.129) is available since the estimated angle depends on the quantity $a_{l,k}$ defined according to (4.133), cf. the discussion in the previous subsection.

Though we decide to check the two angle conditions from (4.124) in the above stated particular order, it is still the goal to meet both. The duophilic update aims at providing a correction δx_l which satisfies both angle conditions. So this will be our main purifying update. Recall from the definition of the duophilic update, (4.49), that the next Jacobian approximation is given via

$$A_{l,k+1} = A_{l,k} - \frac{(A_{l,k} - J_l)\overline{\delta x}_{l,k}\overline{\delta x}_{l,k}^T(I - A_{l,k}^{-1}J_l)}{\overline{\delta x}_{l,k}^T(I - A_{l,k}^{-1}J_l)\overline{\delta x}_{l,k}}. \tag{4.134}$$

For $A_{l,k+1}$ to be well defined it is assumed that $\overline{\delta x}_{l,k}^T(I - A_{l,k}^{-1})\overline{\delta x}_{l,k} \neq 0$. In numerical computations this is practically always the case. However, this term can be very close to zero which may cause

numerical instability of the duophilic update. We will take care of that problem later on but first we consider a case of purifying where the duophilic update is definitely applicable:

The angle conditions (4.124) are given in terms of the correction δx_l which is a scaled version of $\overline{\delta x}_{l,k}$, namely $\delta x_l = (1-\alpha_{l,k})^{-1}\overline{\delta x}_{l,k}$. δx_l is not well defined if $\alpha_{l,k} = 1$ which means that an associated descent update is singular, cf. (4.131). As described in the previous subsection a purifying process is already initiated if $|1-\alpha_{l,0}| < \varepsilon$, $\varepsilon \ll 1$. We will consider this check also for all purifying indices $k > 0$ to ensure that δx_l is not related to a descent update probably close to singular. If

$$|1-\alpha_{l,k}| < \varepsilon \tag{4.135}$$

is true we will employ the duophilic update for purifying purposes. Since

$$\alpha_{l,k} = \frac{\overline{\delta x}_{l,k}^T (I - A_{l,k}^{-1} J_l)\overline{\delta x}_{l,k}}{\overline{\delta x}_{l,k}^T \overline{\delta x}_{l,k}}$$

the duophilic update, (4.134), can be written as

$$A_{l,k+1} = A_{l,k}\left[I - \frac{1}{\alpha_{l,k}} \cdot (I - A_{l,k}^{-1} J_l)\frac{\overline{\delta x}_{l,k}}{\|\overline{\delta x}_{l,k}\|_2}\frac{\overline{\delta x}_{l,k}^T}{\|\overline{\delta x}_{l,k}\|_2}(I - A_{l,k}^{-1} J_l)\right]. \tag{4.136}$$

So if (4.135) is true then

$$\alpha_{l,k} \approx 1$$

which means that no numerical instabilities are introduced by dividing by $\alpha_{l,k}$ in the above update formula and hence the duophilic update is safely applicable.

Notice from (4.134) that the rank-1 update of the duophilic update only depends on the direction of $\overline{\delta x}_{l,k}$ but not on the magnitude of $\overline{\delta x}_{l,k}$. So we do not think it is reasonable to employ a check like $\overline{\delta x}_{l,k}^T (I - A_{l,k}^{-1})\overline{\delta x}_{l,k} < \varepsilon_2$ for some $0 < \varepsilon_2 \ll 1$ to decide whether a duophilic update shall be applied. From the product representation (4.136) of the duophilic update it is seen that it is indeed a better choice to check for

$$\alpha_{l,k} < \varepsilon_2 \tag{4.137}$$

since this check is independent of the magnitude of $\overline{\delta x}_{l,k}$. Also, it can be directly evaluated since $\alpha_{l,k}$ was already computed for the check (4.135). However, this check may tend to disadvantage the duophilic update since it does not take the magnitude of the two vectors

$$(I - A_{l,k}^{-1} J_l)\frac{\overline{\delta x}_{l,k}}{\|\overline{\delta x}_{l,k}\|_2} \qquad \text{and} \qquad \frac{\overline{\delta x}_{l,k}^T}{\|\overline{\delta x}_{l,k}\|_2}(I - A_{l,k}^{-1} J_l)$$

into account which form the rank-1 update in (4.136). In the following we will discuss more sophisticated strategies than the check (4.137) to decide whether a duophilic update or a gradientphilic or Newton-philic update, respectively, shall be used.

4.4.3.1 Gradient related angle – duophilic or gradientphilic update

If the gradient related angle condition

$$\angle\big(\delta x_l, -\operatorname{grad} T(x_l|P_{l,k_l}A_{l,k_l}^{-1})^T\big) \leqslant \phi \tag{4.138}$$

is violated it is either the duophilic or the gradientphilic update we will employ to improve the approximation quality of the next Jacobian approximation. To decide which one to choose one may refer to the check (4.137). As an alternative approach we present the following strategy:

If (4.138) is not true it is safe to assume that $\|\overline{\delta x}_{l,k}^T(I - A_{l,k}^{-1}J_l)\|_2/\|\overline{\delta x}_{l,k}\|_2$ is not close to zero. Because otherwise it would follow from arguments we used in the proof of Corollary 4.30 that the transposed negative gradient of the APNLF at x_l and $\overline{\delta x}_{l,k}$ would be close to each other in terms of magnitude and direction. Suppose that $|\alpha_{l,k}|$ is small. This may be the case because

$$\frac{\|(I - A_{l,k}^{-1}J_l)\overline{\delta x}_{l,k}\|_2}{\|\overline{\delta x}_{l,k}\|_2}$$

is small since

$$|\alpha_{l,k}| \leqslant \frac{\|(I - A_{l,k}^{-1}J_l)\overline{\delta x}_{l,k}\|_2}{\|\overline{\delta x}_{l,k}\|_2}.$$

Hence, in such a case dividing by $\alpha_{l,k}$ of small magnitude in the update (4.136) does not necessarily imply that the rank-1 term 'blows up' since it may be outweighed by the magnitude of the left vector of the rank-1 update in (4.136). Consider the transformed product representation of the duophilic update

$$A_{l,k+1} = A_{l,k}\left[I - \cos\Big[\angle\big(\overline{\delta x}_{l,k}, (I - A_{l,k}^{-1}J_l)\overline{\delta x}_{l,k}\big)\Big]^{-1} \cdot \frac{(I - A_{l,k}^{-1}J_l)\overline{\delta x}_{l,k}}{\|(I - A_{l,k}^{-1}J_l)\overline{\delta x}_{l,k}\|_2}\frac{\overline{\delta x}_{l,k}^T}{\|\overline{\delta x}_{l,k}\|_2}(I - A_{l,k}^{-1}J_l)\right]$$

which is based on the fact that

$$\alpha_{l,k} = \cos\Big[\angle\big(\overline{\delta x}_{l,k}, (I - A_{l,k}^{-1}J_l)\overline{\delta x}_{l,k}\big)\Big] \cdot \frac{\|(I - A_{l,k}^{-1}J_l)\overline{\delta x}_{l,k}\|_2}{\|\overline{\delta x}_{l,k}\|_2}.$$

Therefore, the Euclidean norm of the above rank-1 update to the identity matrix is given via

$$\cos\Big[\angle\big(\overline{\delta x}_{l,k}, (I - A_{l,k}^{-1}J_l)\overline{\delta x}_{l,k}\big)\Big]^{-1} \cdot \frac{\|\overline{\delta x}_{l,k}^T(I - A_{l,k}^{-1}J_l)\|_2}{\|\overline{\delta x}_{l,k}\|_2}.$$

The strategy is to consider a gradientphilic update only if

$$\alpha_{l,k} < \varepsilon_2 \qquad \text{and} \qquad \Big|\cos\Big[\angle\big(\overline{\delta x}_{l,k}, (I - A_{l,k}^{-1}J_l)\overline{\delta x}_{l,k}\big)\Big]\Big| < \varepsilon_3 \tag{4.139}$$

hold for some additional $0 < \varepsilon_3 \ll 1$. Clearly, this approach is more in favor of the duophilic update than just checking the left condition $\alpha_{l,k} < \varepsilon_2$. However, additional computational effort is introduced: Due to our prioritization strategy the direct tangent evaluation $J_l\overline{\delta x}_{l,k}$ is not done yet when the gradient related angle (4.138) is checked. Though this term may be employed for a possible subsequent duophilic update the product $A_{l,k}^{-1}J_l\overline{\delta x}_{l,k}$ is solely needed for the above check.

To avoid possibly unnecessary computational effort we refrain from utilizing the more sophisticated strategy and opt for the simple check (4.135) in our algorithmic realization. However, as we will see in the next paragraph, an adaption of the above described strategy is indeed justifiable regarding computational effort, if purifying is considered because the angle condition related to the Newton correction, i.e., $\angle_{est}(\delta x_l, \Delta x_l) \leqslant \psi$ is violated.

4.4.3.2 Newton correction related angle – duophilic or Newton-philic update

Due to our prioritization strategy the angle check

$$\angle_{est}(\delta x_l, \Delta x_l) \leqslant \psi \tag{4.140}$$

is only evaluated if already the gradient related check (4.138) is passed and hence sufficient gradient related approximation quality of the current Jacobian approximation is given. So if the above angle check fails we will either employ a duophilic or Newton-philic update in order to improve the approximation quality also w.r.t. the Newton correction. By having already evaluated both checks the duophilic update is extremely cheap to compute since both required tangent evaluations $\overline{\delta x}_{l,k}^T A_{l,k}^{-1} J_l$ and $J_l \overline{\delta x}_{l,k}$ are available. On the contrary, for the Newton-philic update we require the unknown adjoint tangent evaluation $\left((I - A_{l,k}^{-1} J_l)\overline{\delta x}_{l,k}\right)^T A_{l,k}^{-1} J_l$, cf. the definition of the Newton-philic update in (4.54) for $A_w = A_{l,k}$. This is additional computational effort we try to avoid if possible. Therefore, we will not employ the simple check (4.137) to decide whether a duophilic update may be applicable but adapt the more sophisticated strategy from the previous paragraph:

For the check (4.140) the term $\|(I - A_{l,k}^{-1} J_l)\overline{\delta x}_{l,k}\|_2 / \|\overline{\delta x}_{l,k}\|_2$ must be available since it is required for the computation of r_{est}^{rel} which in turn is necessary for the angle estimate $\angle_{est}(\delta x_l, \Delta x_l)$, cf. Algorithm 4.1 and 4.2. If $\|(I - A_{l,k}^{-1} J_l)\overline{\delta x}_{l,k}\|_2 / \|\overline{\delta x}_{l,k}\|_2$ is not sufficiently small the check fails. Assume that this is the case. Also assume that $|\alpha_{l,k}|$ is small. This may be due to a small magnitude of $\|\overline{\delta x}_{l,k}^T (I - A_{l,k}^{-1} J_l)\|_2 / \|\overline{\delta x}_{l,k}\|_2$ since

$$|\alpha_{l,k}| \leqslant \frac{\|\overline{\delta x}_{l,k}^T (I - A_{l,k}^{-1} J_l)\|_2}{\|\overline{\delta x}_{l,k}\|_2}.$$

In analogy to the argument in the previous paragraph this does not necessarily mean that the rank-1 update to the identity matrix in (4.136) 'blows up': In the case considered here the magnitude of the right vector of the two vectors which form the rank-1 update in (4.136) may outweigh the magnitude of $\alpha_{l,k}$. Since

$$\alpha_{l,k} = \cos\left[\angle\left(\overline{\delta x}_{l,k}, \left(\overline{\delta x}_{l,k}^T (I - A_{l,k}^{-1} J_l)\right)^T\right)\right] \cdot \frac{\|\overline{\delta x}_{l,k}^T (I - A_{l,k}^{-1} J_l)\|_2}{\|\overline{\delta x}_{l,k}\|_2}$$

the product representation of the duophilic update can be written as

$$\begin{aligned} A_{l,k+1} = A_{l,k} \Bigg[I - \cos\left[\angle\left(\overline{\delta x}_{l,k}, \left(\overline{\delta x}_{l,k}^T (I - A_{l,k}^{-1} J_l)\right)^T\right)\right]^{-1} \\ \cdot (I - A_{l,k}^{-1} J_l) \frac{\overline{\delta x}_{l,k}}{\|\overline{\delta x}_{l,k}\|_2} \frac{\overline{\delta x}_{l,k}^T (I - A_{l,k}^{-1} J_l)}{\|\overline{\delta x}_{l,k}^T (I - A_{l,k}^{-1} J_l)\|_2} \Bigg] \end{aligned} \tag{4.141}$$

and the Euclidean norm of the above rank-1 update to the identity matrix is equal to

$$\cos\left[\angle\left(\overline{\delta x}_{l,k}, \left(\overline{\delta x}_{l,k}^T (I - A_{l,k}^{-1} J_l)\right)^T\right)\right]^{-1} \cdot \frac{\|(I - A_{l,k}^{-1} J_l)\overline{\delta x}_{l,k}\|_2}{\|\overline{\delta x}_{l,k}\|_2}.$$

Hence, we check for

$$\alpha_{l,k} < \varepsilon_2 \qquad \text{and} \qquad \left|\cos\left[\angle\left(\overline{\delta x}_{l,k}, \left(\overline{\delta x}_{l,k}^T (I - A_{l,k}^{-1} J_l)\right)^T\right)\right]\right| < \varepsilon_3 \tag{4.142}$$

and compute a Newton-philic update only if both conditions are true. Analogously to the derived strategy from the previous paragraph a duophilic update is preferred to a Newton-philic update by the introduction of the above second check. Compared to the extended check (4.139) the advantage here is that the second check in (4.142) is readily available since the adjoint tangent evaluation $\overline{\delta x}_{l,k}^T A_{l,k}^{-1} J_l$ was already computed to evaluate the gradient related angle condition (4.138).

4.4.3.3 A strategy to detect and handle singular or ill-conditioned Jacobian approximations

Even though J_l may be supposed to be nonsingular we may run into a singular or ill-conditioned Jacobian approximation $A_{l,k}$ during a purifying process. If we characterize $A_{l,k}$ by the term *ill-conditioned* we mean that $\text{cond}_2(A_{l,k})$ is of magnitude close to the reciprocal value of the available machine precision.

If $A_{l,k}$ is singular or ill-conditioned and if we are able to detect this by our strategy, which we will discuss below, we will employ the Newton-philic update (4.54) to proceed with the purifying process. For the weight matrix A_w we choose the latest nonsingular, not ill-conditioned Jacobian approximation. Hence, we assume that at least $A_{0,0}$ fulfills these properties. For the Newton-philic update we have to take special care of the determination of $\overline{\delta x}_{l,k}$ since either $A_{l,k}^{-1}$ is not well defined or a numerical computation of $\overline{\delta x}_{l,k} = -A_{l,k}^{-1} F_l$ is distorted by an excessive amplification of rounding errors.

If $A_{l,k}$ is singular then according to (4.52) $\overline{\delta x}_{l,k}$ will be chosen such that $\overline{\delta x}_{l,k} \in \ker(A_{l,k}) \setminus \{0\}$. Singularity of $A_{l,k}$ is directly detectable via its LU-factorization. $A_{l,k}$ is singular if and only if at least one diagonal element u_{ii} of U is zero. Let $\underline{i}$ be the smallest index such that the respective diagonal element of U is zero. If we set $(\overline{\delta x}_{l,k})_{(i)} = 0,\ i = \underline{i}+1, \ldots, n$, and $(\overline{\delta x}_{l,k})_{(\underline{i})} = 1$ then backward substitution for the remaining indices $1, \ldots, \underline{i}-1$ in the system $U\overline{\delta x}_{l,k} = 0$ yields $\overline{\delta x}_{l,k} \in \ker(A_{l,k}) \setminus \{0\}$.

However, in numerical computation $A_{l,k}$ is virtually never singular. Though, it may be ill-conditioned. If we consider a nonsingular $A_{l,k}$ to be ill-conditioned by the strategy we will describe below we define $\overline{\delta x}_{l,k}$ to be an approximation to a singular vector w.r.t. the smallest singular value of $A_{l,k}$. To obtain such an approximation we adapt the idea of the LINPACK condition estimator, see e.g. [3] for an explanation of that estimator, and execute a few steps of inverse iteration for the matrix $A_{l,k}^T A_{l,k}$. To solve the associated linear systems we employ the LU-factorization of $A_{l,k}$. The inverse iteration requires an initial guess for the singular vector. We choose $A_{l,k}^{-1} F_l$ for that purpose. Notice that rank revealing strategies for LU-factorizations are based on complete pivoting, i.e. on interchanges of rows and columns—see e.g. [20]. The LU-factorizations of the Jacobian approximations $A_{l,k}$ are based on partial pivoting since we employ the algorithm from [17] to incorporate rank-1 updates into a given LU-factorization. Hence, rank revealing properties are in general not given. However, by the constructed LU-factorizations at least a rough estimate of the condition number of the underlying Jacobian approximations is available. We will make use of it in the following strategy:

I) Let the LU-factorization of $A_{l,k}$ be given via $PA_{l,k} = LU$ for some appropriate permutation matrix P. If one diagonal element u_{ii} of U is zero then $A_{l,k}$ is singular and we determine $\overline{\delta x}_{l,k} \in \ker(A_{l,k}) \setminus \{0\}$ in the above described way.

II) If no u_{ii} is zero then $\mathrm{cond}_2(A_{l,k})$ is finite and it holds that

$$\mathrm{cond}_2(A_{l,k}) \leqslant \mathrm{cond}_2(L) \cdot \mathrm{cond}_2(U)$$

since $\mathrm{cond}_2(P) = \mathrm{cond}_2(P^T) = 1$. To obtain a cheaply computable estimate for the above upper bound we use the eigenvalues of L and U. For both matrices the eigenvalues are given as the diagonal elements l_{ii} and u_{ii}, respectively. Let M be L or U, respectively, and let the diagonal elements of M be denoted by m_{ii}. With

$$\mathrm{cond}_2^{est}(M) := \max_{i,j} \frac{|m_{ii}|}{|m_{jj}|}$$

we obtain

$$\mathrm{cond}_2^{est}(L) \leqslant \mathrm{cond}_2(L) \qquad \text{and} \qquad \mathrm{cond}_2^{est}(U) \leqslant \mathrm{cond}_2(U).$$

Note that $\mathrm{cond}_2^{est}(L) = 1$ since all l_{ii} are equal to one. Therefore, our strategy is to check for

$$\mathrm{cond}_2^{est}(U) > K \tag{4.143}$$

where K is some large prescribed constant. If the above inequality holds we consider U and also $A_{l,k}$ to be ill-conditioned. In this case $\overline{\delta x}_{l,k}$ will be computed as an approximation to a singular vector w.r.t. the smallest singular value of $A_{l,k}$ in the above described way.

Before we construct the Newton-philic update with $\overline{\delta x}_{l,k}$ defined by the above strategy we check if

$$\frac{\|A_w^{-1}(A_{l,k} - J_l)\overline{\delta x}_{l,k}\|_2}{\|\overline{\delta x}_{l,k}\|_2} < \varepsilon_{sing} \tag{4.144}$$

for some positive scalar $\varepsilon_{sing} \ll 1$ holds. If this is the case we deem J_l (nearly) singular and stop the whole iteration.

Remark 4.41 Notice that we are aware of the fact that $\mathrm{cond}_2^{est}(L)$ is a poor estimate since it anyway holds that $\mathrm{cond}_2(L) \geqslant 1$. However, we do not think that in general useful information about the condition number of L is cheaply available. □

4.4.4 Scaling invariance

Like in the Newton case, cf. Subsection 3.4.3, we will see that scaling invariance is obtained by switching to relative quantities $x^{rel} := D_{\hat{x}}^{-1} x$ where $D_{\hat{x}} := \mathrm{diag}(\hat{x})$ with $\hat{x} \in \mathcal{D}$ and $\hat{x}_{(i)} \neq 0$, $i = 1, \ldots, n$. Let a change of variables in the domain space of F again be represented by

$$y := S^{-1}x \qquad \text{with} \quad S := \mathrm{diag}(s_{11}, \ldots, s_{nn}),\ s_{ii} \neq 0,\ i = 1, \ldots, n$$

which leads to the transformed system

$$G(y) = 0, \qquad G(y) := F(Sy).$$

Accordingly, relative quantities are given via

$$y^{rel} := D_{\hat{y}}^{-1} y \quad \text{where} \quad D_{\hat{y}} := S^{-1} D_{\hat{x}}.$$

As we stated already in Subsection 3.4.3 it holds that

$$y^{rel} = x^{rel}.$$

Also, we know from that subsection that for

$$F^{rel}(x^{rel}) := F(D_{\hat{x}}x^{rel}) \qquad \text{and} \qquad G^{rel}(y^{rel}) := G(D_{\hat{y}}y^{rel}) = G(y)$$

we have

$$F^{rel}(x^{rel}) = G^{rel}(y^{rel}) = F(x) \tag{4.145}$$

and

$$(F^{rel})'(x^{rel}) = F'(x)D_{\hat{x}} = G'(y)D_{\hat{y}} = (G^{rel})'(y^{rel}). \tag{4.146}$$

Let A be an approximation to the Jacobian $F'(x)$ and accordingly $B := AS$ an approximation to the Jacobian $G'(y)$. If we set

$$A^{rel} := AD_{\hat{x}} \qquad \text{and} \qquad B^{rel} := BD_{\hat{y}}$$

as approximations to $(F^{rel})'(x^{rel})$ and $(G^{rel})'(y^{rel})$, respectively, then

$$\delta x^{rel} := -(A^{rel})^{-1}F^{rel}(x^{rel}) = -D_{\hat{x}}^{-1}AF(x) = -(B^{rel})^{-1}G^{rel}(y^{rel}) =: \delta y^{rel} \tag{4.147}$$

since (4.145) holds and also

$$B^{rel} = ASS^{-1}D_{\hat{x}} = AD_{\hat{x}} = A^{rel}.$$

This means that scaling invariance is ensured. Next, we will show that the rank-1 updates we employ in our quasi-Newton approach do not destroy scaling invariance. We restrict ourselves to the descent update (4.61). The line of argument for the three purifying updates is done in an analogous way.

We will see that under the condition $A^{rel} = B^{rel}$ it also holds that $A_+^{rel} = B_+^{rel}$ where A_+^{rel} and B_+^{rel} emerge from a descent update of A^{rel} and B^{rel}, respectively.

First consider the system

$$F^{rel}(x^{rel}) = 0.$$

We define

$$\overline{\delta x}_+^{rel} := -(A^{rel})^{-1}F(x_+^{rel}), \quad x_+^{rel} := D_{\hat{x}}^{-1}x_+,\ x_+ \in \mathcal{D} \text{ with } F(x_+) \neq 0.$$

Then a descent update of A^{rel} reads as follows

$$A_+^{rel} = A^{rel}\left[I - \frac{\overline{\delta x}_+^{rel}(\overline{\delta x}_+^{rel})^T}{(\overline{\delta x}_+^{rel})^T\overline{\delta x}_+^{rel}}\Big(I - (A^{rel})^{-1}(F^{rel})'(x_+^{rel})\Big)\right]. \tag{4.148}$$

In case of the system

$$G^{rel}(y^{rel}) = 0$$

we define in accordance to $\overline{\delta x}_+^{rel}$ the correction $\overline{\delta y}_+^{rel}$ via

$$\overline{\delta y}_+^{rel} := -(B^{rel})^{-1}F(y_+^{rel}), \quad y_+^{rel} := D_{\hat{y}}^{-1}y_+,\ y_+ = S^{-1}x_+.$$

Then a descent update of B^{rel} is obtained via

$$B_+^{rel} = B^{rel}\left[I - \frac{\overline{\delta y}_+^{rel}(\overline{\delta y}_+^{rel})^T}{(\overline{\delta y}_+^{rel})^T\overline{\delta y}_+^{rel}}\Big(I - (B^{rel})^{-1}(G^{rel})'(y_+^{rel})\Big)\right].$$

Assume that $A^{rel} = B^{rel}$. Then

$$\overline{\delta x}_+^{rel} = \overline{\delta y}_+^{rel}$$

by the definitions of the two above corrections and since (4.145) holds. Also

$$(F^{rel})'(x_+^{rel}) = (G^{rel})'(y_+^{rel})$$

by (4.146) with x_+^{rel} and y_+^{rel} substituted for x^{rel} and x^{rel}, respectively. Hence, $A_+^{rel} = B_+^{rel}$ which by an argument like in (4.147) implies that scaling invariance is preserved.

Remark 4.42 Let $\overline{\delta x}_+ := -A^{-1}F(x_+)$. Since $(F^{rel})'(x_+^{rel}) = F'(x_+)D_{\hat{x}}$ and $F^{rel}(x_+^{rel}) = F(x_+)$ and by the definitions of A^{rel} and $\overline{\delta x}_+^{rel}$ the matrix A_+^{rel} can be written as

$$\begin{aligned} A_+^{rel} &= A\left[I - \frac{D_{\hat{x}}^{-1}\overline{\delta x}_+\overline{\delta x}_+^T}{\overline{\delta x}_+^T D_{\hat{x}}^{-2}\overline{\delta x}_+}D_{\hat{x}}^{-2}\big(I - A^{-1}F'(x_+)\big)\right]D_{\hat{x}} \\ &=: A_+ D_{\hat{x}}. \end{aligned}$$

Since A_+ is not invariant under $D_{\hat{x}}^{-1}$, a scaling in the domain space of F, a sequence of iterates constructed via

$$x_{l+1}^{rel} = x_l^{rel} + \delta x_l^{rel}, \quad \delta x_l^{rel} := -(A_l^{rel})^{-1}F^{rel}(x^{rel}),$$

where A_l^{rel} is recursively defined by means of (4.148) does not feature affine contravariance. However, it does feature affine covariance. This is in contrast to Newton's method where a sequence of associated iterates satisfies both affine invariance concepts. For an affine contravariant quasi-Newton approach which in turn does not feature affine covariance refer to the work of Schlenkrich, [28]. □

4.4.5 Maintaining an LU-decomposition of the Jacobian approximations

To incorporate rank-1 updates into a given LU-decomposition we employ an algorithm proposed by Kiełbasiński and Schwetlick in [17]. In this subsection we will explain the basic idea of this algorithm. Let $A \in \mathbb{R}^{n\times n}$ and u, $v \in \mathbb{R}^n$ be given. Also, let there be a permutation matrix P such that

$$PA = LU$$

is an LU-decomposition of PA. We seek for an LU-decomposition

$$P_+A_+ = L_+U_+$$

of P_+A_+ where A_+ is defined via

$$A_+ := A + uv^T$$

and P_+ is some appropriate permutation matrix. Therefor, we write A_+ as

$$A_+ = P^T L(U + rv^T) \tag{4.149}$$

where r is the solution of $Lr = Pu$. The idea is to transform r to a multiple of the first unit vector e_1 in $\mathbb{R}^n$ via a Gaussian elimination process. This way the rank-1 update contains relevant data only in its first row and therefore can be safely added to the transformed U-factor, which due to the Gaussian elimination process becomes an upper Hessenberg matrix. In a second step, an additional Gaussian elimination process is initiated to transform the upper Hessenberg matrix back to an upper triangular matrix. During these two Gaussian elimination processes pivoting is taken into account in a particular way. We will explain the pivoting strategy below. If pivoting occurs an intermediate step is to be performed to reobtain the structure of the L-factor.

We perform the first step of the first Gaussian elimination process to convey how the algorithm works and in which way pivoting is considered. This explanation is along the lines of the comments on the algorithm in [28].

Let the Gauß-transformation $L_i(a) \in \mathbb{R}^{n\times n}$ for $a \in \mathbb{R}$ be defined according to

$$L_i(a) := \begin{pmatrix} I_{i-2} & & & \\ & 1 & & \\ & a & 1 & \\ & & & I_{n-i} \end{pmatrix}.$$

By I_k we denote the identity matrix in $\mathbb{R}^{k\times k}$. It holds that $L_i(a)L_i(-a) = I$ and multiplying a vector $x = \big(x_{(1)}, \ldots, x_{(n)}\big)^T \in \mathbb{R}^n$ by $L_i(a)$ from the left yields

$$L_i(a)x = \big(x_{(1)}, \ldots, x_{(i-1)}, ax_{(i-1)} + x_{(i)}, x_{(i+1)}, \ldots, x_{(n)}\big)^T.$$

Especially, if $x_{(i)} \neq 0$ we obtain for the choice $a = -\frac{x_{(i)}}{x_{(i-1)}}$,

$$L_i(a)x = \big(x_{(1)}, \ldots, x_{(i-1)}, 0, x_{(i+1)}, \ldots, x_{(n)}\big)^T.$$

With a focus on the last two rows we write (4.149) as

$$A_+ =: P^T \begin{pmatrix} L' & & \\ l_{n-1}^T & 1 & \\ l_n^T & l_{nn-1} & 1 \end{pmatrix} \cdot \left[\begin{pmatrix} U' & u_{n-1} & u_n \\ & * & * \\ & & * \end{pmatrix} + \begin{pmatrix} r' \\ r_{(n-1)} \\ r_{(n)} \end{pmatrix} \cdot v^T \right].$$

In [17] pivoting is considered if

$$|r_{(n-1)}| < |r_{(n)} + l_{nn-1} \cdot r_{(n-1)}|. \tag{4.150}$$

For the time being assume that $|r_{(n-1)}|$ is sufficiently large and hence no pivoting occurs. With $a = \frac{-r_{(n)}}{r_{(n-1)}}$ we have

$$\begin{aligned} A_+ &= P^T L L_n(-a)\big(L_n(a)U + L_n(a)rv^T\big) \\ &=: P^T \tilde{L}\big(\tilde{U} + \tilde{r}v^T\big) \end{aligned} \tag{4.151}$$

where the matrix $\tilde{L}$ is lower unit-triangular, the last component of $\tilde{r}$ is zero and for $\tilde{U}$ it holds that

$$\tilde{U} = \begin{pmatrix} U' & u_{n-1} & u_n \\ & * & * \\ & * & * \end{pmatrix}.$$

Though fill-in occurs, it is restricted to the index $(n, n-1)$.

If for $r_{(n-1)}$, $r_{(n)}$ and l_{nn-1} the condition (4.150) is true then pivoting is considered. Therefor, a permutation matrix P_n is applied to swap the positions of $r_{(n-1)}$ and $r_{(n)}$. Note that $P_n^T = P_n$. The permutation P_n is also applied from the left to L. This way the new permutation information is incorporated in the already existent permutation information reflected by P. We obtain

$$A_+ = P^T P_n \cdot P_n L P_n (P_n U + P_n r v^T).$$

The right hand side of the above equation reads as follows

$$P^T P_n \cdot \begin{pmatrix} L' & & \\ l_n^T & 1 & l_{nn-1} \\ l_{n-1}^T & 0 & 1 \end{pmatrix} \cdot \left[\begin{pmatrix} U' & u_{n-1} & u_n \\ & 0 & * \\ & * & * \end{pmatrix} + \begin{pmatrix} r' \\ r_{(n)} \\ r_{(n-1)} \end{pmatrix} \cdot v^T \right].$$

Due to the permutation the lower unit-triangular structure of L is lost. To regain it $L_n(a)^T$ with $a = -l_{nn-1}$ is applied in the following way:

$$\begin{aligned} A_+ &= P^T P_n \cdot \left(P_n L P_n L_n(a)^T \right) \cdot \left(L_n(-a)^T P_n U + L_n(-a)^T P_n r v^T \right) \\ &=: P^T P_n \cdot \bar{L}(\bar{U} + \bar{r} v^T). \end{aligned}$$

Indeed, $\bar{L}$ is a lower unit-triangular matrix,

$$\bar{L} = \begin{pmatrix} L' & & \\ l_n^T & 1 & \\ l_{n-1}^T & 0 & 1 \end{pmatrix}.$$

The matrix $\bar{U}$ is given as

$$\begin{pmatrix} U' & u_{n-1} & u_n \\ & * & * \\ & * & * \end{pmatrix}.$$

Again, fill-in is restricted to the index $(n, n-1)$. The vector r transforms to

$$\bar{r} = \left(r'^T, r_{(n)} + l_{nn-1} \cdot r_{(n-1)}, r_{(n-1)} \right)^T.$$

To eliminate the last component of $\bar{r}$ the matrix $L(a)$ with

$$a = -\frac{r_{(n-1)}}{r_{(n)} + l_{nn-1} \cdot r_{(n-1)}}$$

is employed in analogy to (4.151). This yields

$$A_+ = \hat{P}^T \hat{L}(\hat{U} + \hat{r} v^T)$$

where $\hat{L}$ is lower unit-triangular, $\hat{U}$ is of the same structure as $\bar{U}$, i.e. no additional fill-in occurs, and the last component of $\hat{r}$ is zero.

Repeating this procedure all components of r excluding the first one are eliminated. For each step the pivoting condition (4.150) is adapted to the corresponding indices. Finally, this process

yields a rank-1 update which can be added to the transformed U, an upper Hessenberg matrix, without destroying this structure. We obtain the decomposition

$$A_+ = \widetilde{P}^T \widetilde{L} \widetilde{U}$$

where $\widetilde{P}$ is a permutation matrix, $\widetilde{L}$ a lower unit-triangular matrix and $\widetilde{U}$ the aforementioned upper Hessenberg matrix including the transformed rank-1 update. It remains to transform $\widetilde{U}$ back to upper triangular form, i.e., to eliminate the subdiagonal elements $\widetilde{u}_{21}$ to $\widetilde{u}_{nn-1}$. This is done by the same techniques which are used to transform r into a multiple of the first unit vector e_1. However, this time the process starts in the upper left corner. This finally yields

$$A_+ = P_+^T L_+ U_+ \quad \Leftrightarrow \quad P_+ A_+ = L_+ U_+,$$

i.e., an LU-decomposition of $P_+ A_+$ where $A_+ = A + uv^T$.

The described algorithm can be performed in place, e.g., to store the fill-in components of the transformed U one may use the storage space of the second to last component of r. In terms of required floating point operations a complexity of $\mathcal{O}(c \cdot n^2)$ with $c \in [\frac{5}{2}, \frac{9}{2}]$ arises. The actual value of c depends on how often pivoting is considered. If pivoting in every step is considered then $c = \frac{9}{2}$. On the contrary, no pivoting at all results in $c = \frac{5}{2}$. In order to decrease the number of permutations and therefore increase computational efficiency one may introduce a damping factor $\tau \in (0, 1]$ in (4.150),

$$|r_{(n-1)}| < \tau \cdot |r_{(n)} + l_{nn-1} \cdot r_{(n-1)}|.$$

In an attempt to comply with both concepts – stability and efficiency – in [30] the choice $\tau = 0.1$ is suggested. We also opt for this damping factor in our algorithmic realization.

4.4.6 Basics of an adaption of the step size controls from Section 3.4

For an adaption of the step size controls from Section 3.4 we will exploit the structural analogy of the PNLF and APNLF. Let $F_l := F(x_l) \neq 0$ and $J_l := F'(x_l)$ be nonsingular for some iterate $x_l \in \mathcal{D}$. Recall from (3.70) and (3.71) that for the PNLF and the Newton correction Δx_l at x_l we have

$$\frac{T(x_l + \lambda \Delta x_l | P_{N_l} J_l^{-1})}{T(x_l | P_{N_l} J_l^{-1})} = \big(1 - \lambda + \mu_l(\lambda)\big)^2$$

with

$$\mu_l(\lambda) = -\frac{\Delta x_l^T}{\|\Delta x_l\|_2^2} J_l^{-1} \big(F(x_l + \lambda \Delta x_l) - F_l - \lambda J_l \Delta x_l\big).$$

In the following let $A_{l,k}$ be a nonsingular approximation to the Jacobian J_l and assume that $\alpha_{l,k}$ defined according to (4.123) is not equal to one. By Theorem 4.5 for the PNLF it holds that

$$\frac{T(x_l + \lambda \delta x_l | P_{l,k} A_{l,k}^{-1})}{T(x_l | P_{l,k} A_{l,k}^{-1})} = \big(1 - \lambda + \overline{\mu}_l(\lambda)\big)^2$$

with

$$\overline{\mu}_{l,k}(\lambda) = -\frac{\overline{\delta x}_{l,k}^T}{\overline{\delta x}_{l,k}^T \overline{\delta x}_{l,k}} A_{l,k}^{-1} \big(F(x_l + \lambda \delta x_l) - F_l - \lambda J_l \delta x_l\big),$$

$$\delta x_l = \frac{1}{1 - \alpha_{l,k}} \overline{\delta x}_{l,k} \qquad \text{and} \qquad \overline{\delta x}_{l,k} = -A_{l,k}^{-1} F_l.$$

We will first present straightforward adaptations of nonlinearity bounds which we employed in the Newton case. Based on these adaptations a computable corrector step size is presented. Additionally, we will present counterparts of the three predictors introduced in Paragraph 3.4.1.3. For the actual computation of predictor and corrector step sizes and the evaluation of the APNLF we will exploit that the quantities

$$\hat{g}_{l,k}^T := \overline{\delta x}_{l,k}^T A_{l,k}^{-1} \qquad \text{and} \qquad g_{l,k}^T := \hat{g}_{l,k}^T \cdot J_l = -\operatorname{grad} T(x_l | P_{l,k} A_{l,k}^{-1})\big)$$

are available which we already used to compute $\alpha_{l,k}$ and the angles in (4.124). Furthermore, we will discuss a post-purifying process, i.e., purifying after step sizes have been computed and also state a termination criterion and a result about superlinear convergence.

Remark 4.43 In the following we will discuss the determination of step sizes for a general purifying index k instead of the final index $\bar{k}_l$ at step l. This is in contrast to our basic outline of the quasi-Newton approach, Algorithm 4.4. However, there the post-purifying concept is not taken into account. Due to this concept it may be the case that step sizes are determined for intermediate purifying indices k as well—see Paragraph 4.4.6.4 for further information. □

4.4.6.1 Nonlinearity bounds and corrector step sizes

First we will provide the basics of an adaption of the theoretical background of the simple and restricted monotonicity approaches from Subsection 3.4.1 and 3.4.2, respectively. Then, we will define an adaption of the corrector step size (3.80). Additionally, providing definitions of adapted predictor step sizes in Paragraph 4.4.6.3 the crucial elements to carry over the step size controls from Section 3.4 to the quasi-Newton context are present.

As an adaption of the nonlinearity bound (3.73),

$$2\|P_{N_l} J_l^{-1}\big(F(y) - F_l - J_l(y - x_l)\big)\|_2 \leqslant \omega_l \|y - x_l\|_2^2$$

for all $y \in \mathcal{D}$ with $y - x_l = \lambda \Delta x_l$, $\lambda \in [0, 1]$, we introduce

$$2\|P_{l,k} A_{l,k}^{-1}\big(F(y) - F_l - J_l(y - x_l)\big)\|_2 \leqslant \omega_{l,k} \|y - x_l\|_2^2 \tag{4.152}$$

for all $y \in \mathcal{D}$ with $y - x_l = \lambda \delta x_l$, $\delta x_l = (1 - \alpha_{l,k})^{-1} \overline{\delta x}_{l,k}$, $\lambda \in [0, 1]$.

Considering the step size dependent nonlinearity bound (3.102),

$$\omega_l(\lambda) = \sup_{s \in (0,\lambda]} 2 \frac{\|P_{N_l} J_l^{-1}\big(F(x_l + s\Delta x_l) - F_l - s J_l \Delta x_l\big)\|_2}{s^2 \|\Delta x_l\|_2^2}$$

our adaption is as follows

$$\omega_{l,k}(\lambda) := \sup_{s \in (0,\lambda]} 2 \frac{\|P_{l,k} A_{l,k}^{-1}\big(F(x_l + s\delta x_l) - F_l - s J_l \delta x_l\big)\|_2}{s^2 \|\delta x_l\|_2^2} \tag{4.153}$$

where $\delta x_l = (1 - \alpha_{l,k})^{-1} \overline{\delta x}_{l,k}$.

Let

$$\Lambda_l := \{\lambda \in (0, 1] \mid x_l + \lambda \delta x_l \in \mathcal{D}\}.$$

Then by means of (4.152) and (4.153) and in analogy to (3.19) we obtain for $\lambda \in \Lambda_l$

$$\begin{aligned}
\overline{\mu}_{l,k}(\lambda) &= -\frac{\overline{\delta x}_{l,k}^T}{\|\overline{\delta x}_{l,k}\|_2^2} A_{l,k}^{-1}\big(F(x_l+\lambda\delta x_l) - F_l - \lambda J_l \delta x_l\big) \\
&\leqslant |\overline{\mu}_{l,k}(\lambda)| = \frac{\|P_{l,k}A_{l,k}^{-1}\big(F(x_l+\lambda\delta x_l) - F_l - \lambda J_l\delta x_l\big)\|_2}{\lambda^2\|\overline{\delta x}_{l,k}\|_2\|\delta x_l\|_2}\|\delta x_l\|_2\lambda^2 \qquad (4.154)\\
&\leqslant \tfrac{1}{2}|1-\alpha_{l,k}|^{-1}\omega_{l,k}(\lambda)\|\delta x_l\|_2\lambda^2 \\
&\leqslant \tfrac{1}{2}|1-\alpha_{l,k}|^{-1}\omega_{l,k}\|\delta x_l\|_2\lambda^2.
\end{aligned}$$

According to the results of Theorem 4.5 this means that for all $\lambda \in \Lambda_l$ it holds that,

$$\begin{aligned}
\frac{T(x_l+\lambda\delta x_l|P_{l,k}A_{l,k}^{-1})}{T(x_l|P_{l,k}A_{l,k}^{-1})} &\leqslant \big(1-\lambda+\tfrac{1}{2}|1-\alpha_{l,k}|^{-1}\omega_{l,k}(\lambda)\|\delta x_l\|_2\lambda^2\big)^2 \qquad (4.155)\\
&\leqslant \big(1-\lambda+\tfrac{1}{2}|1-\alpha_{l,k}|^{-1}\omega_{l,k}\|\delta x_l\|_2\lambda^2\big)^2. \qquad (4.156)
\end{aligned}$$

The unique minimizer in $[0,1]$ of the polynomial in (4.156) is given as

$$\overline{\lambda}_{l,k} := \min\left(1, \frac{|1-\alpha_{l,k}|}{\omega_{l,k}\|\delta x_l\|_2}\right)$$

and it holds that

$$\frac{T(x_l+\lambda\delta x_l|P_{l,k}A_{l,k}^{-1})}{T(x_l|P_{l,k}A_{l,k}^{-1})} < 1 \qquad \forall\lambda \in (0, 2\overline{\lambda}_{l,k}) \cap \Lambda_l.$$

The model in (4.155) is in analogy to the model (3.104) we derived in the Newton case. So if $\omega_{l,k}(\lambda)$ is uniformly bounded for all $\lambda \in \Lambda_l$ and if the path-connected component of the level set $\{z \in \mathcal{D} \mid T(z|P_{l,k}A_{l,k}^{-1}) \leqslant T(x_l|P_{l,k}A_{l,k}^{-1})\}$ which contains x_l is a subset of $\mathcal{D}$ then following the lines of proof of Theorem 3.42 one can verify that there exist modeling step sizes $\lambda_{l_m} = \lambda_{l_m}(\eta) \in \Lambda_l$, $\eta \in (0,2)$, which fulfill

$$\lambda_{l_m} = \min\left(1, \frac{|1-\alpha_{l,k}|\cdot\eta}{\omega_{l,k}(\lambda_{l_m})\|\delta x_l\|_2}\right)$$

and it holds that

$$\frac{T(x_l+\lambda\delta x_l|P_{l,k}A_{l,k}^{-1})}{T(x_l|P_{l,k}A_{l,k}^{-1})} < 1 \qquad \forall\lambda \in (0, \lambda_{l_m}].$$

Next, we will provide a computable estimate for $\overline{\lambda}_{l,k}$ and λ_{l_m}, respectively. Such an estimate is available if an estimate for $\omega_{l,k}$ and $\omega_{l,k}(\lambda)$, respectively can be derived. Analogously to the definition of $[\omega]_{l,k}$ in the Newton case, cf. (3.78), we define

$$[\omega]_{l,k}(\lambda) := 2\frac{\|P_{l,k}A_{l,k}^{-1}\big(F(x_l+\lambda\delta x_l) - F_l - \lambda J_l\delta x_l\big)\|_2}{\lambda^2\|\delta x_l\|_2^2}. \qquad (4.157)$$

By definition we have

$$[\omega]_{l,k}(\lambda) \leqslant \omega_{l,k}(\lambda) \leqslant \omega_{l,k}$$

and if F is three-times continuously differentiable one can verify by following the lines of proof of Lemma 3.41 that $[\omega]_{l,k}(\lambda)$ and $\omega_{l,k}(\lambda)$ are in close relationship via

$$[\omega]_{l,k}(\lambda) = \omega_{l,k}(\lambda) + \mathcal{O}(\lambda).$$

The estimate (4.157) requires a step size $\lambda > 0$. So in analogy to the Newton case we can use this estimate for the definition of a corrector step size. Let $\lambda_{l,j} \in \Lambda_l$ be given. Then we define the corrector

$$\lambda^c_{l,j} := \frac{|1-\alpha_{l,k}|}{[\omega]_{l,k}(\lambda_{l,j})\|\delta x_l\|_2} = \frac{|1-\alpha_{l,k}|^2}{[\omega]_{l,k}(\lambda_{l,j})\|\overline{\delta x}_{l,k}\|_2}. \tag{4.158}$$

Form this definition we see that if $A_{l,k}$ is affine covariant compatible the corrector is affine covariant.

By means of (4.154) and the definition of $[\omega]_{l,k}(\lambda)$ we see that

$$|\overline{\mu}_{l,k}(\lambda_{l,j})| = \tfrac{1}{2}|1-\alpha_{l,k}|^{-1}[\omega]_{l,k}(\lambda_{l,j})\|\delta x_l\|_2\lambda^2_{l,j} = \tfrac{1}{2}\big(\lambda^c_{l,j}\big)^{-1}\lambda^2_{l,j}.$$

Hence,

$$\lambda^c_{l,j} = \tfrac{1}{2}\big(|\overline{\mu}_{l,k}(\lambda_{l,j})|\big)^{-1}\lambda^2_{l,j}.$$

From (4.9) we derive that $\overline{\mu}_{l,k}(\lambda_{l,j})$ can be written as

$$\frac{\overline{\delta x}^T_{l,k}\overline{\delta x}_{l_j+}}{\|\overline{\delta x}_{l,k}\|^2_2} - (1-\lambda_{l,j})$$

where

$$\overline{\delta x}_{l_j+} := -A^{-1}_{l,k}F(x_l + \lambda_{l,j}\delta x_l).$$

Since the quantity $\hat{g}_{l,k}$ is available and in analogy to the w_l-strategy (3.95)-(3.96) in the Newton case we can determine the product $\overline{\delta x}^T_{l,k}\overline{\delta x}_{l_j+}$ by means of

$$-\hat{g}^T_{l,k}F(x_l + \lambda_{l,j}\delta x_l).$$

Thus, the corrector is cheaply computable via

$$\lambda^c_{l,j} = \frac{1}{2}\cdot\left[\left|\frac{-\hat{g}^T_{l,k}F(x_l+\lambda_{l,j}\delta x_l)}{\|\overline{\delta x}_{l,k}\|^2_2} - (1-\lambda_{l,j})\right|\right]^{-1}\cdot\lambda^2_{l,j}.$$

4.4.6.2 Efficient evaluation of $T(x|P_{l,k}A^{-1}_{l,k})$ and the simple monotonicity check

Since

$$T(x|P_{l,k}A^{-1}_{l,k}) = \tfrac{1}{2}\|P_{l,k}A^{-1}_{l,k}F(x)\|^2_2$$

it holds that

$$T(x|P_{l,k}A^{-1}_{l,k}) = \tfrac{1}{2}\big[|\overline{\delta x}^T_{l,k}A^{-1}_{l,k}F(x)|/\|\overline{\delta x}_{l,k}\|_2\big]^2.$$

Hence,

$$T(x_l+\lambda_{l,j}\delta x_l|P_{l,k}A^{-1}_{l,k}) < T(x_l|P_{l,k}A^{-1}_{l,k})$$

$$\Leftrightarrow$$

$$|\overline{\delta x}^T_{l,k}A^{-1}_{l,k}F(x_l+\lambda_{l,j}\delta x_l)| < \|\overline{\delta x}_{l,k}\|^2_2.$$

The term $|\overline{\delta x}^T_{l,k}A^{-1}_{l,k}F(x_l+\lambda_{l,j}\delta x_l)|$ can be efficiently computed by means of $\hat{g}_{l,k}$:

$$|\overline{\delta x}^T_{l,k}A^{-1}_{l,k}F(x_l+\lambda_{l,j}\delta x_l)| = |\hat{g}^T_{l,k}F(x_l+\lambda_{l,j}\delta x_l)|. \tag{4.159}$$

Notice that the product $\hat{g}^T_{l,k}F(x_l+\lambda_{l,j}\delta x_l)$ can be reused for the evaluation of a subsequent corrector step size.

4.4.6.3 Adaption of predictors

The predictors $\lambda_{l,0}$ we will introduce are of the same structure as the corrector step size (4.158), i.e.,

$$\lambda_{l,0} = \min\left(1, \frac{|1-\alpha_{l,k}|^2}{[\omega]\cdot\|\overline{\delta x}_{l,k}\|_2}\right) \tag{4.160}$$

but substitute $[\omega]_{l,k}(\lambda_{l,j})$ with an estimate $[\omega]$ for the local nonlinearity of F which is based on information from the previous step. We will provide an adaption for each of the Newton predictors from Section 3.4. Since the Newton predictors are of the form

$$\lambda_{l,0}^N = \min\left(1, \frac{1}{[\omega]^N\|\Delta x_l\|_2}\right)$$

it is the quantity $[\omega]^N$ we are going to substitute to obtain the respective predictor in the quasi-Newton context. For comparison we also state $[\omega]^N$. The index $\bar{k}_{l-1}$ will denote the final purifying index from step $l-1$ which may be equal to zero if no purifying was considered in step $l-1$.

- *simple predictor*
  In the Newton case we have
  $$[\omega]^N = [\omega]_{l-1}(\lambda_{l-1}),$$
  cf. (3.87). Hence, the adaption is straightforward by choosing
  $$[\omega] = [\omega]_{l-1,\bar{k}_{l-1}}(\lambda_{l-1}).$$
  This quantity is already known from the previous step, so no additional computational effort is introduced.

- *Deuflhard-like predictor*
  From (3.90) we see that for the projected Deuflhard's predictor the quantity $[\omega]^N$ is given as
  $$[\omega]^N = \frac{\|P_{N_l}J_l^{-1}(J_l - J_{l-1})\overline{\Delta x}_l\|_2}{\lambda_{l-1}\|\Delta x_{l-1}\|_2\|\overline{\Delta x}_l\|_2}$$
  where $\overline{\Delta x}_l := -J_{l-1}^{-1}F_l$. Our adaption is the following quantity
  $$[\omega] = \frac{\|P_{l,k}A_{l,k}^{-1}(J_l - J_{l-1})\delta x_{l-1}\|_2}{\lambda_{l-1}\|\delta x_{l-1}\|_2^2}.$$
  Since $\delta x_{l-1} = (1-\alpha_{l-1,\bar{k}_{l-1}})^{-1}\overline{\delta x}_{l-1,\bar{k}_{l-1}}$ and by means of $\hat{g}_{l,k}$ and $g_{l,k}$ we can write
  $$[\omega] = \frac{|g_{l,k}^T\delta x_{l-1} - (1-\alpha_{l-1,\bar{k}_{l-1}})^{-1}\hat{g}_{l,k}^T J_{l-1}\overline{\delta x}_{l-1,\bar{k}_{l-1}}|}{\lambda_{l-1}\|\delta x_{l-1}\|_2^2\|\overline{\delta x}_{l,k}\|_2}.$$
  An efficient computation of this predictor is possible if from the previous step the direct tangent evaluation $J_{l-1}\overline{\delta x}_{l-1,\bar{k}_{l-1}}$ is known. However, this is not necessarily the case. If the descent update was employed in the previous step it may be the case that the angle estimate $\angle_{est}(\delta x_{l-1}, \Delta x_{l-1})$ was not computed and hence $J_{l-1}\overline{\delta x}_{l-1,\bar{k}_{l-1}}$ is unknown. Assuming the direct tangent evaluation is at hand we still have to compute $g_{l,k}^T\delta x_{l-1}$ and $\hat{g}_{l,k}^T\cdot J_{l-1}\overline{\delta x}_{l-1,\bar{k}_{l-1}}$. Each product introduces additional computational effort of complexity $\mathcal{O}(2n)$.

- *nonlinearity bound predictor*
  Considering the projected nonlinearity bound predictor in the context of Newton's method, according to (3.94) the quantity $[\omega]^N$ reads as follows:

$$[\omega]^N = 2\frac{\|P_{N_l} J_l^{-1}\big(F_{l-1} - F_l + \lambda_{l-1} J_l \Delta x_{l-1}\big)\|_2}{\lambda_{l-1}^2 \|\Delta x_{l-1}\|_2^2}.$$

  Straightforward adaption leads to

$$[\omega] = 2\frac{\|P_{l,k} A_{l,k}^{-1}(F_{l-1} - F_l + \lambda_{l-1} J_l \delta x_{l-1})\|_2}{\lambda_{l-1}^2 \|\delta x_{l-1}\|_2^2}. \tag{4.161}$$

  Using $\hat{g}_{l,k}$ and $g_{l,k}$ once more the right hand side can be written as

$$[\omega] = 2\frac{|\hat{g}_{l,k}^T F_{l-1} + \|\overline{\delta x}_{l,k}\|_2^2 + \lambda_{l-1} g_{l,k}^T \delta x_{l-1})|}{\lambda_{l-1}^2 \|\delta x_{l-1}\|_2^2 \cdot \|\overline{\delta x}_{l,k}\|_2}.$$

  The products $\hat{g}_{l,k}^T F_{l-1}$ and $g_{l,k}^T \delta x_{l-1}$ are unknown but can be cheaply evaluated each with a complexity of $\mathcal{O}(2n)$.

Note that all predictors are affine covariant if this is true for the involved quantities from the previous step and if $A_{l,k}$ is affine covariant compatible.

4.4.6.4 Post-purifying

If the angle conditions (4.124) are only fulfilled for $\phi, \psi > 0$ there is still a difference between δx_l and the Newton correction Δx_l as well as between δx_l and the gradient of the related APNLF. Hence, we cannot guarantee the same behavior of the APNLF in the direction of δx_l and the PNLF in the direction of Δx_l. If the step size control yields too small step sizes we reconsider purifying. More precisely, if $\lambda_{l,j}$ does not lead to descent and if $\lambda_{l,j} \leqslant \lambda_{thresh}$ for some prescribed $\lambda_{thresh} > 0$ then we check the angle conditions (4.124) for a more restrictive choice of ϕ and ψ. This way we may enforce the resumption or initialization of a purifying process.

Since it is not a priori known whether purifying after the initiation of the step size control occurs we refrain from computing the actual descent update (4.125) from $A_{l,0}$ right away, if we opted for it. For the step size control the matrix update is not required, only its effect on F_l is of importance and this is already reflected by the factor $(1-\alpha_{l,0})^{-1}$ in conjunction with the direction $-A_{l,0}^{-1} F_l$.

4.4.6.5 Termination criterion and providing $A_{l+1,0}$ for the next step

In analogy to the Newton case we terminate the iteration if

$$\|\delta x_l\|_2 \leqslant \mathrm{XTOL} \tag{4.162}$$

holds for some prescribed tolerance XTOL and use

$$x_{l+1,*} := x_l + \delta x_l$$

as final approximation to a solution x_*. Like in the Newton case we provide a second termination criterion which involves an estimate for the error at the next iterate x_{l+1}: Assume that simple or

restricted monotonicity, respectively, holds for a predictor of magnitude one. This already implies that $x_{l+1} = x_l + \delta x_l$ and no more purifying will occur, i.e., the current purifying index k becomes $\bar{k}_l$. In case of simple monotonicity we also assume that $\min(1, \lambda^c_{l,0}) = 1$ with the corrector $\lambda^c_{l,0}$ from (4.158) holds. Then we check for

$$\|A^{-1}_{l,\bar{k}_l} F(x_{l+1})\|_2 \leqslant \text{XTOL} \tag{4.163}$$

If this is true we return

$$x_{*,l+1} := x_l - A^{-1}_{l,\bar{k}_l} F(x_{l+1})$$

as the final estimate. If the check fails the computation of $A^{-1}_{l,\bar{k}_l} F(x_{l+1})$ is not in vain since we can use it for the next step: According to Algorithm 4.4, the basic outline of our quasi-Newton approach, at the next step $l+1$ there must be a nonsingular matrix $A_{l+1,0}$ and the direction $\overline{x}_{l+1,0} = -A^{-1}_{l+1,0} F_{l+1}$ available. To provide these quantities our strategy is as follows: If the final purifying index for step l is bigger than zero, i.e., $\bar{k}_l > 0$ than we define $A_{l+1,0} := A_{l,\bar{k}_l}$ and $\overline{x}_{l+1,0} := -A^{-1}_{l,\bar{k}_l} F_{l+1}$. If $\bar{k}_l = 0$ then we apply the descent update (4.125) to obtain A_l and define $A_{l+1,0} := A_l$, $\overline{x}_{l+1,0} := -A_l^{-1} F_{l+1}$. Doing so the chance is given that there is an index $\underline{l}$ such that all A_l with $l \geqslant \underline{l}$ are defined via a descent update of its predecessor. If additionally the iteration converges to a solution of x_* and the step sizes become one then according to our local convergence analysis of the descent update there is the chance for superlinear convergence. Under which conditions we can guarantee superlinear convergence will be shown in the next paragraph.

4.4.6.6 Superlinear convergence

For our result about superlinear convergence it is crucial to assume that the sequence of all Jacobian approximations $\{A_{l,k}\}$ converges to a nonsingular matrix A_* and only a finite number of purifying updates is applied. However, it is not required that $A_* = F'(x_*)$ where x_* is a solution of $F(x) = 0$.

If Algorithm 4.5 decides for a direct application of the descent update, i.e., no purifying process is invoked, and if no post-purifying is considered the final purifying index $\bar{k}_l$ at step l is $\bar{k}_l = 0$ and it holds that

$$|1 - \alpha_{l,\bar{k}_l}| \geqslant \varepsilon \tag{4.164}$$

From the discussion in Subsection 4.4.3 it follows that a termination of a purifying process implies the above inequality for $\bar{k}_l > 0$. We will make use of the relation (4.164) in the proof of the following theorem.

The idea of the proof for superlinear convergence is to meet the conditions of Theorem 4.33 from our local convergence analysis in Subsection 4.3.2. Crucial is to prove that there is an index such that the damped iteration (4.122) turns into a full step iteration. Furthermore, Theorem 4.33 requires the transposed Dennis-Moré to hold. We will show that this is indeed true.

For the sake of simplicity we will only consider the application of the nonlinearity bound predictor which is characterized by the choice (4.161). No post-purifying will be taken into account. Furthermore, a step size $\lambda_{l,j}$ is accepted if simple monotonicity holds, i.e.,

$$T(x_l + \lambda_{l,j}\delta x_l | P_{l,\bar{k}_l} A^{-1}_{l,\bar{k}_l}) < T(x_l | P_{l,\bar{k}_l} A^{-1}_{l,\bar{k}_l}). \tag{4.165}$$

Theorem 4.44 *Suppose Assumption 4.24 holds for F and let $J_* := F'(x_*)$. Additionally, assume that the affine covariant nonlinearity bound*

$$2\|J_*^{-1}\big(F(y) - F(x) - F'(x)(y-x)\big)\|_2 < \bar{\omega}\|y-x\|_2^2 \quad \forall x, y \in \mathcal{D} \tag{4.166}$$

holds. Consider the iteration (4.122)*. The Jacobian approximations are constructed according to Algorithm 4.4 and 4.5. Let the step sizes λ_l be determined by means of the nonlinearity bound predictor given via* (4.161) *and the corrector* (4.158)*. Step sizes are accepted if simple monotonicity* (4.165) *holds. Assume that the sequence of iterates $\{x_l\}$ generated by the iteration* (4.122) *is well defined and satisfies $\lim_{l\to\infty} x_l = x_*$ with $x_l \neq x_*$ for all l. Furthermore, suppose that the sequence of all Jacobian approximations $\{A_i\}$, $i \in \{l, (l,k)\}$, converges to a nonsingular matrix $A_* \in \mathbb{R}^{n\times n}$. Then,*

I) there is an index $\underline{l}$ such that $\lambda_l = 1$ for all $l \geqslant \underline{l}$.

If additionally the number of purifying updates is finite then

II) the convergence is superlinear.

Proof. We abbreviate $F_l := F(x_l)$ and $J_l := F'(x_l)$. Since the sequence of all Jacobian approximations $\{A_i\}$ converges to a nonsingular matrix A_* there is an index l_1 such that A_l and $A_{l,k}$ are nonsingular for $l \geqslant l_1$, $0 \leqslant k \leqslant \bar{k}_l$ where $\bar{k}_l$ is the final purifying index at step l. Furthermore, there are constants Θ_1 and Θ_2 such that

$$\|A_i^{-1}J_*\|_2 \leqslant \Theta_1, \qquad \|J_*^{-1}A_i\|_2 \leqslant \Theta_2 \tag{4.167}$$

for all $i \in \{l, (l,k)\}$ with $l \geqslant l_1$ and $0 \leqslant k \leqslant \bar{k}_l$ holds.

I) Let $0 \leqslant k \leqslant \bar{k}_l$. By means of (4.167) we obtain for $\overline{\delta x}_{l,k}$ and $l \geqslant l_1$

$$\|\overline{\delta x}_{l,k}\|_2 \leqslant \|A_{l,k}^{-1}J_*\|_2\|J_*^{-1}F(x_l)\|_2 \leqslant \Theta_1 \cdot \|J_*^{-1}F(x_l)\|_2.$$

Since convergence of the iterates to x_* is assumed this implies that

$$\lim_{l\to\infty} \overline{\delta x}_{l,k} = 0. \tag{4.168}$$

Analogously, it holds that

$$\lim_{l\to\infty} \delta x_l = 0. \tag{4.169}$$

Consider the nonlinearity bound predictor

$$\lambda_{l,0} = \min\left(1, \frac{|1-\alpha_{l,\bar{k}_l}|^2}{[\omega]\cdot\|\overline{\delta x}_{l,\bar{k}_l}\|_2}\right)$$

where $[\omega]$ is chosen according to (4.161) as

$$[\omega] = 2\frac{\|P_{l,\bar{k}_l}A_{l,\bar{k}_l}^{-1}(F_{l-1} - F_l + \lambda_{l-1}J_l\delta x_{l-1})\|_2}{\lambda_{l-1}^2\|\delta x_{l-1}\|_2^2}.$$

By means of the nonlinearity bound (4.166) and the bounds (4.167) we obtain for $l > l_1$,

$$[\omega] \leqslant \Theta_1 \cdot \bar{\omega}.$$

If the predictor is computed the condition (4.164) is fulfilled. Hence,

$$\frac{|1-\alpha_{l,\bar{k}_l}|^2}{[\omega]\cdot\|\overline{\delta x}_{l,\bar{k}_l}\|_2} \geqslant \frac{\varepsilon^2}{\Theta_1\cdot\bar{\omega}\cdot\|\overline{\delta x}_{l,\bar{k}_l}\|_2}.$$

Since (4.168) holds this implies that there is an index $l_2 > l_1$ such that

$$\lambda_{l,0} = 1 \qquad \forall l \geqslant l_2.$$

According to the results of Theorem 4.5 we have

$$\frac{T(x_l+\lambda\delta x_l|P_{l,\bar{k}_l}A_{l,\bar{k}_l}^{-1})}{T(x_l|P_{l,\bar{k}_l}A_{l,\bar{k}_l}^{-1})} = \big(1-\lambda+\overline{\mu}_{l,\bar{k}_l}(\lambda)\big)^2$$

where

$$\overline{\mu}_{l,\bar{k}_l}(\lambda) = -\frac{\overline{\delta x}_{l,\bar{k}_l}^T}{\|\overline{\delta x}_{l,\bar{k}_l}\|_2^2}A_{l,\bar{k}_l}^{-1}\big(F(x_l+\lambda\delta x_l)-F_l-\lambda J_l\delta x_l\big).$$

By means of the first bound in (4.167), by the bound (4.164) and the nonlinearity bound (4.166) we obtain for $l \geqslant l_1$,

$$\begin{aligned}
\overline{\mu}_{l,\bar{k}_l}(\lambda) &= -\frac{\overline{\delta x}_{l,\bar{k}_l}^T}{\|\overline{\delta x}_{l,\bar{k}_l}\|_2^2}A_{l,\bar{k}_l}^{-1}\big(F(x_l+\lambda\delta x_l)-F_l-\lambda J_l\delta x_l\big)\\
&\leqslant |\overline{\mu}_{l,\bar{k}_l}(\lambda)| = \frac{\|P_{l,\bar{k}_l}A_{l,\bar{k}_l}^{-1}\big(F(x_l+\lambda\delta x_l)-F_l-\lambda J_l\delta x_l\big)\|_2}{\lambda^2\|\overline{\delta x}_{l,\bar{k}_l}\|_2\|\delta x_l\|_2}\|\delta x_l\|_2\lambda^2\\
&\leqslant \tfrac{1}{2}|1-\alpha_{l,\bar{k}_l}|^{-1}\Theta_1\bar{\omega}\|\delta x_l\|_2\lambda^2\\
&\leqslant \tfrac{1}{2}\cdot\varepsilon^{-1}\cdot\Theta_1\cdot\bar{\omega}\|\delta x_l\|_2\lambda^2 =: \tfrac{1}{2}\Omega\|\delta x_l\|_2\lambda^2.
\end{aligned}$$

Hence, for the relative change of the level function it holds that

$$\frac{T(x_l+\lambda\delta x_l|P_{l,\bar{k}_l}A_{l,\bar{k}_l}^{-1})}{T(x_l|P_{l,\bar{k}_l}A_{l,\bar{k}_l}^{-1})} \leqslant q_l(\lambda) \quad \forall l \geqslant l_1 \tag{4.170}$$

where

$$q_l(\lambda) := \big(1-\lambda+\tfrac{1}{2}\Omega\|\delta x_l\|_2\lambda^2\big)^2.$$

Since the corrections δx_l converge to zero there is an index $\underline{l} \geqslant l_2$ such that for all $l \geqslant \underline{l}$ the predictor $\lambda_{l,0}$ is equal to one and it holds that

$$p_l(\lambda_{l,0}) < 1$$

which by (4.170) implies that simple monotonicity is given and hence $\lambda_l = \lambda_{l,0} = 1\ \forall l \geqslant \underline{l}$.

II) Since it is assumed that only a finite number of purifying updates is performed there is an index $l_3 > \underline{l}$ such that $\lambda_l = 1$ and $\bar{k}_l = 0$ for $l \geqslant l_3$. No invocation of a purifying process for $l \geqslant l_3$ implies that each A_l is defined via a descent update of $A_{l,0}$ which by Algorithm 4.4 is given by A_{l-1}. Therefore, $\overline{\delta x}_{l,0}$ reads as follows

$$\overline{\delta x}_{l,0} = -A_{l-1}^{-1}F_l.$$

By means of the matrix representation of the descent update (4.125) we obtain

$$\|I - A_{l-1}^{-1} A_l\|_2 = \frac{\|(A_{l-1}^{-1} F_l)^T (I - A_{l-1}^{-1} J_l)\|_2}{\|A_{l-1}^{-1} F_l\|_2}.$$

Since the sequence of Jacobian approximations converges to A_* we have

$$\lim_{l \to \infty} \|I - A_{l-1}^{-1} A_l\|_2 = 0$$

and hence

$$\lim_{l \to \infty} \frac{\|(A_{l-1}^{-1} F_l)^T (I - A_{l-1}^{-1} J_l)\|_2}{\|A_{l-1}^{-1} F_l\|_2} = 0.$$

Furthermore,

$$\frac{\|(A_{l-1}^{-1} F_l)^T (I - A_{l-1}^{-1} J_*)\|_2}{\|A_{l-1}^{-1} F_l\|_2} \leqslant \frac{\|(A_{l-1}^{-1} F_l)^T (I - A_{l-1}^{-1} J_l)\|_2}{\|A_{l-1}^{-1} F_l\|_2} + \Theta_1 \cdot \|I - J_*^{-1} J_l\|_2$$

and by means of the convergence of the iterates to x_* we obtain

$$\lim_{l \to \infty} \frac{\|(A_{l-1}^{-1} F_l)^T (I - A_{l-1}^{-1} J_*)\|_2}{\|A_{l-1}^{-1} F_l\|_2} = 0.$$

This means that the transposed Dennis-Moré condition (4.94) for the sequences $\{A_l\}$ and $\{A_{l-1}^{-1} F_l\}$, $l \geqslant l_3$, holds. By Proposition 4.34 the sequence $\{A_{l-1}^{-1} F_l\}$, $l \geqslant l_3$, also has the affine covariant residual property 4.31. Hence, all the conditions of Theorem 4.33 are fulfilled for $l \geqslant l_3$ which yields superlinear convergence of the sequence $\{x_l\}$ to x_*.

■

Chapter 5

A Global Convergence Result for a Newton-like Iteration

There is no global convergence result if one solely employs the natural level function concept to determine step sizes in a damped Newton iteration. Such is also true for the concepts of the projected natural level function and the approximate projected natural level function which are introduced and discussed in this work. This is due to the fact that every step descent is measured in a different metric and cycles in the iterates cannot be excluded.

However, employing the general level function (2.10), i.e.,

$$T(x|A) := \tfrac{1}{2}\|AF(x)\|_2^2, \quad A \in \mathbb{R}^{n\times n}, \tag{5.1}$$

for a *fixed nonsingular* choice of A global convergence results for a damped Newton iteration are available, cf. Theorem 3.13 in [11].

In this section we will provide a global convergence result of the type presented in the aforementioned Theorem 3.13, however, for an *approximate* damped Newton iteration

$$x_{l+1} = x_l + \lambda_l \delta x_l, \quad \delta x_l = -B_l^{-1} F_l, \; F_l := F(x_l),$$

where the matrices B_l are to be considered as approximations to the respective Jacobians $J_l := F'(x_l)$. The basic techniques used in the proof of our global convergence statement are adopted from the proof of the above mentioned Theorem 3.13. However, due to the fact that we consider an approximate damped Newton iteration some extra care is necessary. For the proof we will define a polynomial model $q_l(\lambda)$ as an estimate for the relative change of $T(x|A)$ at x_l in the direction of δx_l. By means of this model the step sizes λ_l will be determined. For the development of our polynomial model it is crucial that the approximate correction δx_l fulfills

$$\begin{aligned} \frac{d}{d\lambda} T(x_l + \lambda \delta x_l|A)_{|\lambda=0} &= \operatorname{grad} T(x_l|A)\delta x_l = \left(AF_l\right)^T A J_l \delta x_l \\ &= -\|AF_l\|_2^2. \end{aligned} \tag{5.2}$$

This can be achieved by simple scaling of some given direction $\overline{\delta x_l}$ provided $\overline{\delta x_l}$ and $\operatorname{grad} T(x_l|A)^T$ are not perpendicular, cf. the definition of the descent approximation from (4.6) in the context of

the APNLF. However, as we will see we can also use a generalization of the descent update from Section 4.3 to obtain a correction δx_l which fulfills (5.2). We will combine the generalized descent update with a generalization of the purifying techniques from Section 4.2 to provide a sufficiently good approximation quality of the matrices B_l.

As a first preparation for our global convergence statement we will develop a polynomial model $p(\lambda)$ which serves as a basis for $q_l(\lambda)$. We introduce this second model $p(\lambda)$ also for the reason that it may be exploited to provide the basics for a broader range of step size control algorithms. E.g., we will see that it may be used for a step size control in the context of an approximate natural level function (without projection)—see Remark 5.3.

The model $p(\lambda)$ depends on a rather general nonlinearity bound:

For nonsingular W, $U \in \mathbb{R}^{n\times n}$ let

$$2 \cdot \|W\big(F(y) - F(x) - F'(x)(y-x)\big)\|_2 \leqslant \omega \|U(y-x)\|_2^2 \quad \forall x, y \in \mathcal{D}. \tag{5.3}$$

As a further generalization W and U may depend on x. If this is the case we assume that W and U are continuous in x and that $W(x)$, $U(x)$ are nonsingular for each $x \in \mathcal{D}$.

This bound is innately neither affine covariant nor affine contravariant. It depends on the choice of W and U which concept is favored. For example $W = W(x) = F'(x)^{-1}$ and $U = I$ leads to an affine covariant bound whereas $W = I$ and $U = U(x) = F'(x)$ to an affine contravariant one. This is on purpose to ensure compatibility of the polynomial $p(\lambda)$ with both concepts. Furthermore one may consider the above bound only for an appropriate subset $\tilde{\mathcal{D}}$ of $\mathcal{D}$ which leads to a locally defined bound. This way an adaptive choice of W and U is possible—see again Remark 5.3.

The polynomial model $p(\lambda)$ reads as follows.

Theorem 5.1 (Polynomial model) *Suppose that F fulfills Assumption 2.1 and let $x \in \mathcal{D}$ such that $F(x) \neq 0$. Abbreviate $J := F'(x)$. Assume that the nonlinearity bound* (5.3) *holds. For given nonsingular $A \in \mathbb{R}^{n\times n}$ consider the general level function* (5.1)*. Let $B \in \mathbb{R}^{n\times n}$ be nonsingular such that $\delta x := -B^{-1}F(x)$ fulfills*

$$\frac{d}{d\lambda} T(x + \lambda\delta x|A)_{|\lambda=0} = -\|AF(x)\|_2^2. \tag{5.4}$$

Define

$$\eta := \frac{\|A\big(B-J\big)\delta x\|_2}{\|AF(x)\|_2}, \qquad \tilde{h} := \omega \cdot \|AW^{-1}\|_2 \cdot \frac{\|U\delta x\|_2^2}{\|AF(x)\|_2}$$

and

$$\Lambda := \{\lambda \in (0,1] \mid x + \lambda\delta x \in \mathcal{D}\}. \tag{5.5}$$

Then for $\lambda \in \Lambda$ one has

$$\begin{aligned} T(x + \lambda\delta x|A) &\leqslant \Big[\big(1 - \lambda + \tfrac{1}{2}\tilde{h}\lambda^2\big)^2 + \eta^2\lambda^2 + \tilde{h}\eta\lambda^3\Big] T(x|A) \\ &=: p(\lambda) T(x|A). \end{aligned} \tag{5.6}$$

For the polynomial p it holds that

I) $\frac{d}{d\lambda}p(\lambda)_{|\lambda=0} = -2$,

II) p *is strictly convex on* $[0,1]$ *and has a unique minimizer* $\tilde{\lambda}$ *in* $[0,1]$ *with* $0 < \tilde{\lambda} \leqslant \min\big(1, 1/\tilde{h}\big)$ *where* $1/\tilde{h} := \infty$ *if* $\omega = 0$.

Proof. Consider $\lambda \in \Lambda$. For ease of writing we use

$$\phi(\lambda) := A\big(F(x+\lambda\delta x) - F(x) - \lambda J\delta x\big).$$

Since (5.4) holds, i.e.,

$$\big(AF(x)\big)^T AJ\delta x = -\big(AF(x)\big)^T AF(x)$$

and

$$B\delta x = -F(x)$$

we have

$$\big(AF(x)\big)^T A(B-J)\delta x = 0.$$

All three statements are exploited in the following without any explicit reference.

$$\begin{aligned}
\|AF(x+\lambda\delta x)\|_2^2 &= \big(AF(x+\lambda\delta x)\big)^T AF(x+\lambda\delta x)\\
&= \big(AF(x)\big)^T AF(x+\lambda\delta x)\\
&\quad + \Big[A\big(F(x+\lambda\delta x) - F(x)\big)\Big]^T AF(x+\lambda\delta x)\\
&=: a + b_1^T b_2.
\end{aligned}$$

For a we have

$$\begin{aligned}
a &= (AF(x))^T AF(x+\lambda\delta x)\\
&= \big(AF(x)\big)^T A\big(F(x+\lambda\delta x) - F(x) - J\delta x\big)\\
&= \big(AF(x)\big)^T \Big[A\big(F(x+\lambda\delta x) - F(x) - \lambda J\delta x\big) + (1-\lambda)AF(x)\Big]\\
&= (1-\lambda)\big(AF(x)\big)^T AF(x) + \big(\phi(\lambda)\big)^T AF(x).
\end{aligned}$$

For b_1 we obtain

$$\begin{aligned}
b_1 &= A\big(F(x+\lambda\delta x) - F(x)\big)\\
&= A\big(F(x+\lambda\delta x) - F(x) - \lambda J\delta x\big) + \lambda AJ\delta x\\
&= \phi(\lambda) + \lambda AJ\delta x
\end{aligned}$$

and for b_2,

$$\begin{aligned}
b_2 &= AF(x+\lambda\delta x)\\
&= A\big(F(x+\lambda\delta x) - F(x) - \lambda J\delta x - B\delta x + \lambda J\delta x\big)\\
&= \phi(\lambda) - \lambda A(B-J)\delta x + (1-\lambda)AF(x).
\end{aligned}$$

Combining these results and rearranging the terms the sum $a + b_1^T b_2$ reads as follows

$$\begin{aligned}
a + b_1^T b_2 = (1-\lambda)\big(AF(x)\big)^T AF(x) &+ \lambda(1-\lambda)\big(AF(x)\big)^T AJ\delta x\\
&+ (1-\lambda)\big(\phi(\lambda)\big)^T AF(x) + \big(\phi(\lambda)\big)^T AF(x) + \big(\phi(\lambda)\big)^T \phi(\lambda)\\
&- \lambda^2\big(AJ\delta x\big)^T A(B-J)\delta x - \lambda\big(\phi(\lambda)\big)^T A(B-J)\delta x + \lambda\big(\phi(\lambda)\big)^T AJ\delta x.
\end{aligned}$$

Recall that

$$\big(AF(x)\big)^T AJ\delta x = -\big(AF(x)\big)^T AF(x)$$

and note that

$$-\big(AJ\delta x\big)^T A(B-J)\delta x = \big(A(B-J)\delta x\big)^T A(B-J)\delta x.$$

So with an additional adding and subtracting of $\lambda\big(\phi(\lambda)\big)^T AB\delta x = -\lambda\big(\phi(\lambda)\big)^T AF(x)$ we have

$$\begin{aligned} a + b_1^T b_2 = (1-\lambda)^2\big(AF(x)\big)^T AF(x) + 2(1-\lambda)\big(\phi(\lambda)\big)^T AF(x) + \big(\phi(\lambda)\big)^T\phi(\lambda) \\ + \lambda^2\|A(B-J)\delta x\|_2^2 - 2\lambda\big(\phi(\lambda)\big)^T A(B-J)\delta x. \end{aligned} \tag{5.7}$$

By the nonlinearity bound (5.3) and the definition of $\tilde{h}$ it holds that

$$\|\phi(\lambda)\|_2 \leqslant \tfrac{1}{2}\tilde{h}\lambda^2\|AF(x)\|_2.$$

Thus, by means of this estimate, the relation (5.7), and by means of the Cauchy-Schwarz inequality we obtain

$$\begin{aligned} \|AF(x+\lambda\delta x)\|_2^2 \leqslant (1-\lambda)^2\|AF(x)\|_2^2 + (1-\lambda)\lambda^2\tilde{h}\|AF(x)\|_2^2 + \tfrac{1}{4}\tilde{h}^2\lambda^4\|AF(x)\|_2^2 \\ + \lambda^2\frac{\|A(B-J)\delta x\|_2^2}{\|AF(x)\|_2^2}\|AF(x)\|_2^2 + \tilde{h}\lambda^3\frac{\|A(B-J)\delta x\|_2}{\|AF(x)\|_2}\|AF(x)\|_2^2. \end{aligned}$$

According to the definition of η and $p(\lambda)$ and by means of some minor rearranging we can write the above relation as

$$\begin{aligned} \|AF(x+\lambda\delta x)\|_2^2 &\leqslant \Big[\big(1-\lambda+\tfrac{1}{2}\tilde{h}\lambda^2\big)^2 + \eta^2\lambda^2 + \tilde{h}\eta\lambda^3\Big]\|AF(x)\|_2^2 \\ &= p(\lambda)\|AF(x)\|_2^2 \end{aligned}$$

which is just (5.6).

For the remaining part of the proof let $\lambda \in [0,1]$. The first derivative of p at λ is

$$\frac{d}{d\lambda}p(\lambda) = 2(1-\lambda+\tfrac{1}{2}\tilde{h}\lambda^2)(-1+\tilde{h}\lambda) + 2\eta^2\lambda + 3\tilde{h}\eta\lambda^2.$$

Setting $\lambda = 0$ leads to $\frac{d}{d\lambda}p(\lambda)_{|\lambda=0} = -2$. From

$$\frac{d^2}{d\lambda^2}p(\lambda) = 2(\tilde{h}\lambda-1)^2 + 2\tilde{h}(1-\lambda+\tfrac{1}{2}\tilde{h}\lambda^2) + 2\eta^2 + 6\tilde{h}\eta\lambda$$

we derive that p is strictly convex on $[0,1]$ and $\frac{d}{d\lambda}p$ is strictly increasing on $[0,1]$. Together with $\frac{d}{d\lambda}p(\lambda)_{|\lambda=0} < 0$ and $\tilde{h}, \eta < \infty$ this yields a unique minimizer $\tilde{\lambda} > 0$ of p in $[0,1]$. Since the term $2\eta^2\lambda + 3\tilde{h}\eta\lambda^2$ is nonnegative for $\lambda = 1/\tilde{h}$ the minimizer is bounded from above by $\min\big(1, 1/\tilde{h}\big)$.

■

By the quantity η the influence of the deviation from the Newton correction is expressed because

$$\eta = 0 \quad \Leftrightarrow \quad \delta x = -F'(x)^{-1}F(x).$$

For an evaluation of η the direct tangent evaluation $F'(x)\delta x$ is required. This can easily be done via the forward mode of Automatic Differentiation. Note that for $\eta = 0$, $W = W(x) = F'(x)^{-1}$ and $U = I$ the above polynomial simplifies to the the square of the polynomial from Theorem 2.10 if additionally the Lipschitz condition (2.23) is considered instead of the above nonlinearity bound (5.3) for this specific choices of W and U, i.e. the nonlinearity bound (3.3), and if A is chosen

as $A = F'(x)^{-1}$. An analogous simplification is true in the context of affine contravariance for the choices $W = I$, $U(x) = F'(x)$ and $A = I$. To avoid the introduction of new notation just for the purposes of comparison we omit details here. The interested reader may be referred to Theorem 3.7 in [11]. Summarizing, the polynomial model p provides a generalization of existing polynomial models from the literature in the context of both affine invariance concepts.

Remark 5.2 The minimizer $\tilde{\lambda}$ of p in $[0, 1]$ can be stated explicitly. We omit the formula here since it is lengthy and does not provide any readily identifiable further insight. □

Remark 5.3 If we only consider a subset $\tilde{\mathcal{D}}$ of $\mathcal{D}$ in (5.3), e.g., for given $x \in \mathcal{D}$ let $y \in \mathcal{D}$ with $y - x = \lambda\delta x$, $\lambda \in [0, 1]$, we obtain a local nonlinearity bound. If these bounds are employed iteratively one may also choose W and U adaptively. Consider for example the context of an approximate natural level function where $\tilde{B}_l$ is some nonsingular approximation for $F'(x_l)$ and B_l a second nonsingular approximation for $F'(x_l)$ which additionally fulfills (5.2) for the choice $A = \tilde{B}_l$. Choosing $W_l = \tilde{B}_l$ at x_l and $U_l \equiv I$ opens the door to have meaningful theoretical quantities available which can also be estimated in a reasonable way, cf. the bound (4.152) and its estimate (4.157). □

For the upcoming global convergence result we must ensure that the deviation of δx_l from the Newton correction at x_l is uniformly bounded. To meet this condition, we will employ an adaption of the purifying techniques from Section 4.2. Recall that we also have to meet the requirement (5.2). Therefore, we will use an adaption of the descent update from Section 4.3. Let $x \in \mathcal{D}$ such that $F(x) \neq 0$ and $F'(x)$ is nonsingular. With the abbreviations $F := F(x)$ and $J := F'(x)$ and with δx_k determined in analogy to $\overline{\delta x}_k$ in (4.52) the updates we are going to use look as follows:

- descent:

$$B_{k+1} = B_k - \frac{F(AF)^T A(B_k - J)}{\|AF\|_2^2} \tag{5.8a}$$

- Newton-philic:

$$B_{k+1} = B_k - \frac{(B_k - J)\delta x_k \big(A(B_k - J)\delta x_k\big)^T A(B_k - J)}{\|A(B_k - J)\delta x_k\|_2^2} \tag{5.8b}$$

Remark 5.4 The above updates are in close relationship to Schlenkrich's residual update

$$S_{k+1} = S_k - \frac{FF^T(S_k - J)}{\|F\|_2^2}$$

and transposed tangent Broyden update

$$S_{k+1} = S_k - \frac{(S_k - J)s_k\big((S_k - J)s_k\big)^T(S_k - J)}{\|(S_k - J)s_k\|_2^2}, \quad s_k \neq 0 \text{ s.t. } S_k s_k = \begin{cases} -F & \text{if } S_k \text{ is nonsingular} \\ 0 & \text{if } S_k \text{ is singular,} \end{cases}$$

respectively: Applying Schlenkrich's updates to $G(x) := AF(x)$ and generating approximations $S_{l,k}$ hereby then the relation $AB_{l,k} = S_{l,k}$ holds if $AB_{l,0} = S_{l,0}$. Hence, regarding global convergence one may also refer to the results in [28]. However, the step size control exploited in [28] is not based on a polynomial such as $p(\lambda)$ from Theorem 5.1. Thus, we provide an alternative approach. □

If B_{k+1} is constructed via the descent update then

$$(AF)^T AB_{k+1} = (AF)^T AJ \tag{5.9}$$

and hence for nonsingular B_{k+1} (5.4) is true. Due to heredity all following approximations constructed via the Newton-philic update also fulfill (5.9). If such a matrix is nonsingular then (5.4) is true too. Note that $\operatorname{grad} T(x|A)$ is given via $(AF)^T AJ$ and hence does not depend on any of the approximations B_k. Therefore, we do not consider adapted versions of the duophilic and the gradientphilic update, (4.49) and (4.51), respectively. The basic purifying process at an iterate x_l is as follows:

Algorithm 5.1 (Purifying process w.r.t $T(x|A)$ at $x_l \in \mathcal{D}$ with $F'(x_l)$ nonsingular)

---

given: $B_{l,0} \in \mathbb{R}^{n\times n}$, $F_l := F(x_l) \neq 0$, $J_l := F'(x_l)$ nonsingular,
set $k = 0$
while ($B_{l,k}$ singular) || ($B_{l,k}$ nonsingular && $B_{l,k}^{-1}F_l \neq J_l^{-1}F_l$) **do**
  if $k = 0$ **then**
    construct $B_{l,k+1}$ via the descent update (5.8a)
  else
    determine $\delta x_{l,k} \neq 0$ according to

$$B_{l,k}\delta x_{l,k} = \begin{cases} -F_l & \text{if } B_{l,k} \text{ is nonsingular} \\ 0 & \text{if } B_{l,k} \text{ is singular} \end{cases}$$

    construct $B_{l,k+1}$ via the Newton-philic update (5.8b)
  end if
  set $k = k + 1$
end while

The adapted Newton-philic update (5.8a) is a specific instance of the basic purifying update, cf. (4.39). Hence, by Proposition 4.16 the above algorithm terminates after a finite number of steps providing the Newton correction. Also, every constructed approximation $B_{l,k}$ fulfills an indexed version of (5.9), i.e.,

$$\big(AF_l\big)^T AB_{l,k} = (AF_l)^T J_l. \tag{5.10}$$

So there will be an index $\bar{k}_l$ such that $B_{l,\bar{k}_l}$ is nonsingular and of sufficient approximation quality. This vague statement will be concretized in the proof of Theorem 5.5. We use $B_{l,\bar{k}_l}$ to compute the actual correction δx_l, i.e.,

$$\begin{aligned} B_l &:= B_{l,\bar{k}_l}, \\ \delta x_l &= -B_l^{-1}F_l. \end{aligned} \tag{5.11}$$

Since the quantities W and U from (5.3) may depend on x and hence on an iterate x_l we use the notation $W_{(l)}$ and $U_{(l)}$ to indicate that possible dependency. Also, we introduce indices in the definition of η, i.e.,

$$\eta_l := \frac{\|A\big(B_l - J_l\big)\delta x_l\|_2}{\|AF_l\|_2}. \tag{5.12}$$

Verify that

$$\|U_{(l)}(B_l^{-1} - J_l^{-1})F_l\|_2 + \|U_{(l)}J_l^{-1}F_l\|_2 \geqslant \|U_{(l)}\delta x_l\|_2.$$

Also, it holds that

$$\|U_{(l)}J_l^{-1}F_l\|_2 \leqslant \|U_{(l)}J_l^{-1}A^{-1}\|_2\|AF_l\|_2.$$

Hence,

$$\begin{aligned}\widehat{h}_l &:= \omega \cdot \|AW_{(l)}^{-1}\|_2 \cdot \left[\frac{\|U_{(l)}(B_l^{-1}-J_l^{-1})F_l\|_2}{\|AF_l\|_2} + \|U_{(l)}J_l^{-1}A^{-1}\|_2\right] \\ &\quad \cdot \left[\|U_{(l)}(B_l^{-1}-J_l^{-1})F_l\|_2 + \|U_{(l)}J_l^{-1}F_l\|_2\right] \\ &\geqslant \omega \cdot \|AW_{(l)}^{-1}\|_2 \cdot \frac{\|U_{(l)}\delta x_l\|_2^2}{\|AF_l\|_2} =: \tilde{h}_l.\end{aligned} \tag{5.13}$$

So introducing indices in the definition of $p(\lambda)$ we substitute $\widehat{h}_l$ for $\tilde{h}_l$ to obtain the polynomial model $q_l(\lambda)$ on which the following result is based.

Theorem 5.5 (Global convergence) *Suppose F fulfills Assumption 2.1 and additionally let the Jacobian $F'(x)$ be nonsingular for all $x \in \mathcal{D}$. Assume that the nonlinearity bound (5.3) holds. Consider the general level function (5.1) for nonsingular $A \in \mathbb{R}^{n\times n}$ and let $G(x|A)$ be the level set of the general level function at x, i.e., $G(x|A) := \{z \in \mathcal{D} \mid T(z|A) \leqslant T(x|A)\}$. For some $x_0 \in \mathcal{D}$ let $\mathcal{D}_0$ denote the path-connected component of $G(x_0|A)$ which contains x_0. Assume that $\mathcal{D}_0$ is compact. Let $B_{0,0} \in \mathbb{R}^{n\times n}$ nonsingular and nonnegative constants ζ_1, ζ_2 and ζ_3 be given. Consider the iteration*

$$x_{l+1} = x_l + \lambda_l \delta x_l, \quad \delta x_l = -B_l^{-1}F_l,\ F_l := F(x_l),$$

with λ_l being the unique minimizer in $[0,1]$ of

$$q_l(\lambda) := \left(1 - \lambda + \tfrac{1}{2}\widehat{h}_l\lambda^2\right)^2 + \eta_l^2\lambda^2 + \widehat{h}_l\eta_l\lambda^3.$$

The matrix B_l is given as in (5.11). The quantities η_l and $\widehat{h}_l$ are defined as in (5.12) and (5.13).

Then for each l with $F_l \neq 0$ there exists an index $\bar{k}_l \leqslant n$ such that

I) B_l is nonsingular, hence, δx_l is well defined,

II) it holds that

$$\|\left(B_l^{-1} - J_l^{-1}\right)F_l\|_2 \leqslant \zeta_1, \qquad \frac{\|\left(B_l^{-1} - J_l^{-1}\right)F_l\|_2}{\|AF_l\|} \leqslant \zeta_2, \qquad \eta_l \leqslant \zeta_3$$

where $J_l := F'(x_l)$,

III) λ_l is well defined and $\lambda_l \geqslant \varepsilon > 0$ with ε independent of l,

IV) there is a constant $0 \leqslant C_\varepsilon < 1$ independent of l such that

$$\begin{aligned}T(x_l + \lambda_l\delta x_l|A) &\leqslant q_l(\lambda_l)T(x_l|A) \\ &\leqslant C_\varepsilon T(x_l|A) < T(x_l|A).\end{aligned}$$

This implies that the sequence x_l converges to some x_* with $F(x_*) = 0$.

The constant ε depends on $\mathcal{D}_0$ and ζ_1, ζ_2 and ζ_3.

Proof. We define for $x_l \in \mathcal{D}$ the set $\mathcal{D}_l$ in accordance with $\mathcal{D}_0$.

The proof is by induction. Assume that $F_0 \neq 0$. Since J_0 is nonsingular and by means of Proposition 4.16 and Algorithm 5.1 there is an index $0 \leqslant \bar{k}_0 \leqslant n$ such that $B_{0,\bar{k}_0} =: B_0$ is nonsingular and (5.10) holds. Hence, $\delta x_0 = -B_0^{-1}F_0$ fulfills (5.2) for $l = 0$. Since also the nonlinearity bound (5.3) is assumed to hold, we can exploit the results of Theorem 5.1. Introducing indices in (5.5) and (5.6) we obtain

$$T(x_0 + \lambda\delta x_0|A) \leqslant p_0(\lambda)T(x_0|A) \quad \forall\lambda \in \Lambda_0.$$

Recall from (5.13) that $\widehat{h}_0 \geqslant \tilde{h}_0$. Hence, substituting $\widehat{h}_0$ for $\tilde{h}_0$ in $p_0(\lambda)$ yields

$$p_0(\lambda) \leqslant q_0(\lambda) \quad \forall\lambda \in \Lambda_0$$

and therefore

$$T(x_0 + \lambda\delta x_0|A) \leqslant q_0(\lambda)T(x_0|A) \quad \forall\lambda \in \Lambda_0.$$

Also, we may argue like in the proof of Theorem 5.1 that $\frac{d}{d\lambda}q_0(\lambda)_{|\lambda=0} = -2$ and that q_0 is strictly convex on $[0,1]$ and has a unique minimizer $\lambda_0 > 0$ in $[0,1]$. We show by contradiction that $x_0 + \lambda\delta x_0 \in \mathcal{D}_0$ for all $0 \leqslant \lambda \leqslant \lambda_0$. For this, assume it is not the case. Since $\mathcal{D}$ is open and nonempty and due to the compactness of $\mathcal{D}_0$ the set $Y_0 := \{0 < \lambda \leqslant \lambda_0 \,|\, x_0 + \lambda\delta x_0 \in \mathcal{D} \setminus \mathcal{D}_0\}$ is nonempty. By assumption $\mathcal{D}$ is also convex. This means that for any $\lambda \in Y_0$ we have $x_0 + s\delta x_0 \in \mathcal{D}$, $0 \leqslant s \leqslant \lambda$. And for any such $s > 0$ we obtain by the above stated properties of q_0 the estimate $q_0(s) < q_0(0) = 1$. Consequently,

$$T(x_0 + s\delta x_0|A) \leqslant q_0(s)T(x_0|A) < T(x_0|A), \quad 0 < s \leqslant \lambda, \quad \lambda \in Y_0.$$

But from this it follows that $x_0 + \lambda\delta x_0 \in \mathcal{D}_0$ for all $\lambda \in Y_0$ which contradicts the definition of Y_0. Thus, we can conclude that $x_0 + \lambda\delta x_0 \in \mathcal{D}_0$ for all $0 \leqslant \lambda \leqslant \lambda_0$ which means that

$$T(x_0 + \lambda\delta x_0|A) < T(x_0) \quad \text{for all } 0 < \lambda \leqslant \lambda_0.$$

Hence, in particular

$$T(x_1|A) < T(x_0|A)$$

and therefore

$$\mathcal{D}_1 \subset \mathcal{D}_0.$$

We turn to the question whether there is an $\varepsilon > 0$ such that $\lambda_0 \geqslant \varepsilon$. Recall that W and U in the nonlinearity bound (5.3) may depend continuously on x. If this is not the case then $W(x) \equiv W \in \mathbb{R}^{n\times n}$ and $U(x) \equiv U \in \mathbb{R}^{n\times n}$, respectively. Anyway, the compactness of $\mathcal{D}_0$ yields for some positive constants C_1, C_2 and C_3,

$$\begin{aligned}
\max_{x\in\mathcal{D}_0}\|U(x)F'(x)^{-1}F(x)\|_2 &\leqslant C_1,\\
\max_{x\in\mathcal{D}_0}\|AW(x)^{-1}\|_2 &\leqslant C_2,\\
\max_{x\in\mathcal{D}_0}\|U(x)F'(x)^{-1}A^{-1}\|_2 &\leqslant C_3.
\end{aligned}$$

W.l.o.g. we can also assume that $\bar{k}_0$ is chosen such that for the given nonnegative constants ζ_1, ζ_2 and ζ_3 the bounds

$$\begin{aligned} \|\big(B_0^{-1} - J_0^{-1}\big)F_0\|_2 &\leqslant \zeta_1, \\ \frac{\|\big(B_0^{-1} - J_0^{-1}\big)F_0\|_2}{\|AF_0\|} &\leqslant \zeta_2, \\ \eta_0 &\leqslant \zeta_3 \end{aligned}$$

hold. Hence, by the definition of $\widehat{h}_l$ in (5.13),

$$\widehat{h}_l \leqslant \omega C_2(\zeta_2 + C_3)(\zeta_1 + C_1) =: C_4$$

and for

$$\overline{q}(\lambda) := (1 - \lambda + \tfrac{1}{2}C_4\lambda^2)^2 + \zeta_3^2\lambda^2 + C_4\zeta_3\lambda^3$$

we obtain

$$q_0(\lambda) \leqslant \overline{q}(\lambda) \quad \forall \lambda \in \Lambda_0.$$

Since $\overline{q}$ is of the same structure as p_0 and q_0 the properties stated in paragraph I) and II) of Theorem 5.1 are true for $\overline{q}$ as well. We denote the unique minimizer by ε and define the nonnegative constant C_ε via

$$C_\varepsilon := \overline{q}(\varepsilon) < 1.$$

Exploiting the properties stated in paragraph I) and II) of Theorem 5.1 for q_0 and $\overline{q}$ some elementary considerations yield

$$\lambda_0 \geqslant \varepsilon > 0.$$

Also,

$$q_0(\lambda_0) \leqslant C_\varepsilon \quad \text{and hence} \quad T(x_1|A) \leqslant C_\varepsilon T(x_0|A).$$

Assuming for $l \geqslant 1$ that $x_l \in \mathcal{D}$ with $F_l \neq 0$ and $\mathcal{D}_l \subset \mathcal{D}_0$ holds we argue in the same manner as before to prove that

- there is an index $\bar{k}_l$ such that δx_l is well defined,
- $x_l + \lambda_l \delta x_l = x_{l+1} \in \mathcal{D}_l$ and $\mathcal{D}_{l+1} \subset \mathcal{D}_l$,
- $\lambda_l \geqslant \varepsilon$,
- $T(x_{l+1}|A) \leqslant C_\varepsilon T(x_l|A)$.

With the assumption $\mathcal{D}_l \subset \mathcal{D}_0$ the second of the above relations implies $\mathcal{D}_{l+1} \subset \mathcal{D}_0$. This yields either convergence in a finite number of steps or

$$\lim_{l \to \infty} x_l = x_*$$

completing the proof.

■

Chapter 6

Numerical Experiments

In this chapter we will present numerical experiments to test out our algorithmic realizations of the concept of the projected natural level function (PNLF) and the approximate projected natural level function (APNLF).

For the latter approach we will use the purifying updates and the descent update from Section 4.2 and 4.3, respectively, to obtain an approximation of the Jacobian. This is done in a way as described in Section 4.4 and implemented along the lines of the algorithms in Appendix II.

Step sizes are either determined according to the simple monotonicity or the restricted monotonicity step size strategy from Subsection 3.4.1 and Subsection 3.4.2, respectively, with the modifications discussed in Subsection 4.4.6 in case of the APNLF-algorithm.

As reference method for comparison purposes we will use an algorithm which employs the natural level function (NLF). This algorithm is in its basic functionality very similar to the algorithm `NLEQ1` which is discussed and tested in [26]. We prefer our self written code instead of employing `NLEQ1` directly because the implementation of our NLF-algorithm is identical to the one of the PNLF-algorithm except that step sizes are monitored by the NLF instead of the PNLF and predictor step sizes are always computed via the Deuflhard predictor (3.92). This has the advantage that a different behavior of the algorithms is not related to differing aspects of the implementation but solely to the fact that different concepts are employed.

The algorithms we will test out in this chapter are designed to solve general systems of nonlinear equations

$$F(x) = 0$$

where F fulfills Assumption 2.1, i.e., $F : \mathcal{D} \to \mathbb{R}^n$ is continuously differentiable on $\mathcal{D} \subseteq \mathbb{R}^n$ with $\mathcal{D} \neq \emptyset$ open and convex. Least squares problems like they are discussed in Section 3.3 are not considered.

Algorithmic settings

In the course of their execution the considered algorithms require solutions of linear systems of the form

$$My = b, \quad b \in \mathbb{R}^n, \tag{6.1a}$$

where M is a Jacobian matrix or Jacobian approximation, respectively. Furthermore, for the APNLF-algorithm and also optionally for the PNLF-algorithm linear systems of the form

$$z^T M = w^T, \quad w \in \mathbb{R}^n, \tag{6.1b}$$

arise. We employ an LU-decomposition of M to solve the above linear systems. In the case of the APNLF-algorithm we incorporate the arising rank-1 updates into the current LU-decomposition by means of the update algorithm from [17] which we discussed in Subsection 4.4.5. Throughout, the APNLF-algorithm is started with the Jacobian at the initial guess x_0. This way, affine covariance is ensured for the iterates provided by the APNLF-algorithm.

To run the algorithms concrete values for several constants are required:

- *step size related quantities*
  As initial step size we choose $\lambda_0 = 10^{-2}$. The minimal allowed step size λ_{min} is chosen according to

  $$\lambda_{min} = 10^{-4} \quad \text{(simple monotonicity)}, \qquad \lambda_{min} = 10^{-6} \quad \text{(restricted monotonicity)}.$$

  If restricted monotonicity is used the respective check (3.115), i.e.,

  $$\Big(\lambda \in \Lambda_l(\underline{\eta}, \overline{\eta})\Big) \qquad \text{or} \qquad \Big(\lambda = 1 \quad \text{and} \quad \lambda < \underline{\eta}/[\omega]_l(\lambda)\|\Delta x_l\|_2\Big) \tag{6.2}$$

  with

  $$\Lambda_l\big(\underline{\eta}, \overline{\eta}\big) = \{\lambda \in (0,1] \mid \lambda = \tilde{\eta}/[\omega]_l(\lambda)\|\Delta x_l\|_2, \;\; \tilde{\eta} \in [\underline{\eta}, \overline{\eta}]\}$$

  is considered for the values

  $$\underline{\eta} = \frac{1}{2} \qquad \text{and} \qquad \overline{\eta} = \frac{3}{2}.$$

  Certainly, in case of the APNLF-algorithm δx_l is substituted for Δx_l and $[\omega]_l(\lambda)$ is adapted accordingly, cf. (4.157).

- *scaling threshold*
  If we consider adaptive scaling the scaling matrices from (3.119), i.e.,

  $$\begin{aligned} D_l &= \operatorname{diag}(d_1, \ldots, d_n) \quad \text{where} \quad d_i = \max\Big[\tfrac{1}{2}\big(|x_{l-1,(i)}| + |x_{l,(i)}|\big), \operatorname{thrsh}\Big], \; l > 0, \\ D_0 &= \operatorname{diag}(d_1, \ldots, d_n) \quad \text{where} \quad d_i = \max(|x_{0,(i)}|, \operatorname{thrsh}) \end{aligned} \tag{6.3}$$

  are computed for the value $\operatorname{thrsh} = 10^{-6}$. Notice that in case of adaptive scaling also range space related scaling of linear systems as stated in Section 3.4.3 is taken into account.

- *angle conditions*
  In case of the APNLF-algorithm a purifying process is terminated if the angle conditions (4.124), i.e.,

  $$\angle\big(\delta x_l, -\operatorname{grad} T(x_l | P_{l,k} A_{l,k}^{-1})^T\big) \leqslant \phi \qquad \text{and} \qquad \angle_{est}(\delta x_l, \Delta x_l) \leqslant \psi \tag{6.4}$$

  are fulfilled at a purifying index k for

  $$\phi = \frac{\pi}{6} \mathrel{\hat{=}} 30^\circ \qquad \text{and} \qquad \psi = \frac{\pi}{10} \mathrel{\hat{=}} 18^\circ.$$

For the additional singularity check (4.135),

$$|1 - \alpha_{l,k}| < \varepsilon,$$

we choose $\varepsilon = \frac{1}{2}$.

Regarding the decision whether a descent update is performed or a purifying process is initiated the gradient related angle check and the singularity check are considered for a slightly less restrictive choice of ϕ and ε: $\phi = \frac{\pi}{5} \hat{=} 36°$ and $\varepsilon = \frac{3}{10}$.

- *error tolerance*

  For the NLF-algorithm we use the same termination criteria as for the PNLF-algorithm. For both simple and restricted monotonicity the iteration is terminated if (3.98) holds, i.e., if

  $$\|\Delta x_l\|_2 \leqslant \text{XTOL} \tag{6.5}$$

  is true. Then the final iterate is determined via $x_{*,l+1} := x_l + \Delta x_l$. Recall that there is a second termination criterion: In case of simple monotonicity it is given by (3.100), i.e., if descent holds for a predictor of magnitude one and also the corrector is one then we check for

  $$\|\Delta x_l\|_2 \leqslant \sqrt{10 \cdot \text{XTOL}} \qquad \text{and} \qquad \|\overline{\Delta x}_{l+1}\|_2 \leqslant \text{XTOL}. \tag{6.6}$$

  In case of restricted monotonicity the above inequalities are considered if the predictor is one and the predictor passes the restricted monotonicity test (6.2).

  If (6.6) holds we return $x_{*,l+1} := x_l + \overline{\Delta x}_{l+1}$ as final iterate.

  For the APNLF-algorithm we apply for both monotonicity concepts analogous termination criteria where according to (4.162) and (4.163) the checks (6.5) and (6.6) are substituted by

  $$\|\delta x_l\|_2 \leqslant \text{XTOL}$$

  and

  $$\|A_{l,\bar{k}_l}^{-1} F(x_{l+1})\|_2 \leqslant \text{XTOL}$$

  where $\bar{k}_l$ is the final purifying index at step l. The final iterate is $x_{*,l+1} := x_l + \delta x_l$ or $x_{*,l+1} := x_l - A_{l,\bar{k}_l}^{-1} F(x_{l+1})$, respectively.

  In all cases the value

  $$\text{XTOL} = \sqrt{n} \cdot 10^{-10}$$

  is taken.

  If adaptive scaling is considered we obtain for the chosen value of XTOL and thrsh an equivalent termination criterion to the one used in [26] for tests of the NLF-algorithm `NLEQ1`. Also, the values for λ_0 and λ_{min} (simple monotonicity) are equal to the default values used in [26].

There are a couple of more constants to be set. E.g., for the decision when to increase step sizes in case of simple monotonicity or for the decision which kind of purifying update during a purifying process is performed. For our chosen values of these additional constants we refer the interested reader to Appendix III.

Test environment
All algorithms are implemented in MATLAB, [19]. Computations are carried out on a Dell Precision 380 - Intel Pentium processor Extreme Edition 955 (3.4 GHz, 2x2MB Cache, 1066MHz FSB) with 8 GB of memory and for MATLAB 7.9.0.529 (2009b). Run times are measured by means of MATLAB's `cputime`-command following the scheme

$$\texttt{t} = \texttt{cputime}; \qquad \texttt{invoke solver}; \qquad \texttt{run\_time} = \texttt{cputime} - \texttt{t};$$

To obtain the Jacobian $J := F'(x)$, $x \in \mathcal{D}$, and adjoint or direct tangent evaluations, i.e.,

$$w^T \cdot J \quad \text{or} \quad J \cdot d,$$

respectively, we either use an implementation of the analytical derivative in MATLAB or compute these quantities by means of Automatic Differentiation techniques. For the latter approach we employ version 2.0 of the tool ADOL-C, [13, 1]. This tool provides Automatic Differentiation functionalities in the context of the languages C/C++. MATLAB has the capacity to run functions written in C/C++. This functionality is provided by the MATLAB MEX interface. We provide a computational description of F in C++ including the necessary modifications required to be compatible to MATLAB's MEX interface. The C++ file is compiled in the MATLAB-environment by means of the `mex`-command. An execution of this command requires a C/C++ compiler to be available. We use *gcc* 4.3.1.

If we consider the run time of an algorithm derivative information is always provided by means of ADOL-C.

For the rank-1 update algorithm of a given LU-decomposition from [17] we provide an implementation in MATLAB and in C++. The latter implementation is compatible to MATLAB's MEX interface and can be invoked like a built-in function. We use the C++-implementation if run times of the APNLF-algorithm are of interest.

Additionally, we wrote in C++ linear system solvers to solve for y and z in (6.1) by means of a LU-decomposition of M. The solutions are computed by forward substitution followed by back substitution or vice versa, respectively. Such a functionality is also provided by MATLAB's built-in functions `mldivide` and `mrdivide`, respectively. We do not use these built-in function since it turned out that `mrdivide` is significantly slower than `mldivide`. Such a drawback does not exist for our C++ functions. We employ the C++ functions every time run times of algorithms are considered.

6.1 Test Set

The test problems which we will consider are stated in Table 6.1. The set is a mixture of quite different problems. There are artificial test problems (1-3 and 7) as well as problems which are are related to real life applications (4-6). Problem 8 is some kind of hybrid. It is defined via a discretization of an integral equation, however, the defining quantities are of a rather academical choice.

We refer to the problems 1-6 as *basic test set*. The main purpose of these problems is to test the robustness of the PNLF- and APNLF-algorithm. We will invoke the algorithms with the initial guesses from the literature and check for convergence. Additionally, we will compare the computational efficiency of the NLF- and the PNLF-algorithm. Note that problem 3 has its first appearance in this work. We will provide an initial guess via (6.7).

The problems 7 and 8 are of variable dimension. We will consider them for rather large values of n compared to the other problems, i.e., for $n \in \mathcal{O}(10^3)$. The main purpose is to compare the quasi-Newton approach of the APNLF-algorithm to the Jacobian based approaches of the NLF- and PNLF-algorithm in terms of run time of the algorithms.

In the context of the projected level functions three predictors are available (simple, Deuflhard-like and nonlinearity bound predictor). We will perform an additional test for problem 2 to compare these predictors. As it will turn out the nonlinearity bound predictor performs best in this test. Therefore, the test problems stated in Table 6.1 are only considered for this choice of predictor if the PNLF- or APNLF-algorithm is chosen.

No	Name	Abbreviation	Dimension n	Reference
1	Example from Subsection 3.2.7	Quadpoly	2	[5, 6]
2	Exponential/sine function	Expsin	2	[26, 11]
3	5-spheres function	5spheres	3	†
4	Semiconductor boundary condition	Semicon	6	[21, 26]
5	Distillation column – Hydrocarbon 6	Hydro6	29	[23]
6	Distillation column – Methanol 8	Metha8	31	[23]
7	Trigonometric function	Trigo	variable	[25]
8	Discrete integral equation function	Discint	variable	[24, 25]

†: This problem is introduced in this work.

Table 6.1: Test set

Description of problems

In the following we will give a short description of the test problems from Table 6.1. Note that by $x_{(i)}$ and $F_{(i)}$ we refer to the i-th component of the vector x or F, respectively.

- *Quadpoly*
  Recall from Subsection 3.2.7 that this is a parameter dependent problem given via

$$F(x) := \begin{pmatrix} x_{(1)} \\ a \cdot x_{(2)} + \frac{1}{4}(x_{(1)} - 50)^2 \end{pmatrix} = 0.$$

  The unique solution is

$$x_* = \begin{pmatrix} 0 \\ -625 \cdot a^{-1} \end{pmatrix}.$$

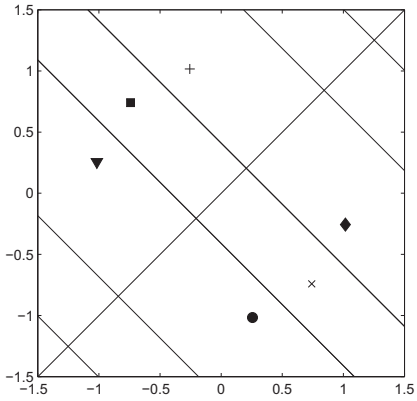


Figure 6.1: *Expsin* – The six roots, separated by lines of singular Jacobians (—)

We consider this problem for the initial guess

$$x_0 = \begin{pmatrix} 50 \\ 1 \end{pmatrix}$$

and for two choices of a given via $a = 50$ and $a = 1$. If the first choice of a is considered we refer to the problem as Quadpoly_{50}. Accordingly, Quadpoly_1 relates to the second choice.

- *Expsin*
  The system to be solved reads as follows

$$F(x) := \begin{pmatrix} \exp\bigl(x_{(1)}^2 + x_{(2)}^2\bigr) - 3 \\ x_{(1)} + x_{(2)} - \sin\bigl(3(x_{(1)} + x_{(2)})\bigr) \end{pmatrix} = 0.$$

  Critical interfaces with singular Jacobians are given for the lines

$$x_{(2)} = x_{(1)} \quad \text{and} \quad x_{(2)} = -x_{(1)} \pm \tfrac{1}{3} \arccos\bigl(\tfrac{1}{3}\bigr) \pm \tfrac{2}{3}\pi \cdot j, \quad j = 0, 1, 2, \ldots.$$

  By means of these lines parallel sectors are defined such that for each x inside a sector the Jacobian is nonsingular.

  All six solutions are in the domain

$$-1.5 \leqslant x_{(1)},\, x_{(2)} \leqslant 1.5$$

  and inside a sector as depicted in Figure 6.1. For the basic test set we use the initial guess

$$x_0 = \begin{pmatrix} 0.81 \\ 0.82 \end{pmatrix}$$

which is also chosen in [26]. Note that this starting value is in close vicinity to two of the line interfaces where the Jacobian is singular.

- *5spheres*
  With

$$\begin{aligned} x_{(1)}^2 + x_{(2)}^2 + x_{(3)}^2 - 4 &=: K_1(x) \\ (x_{(1)} - 2)^2 + x_{(2)}^2 + x_{(3)}^2 - 1 &=: K_{2a}(x) \\ (x_{(1)} + 2)^2 + x_{(2)}^2 + x_{(3)}^2 - 1 &=: K_{2b}(x) \\ x_{(1)}^2 + x_{(2)}^2 + (x_{(3)} - 5)^2 - 25 &=: K_{3a}(x) \\ x_{(1)}^2 + x_{(2)}^2 + (x_{(3)} + 5)^2 - 25 &=: K_{3b}(x) \end{aligned}$$

  we define

$$F(x) := \begin{pmatrix} K_1(x) \\ K_{2a}(x) \cdot K_{2b}(x) \\ K_{3a}(x) \cdot K_{3b}(x) \end{pmatrix} = 0.$$

  This problem describes the intersection of five spheres in $\mathbb{R}^3$, see Figure 6.2(a). There are eight roots. Due to the symmetry of the problem these roots are of the form

$$x_{\cdot,*} = \begin{pmatrix} \pm a \\ \pm b \\ \pm c \end{pmatrix}, \quad a, b, c \in \mathbb{R}_+.$$

  In Figure 6.2(b) two of the roots are depicted. The eight solutions are separated by interfaces of singular Jacobians which again due to symmetry reasons turn out to be the planes

$$E_i := \{x \in \mathbb{R}^3 \mid x^T e^{(i)} = 0\}, \quad i = 1, 2, 3,$$

  where $e^{(i)}$ is the i-th unit vector in $\mathbb{R}^3$. The initial guess is chosen close to two of these separating planes:

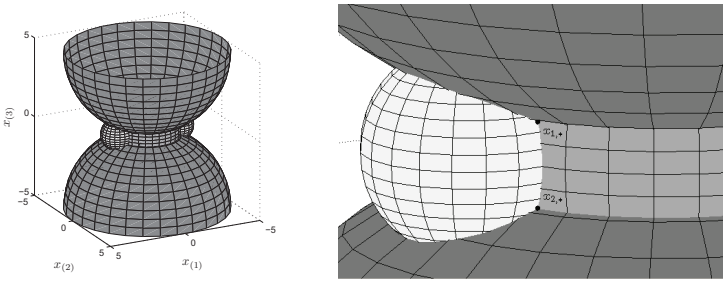
$$x_0 = \begin{pmatrix} 1 \\ 10^{-2} \\ 10^{-4} \end{pmatrix}. \tag{6.7}$$

- *Semicon*
  This problem describes the boundary conditions for a 2D semiconductor device simulation:

$$F(x) := \begin{pmatrix} \exp\big(\alpha(x_{(3)} - x_{(1)})\big) - \exp\big(\alpha(x_{(1)} - x_{(2)})\big) - D/n_i \\ x_{(2)} \\ x_{(3)} \\ \exp\big(\alpha(x_{(6)} - x_{(4)})\big) - \exp\big(\alpha(x_{(4)} - x_{(5)})\big) + D/n_i \\ x_{(5)} - V \\ x_{(6)} - V \end{pmatrix} = 0$$

  where

$$\begin{aligned} \alpha &:= 38.683, & n_i &:= 1.22 \cdot 10^{10}, \\ V &:= 100, & D &:= 10^{17}. \end{aligned}$$



(a) Intersecting spheres

(b) Zoom to identify two of the eight roots

Figure 6.2: Visualization of problem *5spheres*

In conjunction with the initial guess

$$x_0 = \begin{pmatrix} 1 \\ \vdots \\ 1 \end{pmatrix} \in \mathbb{R}^6$$

this is an extremely nonlinear and sensitive problem according to [26].

- *Hydro6* and *Metha8*
  These problems describe the steady state conditions of a k-stage distillation column consisting of a reboiler, $k-2$ plates and a condenser. The problems differ in the type of involved chemical substances and in the number of stages, $k = 6$ for *Hydro6* and $k = 8$ for *Hydro8*. The various equations which define the steady state, i.e., $F(x) = 0$ and the initial guesses x_0 can be found in [23].

- *Trigo*
  Let
  $$F(x) := \big(F_{(i)}(x)\big)_{i=1,\dots,n} = 0$$
  where
  $$F_{(i)}(x) := n - \sum_{j=1}^{n} \cos(x_{(j)}) + i \cdot \big(1 - \cos(x_{(i)})\big) - \sin(x_{(i)}).$$
  A solution to this problem is given by $x_* = 0 \in \mathbb{R}^n$. The Jacobian of this artificial problem is of special structure, i.e.,
  $$F'(x) = \begin{pmatrix} 1 \\ \vdots \\ 1 \end{pmatrix} \cdot s(x)^T + D(x) \tag{6.8a}$$
  where
  $$s(x) := \big(\sin(x_{(i)})\big)_{i=1,\dots,n} \quad \text{and} \quad D(x) := \operatorname{diag}\Big[\big(i \cdot \sin(x_{(i)}) - \cos(x_{(i)})\big)_{i=1,\dots,n}\Big]. \tag{6.8b}$$

We are interested if this structure can be exploited by the APNLF-algorithm. For the test runs we consider the problem for $n \in \{2 \cdot 10^3, 4 \cdot 10^3\}$ and

$$x_0 = \tfrac{3}{5} \cdot \hat{x}_0 \qquad \text{where} \qquad \hat{x}_0 = \frac{1}{n} \cdot \begin{pmatrix} 1 \\ \vdots \\ 1 \end{pmatrix} \in \mathbb{R}^n$$

is the initial guess proposed in [25]. For $\hat{x}_0$ none of the considered methods converges for the chosen dimensions. Note that in [26] convergence for $\hat{x}_0$ and $n = 10$ is obtained if a special rank reduction strategy for the Jacobians is applied. We do not consider such strategies in this work and therefore change the initial guess to guarantee convergence for the considered methods.

- *Discint*

  We consider the problem

  $$F(x) := \big(F_{(i)}(x)\big)_{i=1,\dots,n} = 0$$

  where

  $$F_{(i)}(x) := x_{(i)} + \frac{h}{2}\left[(1-t_i)\sum_{j=1}^{i} t_j(x_{(j)} + t_j + 1)^3 + t_i \sum_{j=i+1}^{n} (1-t_j)(x_{(j)} + t_j + 1)^3\right]$$

  with

  $$h := 1/(n+1), \quad t_i := i \cdot h, \quad x_{(0)} := 0 =: x_{(n+1)}.$$

  This problem arises from a discretization of the nonlinear integral equation

  $$u(t) + \int_0^1 H(s,t)\big(u(s) + s + 1\big)^3 ds = 0$$

  where

  $$H(s,t) := \begin{cases} s(1-t), & s \leqslant t \\ t(1-s), & s \geqslant t. \end{cases}$$

  See [24] for details. The unknowns $x_{(i)}$ represent the function u evaluated at t_i, i.e., $x_{(i)} = u(t_i)$. The function F is differentiable and the Jacobian is dense. According to [24] this problem has a unique solution x_* with

  $$-\frac{1}{2} \leqslant x_{*,(i)} \leqslant 0, \quad i = 1, \dots, n. \tag{6.9}$$

  The default initial guess from [24] which is also proposed in [25] is given via

  $$\hat{x}_0 = (\hat{x}_{0,(i)})_{i=1,\dots,n} \qquad \text{with} \qquad \hat{x}_{0,(i)} = t_i \cdot (t_i - 1).$$

  We consider this problem for $n \in \{2 \cdot 10^3, 4 \cdot 10^3\}$ combined with

  $$x_{1,0} = 10^2 \cdot \hat{x}_0, \qquad x_{2,0} = 5 \cdot 10^2 \cdot \hat{x}_0.$$

6.1.1 *Expsin* grid test

We define with $\triangle = 0.06$ a grid of 'initial guesses'

$$x_0^{(i,j)} := \begin{pmatrix} -1.5 + i \cdot \triangle \\ -1.5 + j \cdot \triangle \end{pmatrix}, \quad i, j = 0, \ldots, 50,$$

and apply the NLF-, PNLF- and APNLF-algorithm to the problem *Expsin* for these values. If for an $x_0^{(i,j)}$ the Euclidean distance to any of the critical interfaces is smaller than 10^{-4} we skip this initial guess. The grid test was introduced in [26] to test out the reliability of the NLF related algorithm `NLEQ1`. We use this test to investigate the reliability of the PNLF- and APNLF-algorithm in general and for a comparison of the three available predictors. As reference method we use our NLF-algorithm. Additionally, we compare the reference method to the PNLF-algorithm in terms of required function evaluations (fevals) and Jacobian evaluations (Jevals) to run the test. This is done for all three predictors.

6.1.1.1 Reliability

We consider an algorithm to be reliable for a given initial guess if the iterates which are produced by the algorithm converge to a root x_* such that the Newton path $\overline{x}$ at the initial guess is 'in relationship with this root', i.e., $\overline{x}(0) = x_0$, $\overline{x}(\lambda)$ is well defined for $\lambda \in [0, 1]$ and $\overline{x}(1) = x_*$.

Regarding the *Expsin* problem this means that convergence shall only occur if the initial guess and the root the iteration converges to are in the same sector. If such a convergence occurs we say that the iteration converges to the 'correct' root. Otherwise, if an iteration converges to a root which is not located in the same sector as the initial guess we call this a 'misleading' iteration.

We consider the grid test for no scaling in the domain space of F and for adaptive scaling according to (6.3). The number of 'misleading' iterations for the considered methods and predictors in case of simple monotonicity is stated in Table 6.2. The abbreviation *nlb* refers to the projected nonlinearity bound predictor, *Dflhd-like* to the Deuflhard-like predictor and *simple* to the simple predictor. For the tuple $l|r$ the left value l refers to the default values of λ_0 and λ_{min}, i.e., $\lambda_0 = 10^{-2}$ and $\lambda_{min} = 10^{-4}$, whereas r refers to the more restrictive choices $\lambda_0 = 10^{-4}$ and $\lambda_{min} = 10^{-6}$. The overall performance for all methods is rather satisfying. The PNLF introduces only a small number of additional 'misleading' iterations and only in case of the Deuflhard-like and the simple predictor. For the nonlinearity bound predictor the same performance as for the reference method is obtained. The four 'misleading' iterations which occur for all methods in case of the default values of λ_0 and λ_{min} are related to the initial guesses marked by a frame in Figure 6.3. These are also the same initial guesses the algorithm `NLEQ1` in [26] fails for. These failures occur because simple monotonicity is fulfilled for the (too large) initial step size λ_0. It is this first step which lets the iteration cross an interface, see Figure 6.4(a). A restriction of λ_0 and λ_{min} to the values $\lambda_0 = 10^{-4}$ and $\lambda_{min} = 10^{-6}$ eliminates the 'misleading' iterations in almost all cases. Only for the PNLF-algorithm combined with the simple predictor and in case of no scaling four 'misleading' iterations occur. These vanish too if we further restrict λ_0 and λ_{min} to be $\lambda_0 = 10^{-7}$ and $\lambda_{min} = 10^{-10}$.

	NLF	PNLF			APNLF		
		nlb	Dflhd-like	simple	nlb	Dflhd-like	simple
adaptive scaling	4|0	4|0	4|0	8|0	4|0	4|0	4|0
no scaling	4|0	4|0	8|0	4|4	4|0	4|0	4|0

$l|r$: l refers to default values of λ_0 and λ_{min}, i.e., $\lambda_0 = 10^{-2}$ and $\lambda_{min} = 10^{-4}$
r to the restricted choices $\lambda_0 = 10^{-4}$ and $\lambda_{min} = 10^{-6}$

Table 6.2: Grid test – # of 'misleading' iterations for the stated methods and predictors in the context of simple monotonicity

Regarding restricted monotonicity no method produces 'misleading' iterations for the default values of λ_0 and λ_{min} except the APNLF-algorithm combined with the simple predictor in case of adapted scaling. The four arising 'misleading' iterations are gone if we choose $\lambda_0 = 10^{-4}$ and $\lambda_{min} = 10^{-8}$. This result nicely shows the potential of the restricted monotonicity step size strategy. This can also be seen from Figure 6.4(b): The restricted monotonicity step size strategy sufficiently reduces the default value of λ_0 to let the iteration stay in the sector. Finally, convergence to the 'correct' root is obtained. However, the safety provided by the restricted monotonicity concept has its price. As expected the number of fevals and Jevals rises compared to the simple monotonicity concept. For the NLF- and PNLF-algorithm and both types of scaling simple monotonicity requires only approximately 92% of fevals and 96% of Jevals compared to restricted monotonicity to run the test.

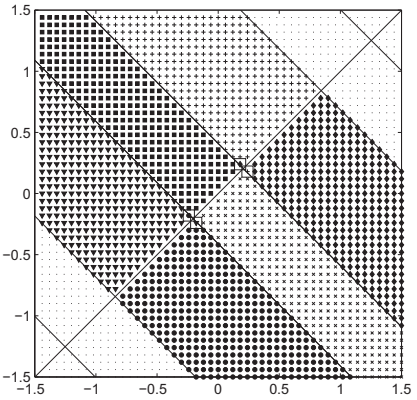
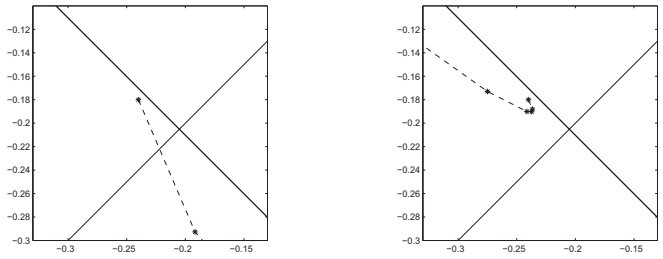


Figure 6.3: Result of the grid test for the the PNLF- and APNLF-algorithm combined with the nlb-predictor and the reference method in case of simple monotonicity. Initial values of 'misleading' iterations are marked by a frame.



(a) Simple monotonicity (b) Restricted monotonicity

Figure 6.4: *Expsin* – Behavior near critical interface for default value of λ_0

6.1.1.2 Efficiency

The investigations from the previous section concerning reliability indicate that the nonlinearity bound predictor is a better choice over the Deuflhard-like and simple predictor. To have a second criterion available we investigate the number of fevals and Jevals to run the grid test for the reference method and the PNLF-algorithm combined with all three predictors and for default λ_0 and λ_{min}.

We compare the following quantities:

rfa Considering a run of the grid test for the reference method let f_1 be the total number of fevals for all sequences which converge to the 'correct' root. Accordingly, let f_2 be the total number of fevals for all iterations in the context of the PNLF starting at the same initial iterates. The quantity *rfa* is the ratio f_2/f_1 apart from one exception:

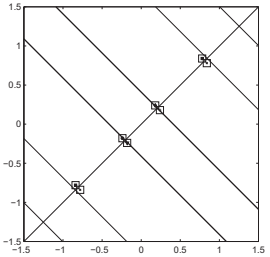
In case of simple monotonicity and no scaling we slightly change the definition of *rfa*. The reason for this is that four additional 'misleading' iterations occur in case that the PNLF-algorithm is combined with the Deuflhard-like predictor, see Table 6.2 and Figure 6.5(a). Considering the associated initial guesses the reference method converges to the 'correct' roots. It is not reasonable to take these four iterations into account for comparison. Hence, we discard the associated initial guesses and run the grid test for the remaining ones for both methods. We adapt the definition of f_1 and f_2 accordingly and therefore the definition of *rfa*.

Table 6.2 shows that there are also four additional 'misleading' iterations for the PNLF-algorithm combined with the simple predictor in case of simple monotonicity and adaptive scaling. These iterations need no extra care since the associated initial guesses are in sectors where no root is located, see Figure 6.5(b). The reference method does not converge for these guesses. Hence, they are anyway not considered for the ratio *rfa*.

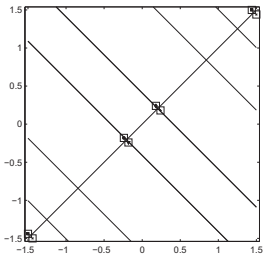
rJa As above, though the Jevals are counted instead of the fevals.

rfnl This ratio is like *rfa*, except that only these iterations are taken into consideration where the convergence for the reference method is nonlocal. Nonlocal means that damping occurs.

rJnl Like *rfnl*, but considering Jevals instead of fevals.



(a) Deuflhard-like predictor (no scaling)



(b) Simple predictor (adaptive scaling)

Figure 6.5: Grid test – Additional 'misleading' iterations for the PNLF-algorithm in the context of simple monotonicity

The results are stated in Table 6.3 and 6.4. For any combination of step size strategy and scaling type the PNLF-algorithm in conjunction with the nonlinearity bound predictor shows the best performance. Though there is no significant boost in terms of efficiency compared to the reference method, there are at least minor improvements.

Summarizing, for the grid test the nonlinearity bound predictor appears to be superior to the Deuflhard-like and simple predictor in terms of reliability and efficiency. As a consequence, we will consider the PNLF- and APNLF-algorithm for the problems in the basic test set and for the problems of variable dimension only in combination with the nonlinearity bound predictor.

simple monotonicity adaptive scaling $\lambda_0 = 10^{-2}$ $\lambda_{min} = 10^{-4}$	predictor	rfa	rJa	rfnl	rJnl
	nlb	98.65%	98.51%	98.39%	98.25%
	Dflhd-like	99.10%	99.07%	98.93%	98.90%
	simple	104.97%	100.36%	105.92%	100.42%

restricted monotonicity adaptive scaling $\lambda_0 = 10^{-2}$ $\lambda_{min} = 10^{-6}$	predictor	rfa	rJa	rfnl	rJnl
	nlb	98.10%	98.57%	97.88%	98.30%
	Dflhd-like	98.88%	98.57%	98.66%	98.30%
	simple	103.49%	100.97%	104.32%	101.15%

Table 6.3: Grid test – Ratios, adaptive scaling – see page 166 for an explanation of the ratios

simple monotonicity no scaling $\lambda_0 = 10^{-2}$ $\lambda_{min} = 10^{-4}$	predictor	rfa	rJa	rfnl	rJnl
	nlb	98.51%	98.61%	98.12%	98.28%
	Dflhd-like	99.25%	99.12%	99.06%	98.91%
	simple	102.44%	100.85%	103.01%	101.01%

restricted monotonicity no scaling $\lambda_0 = 10^{-2}$ $\lambda_{min} = 10^{-6}$	predictor	rfa	rJa	rfnl	rJnl
	nlb	98.04%	98.61%	97.93%	98.19%
	Dflhd-like	99.64%	98.88%	99.60%	98.58%
	simple	101.17%	101.27%	101.99%	101.62%

Table 6.4: Grid test – Ratios, no scaling – see page 166 for an explanation of the ratios

6.1.2 Basic test set

The basic test set consists of the problems 1-6 from Table 6.1. We invoke the NLF-, PNLF- and APNLF-algorithm with the initial guesses x_0 we stated in the description of the problems above. We run the algorithms for no scaling in the domain of F (nosc) and for adaptive scaling (adapt) according to (6.3). To refer to a problem for a particular type of scaling we use the notation *problem(sctype)*, e.g., *Expsin(adapt)*. Regarding step size strategies we only consider simple monotonicity. Compared to restricted monotonicity this is a less safe choice since it is more likely that 'misleading' iterations occur. However, in case of no failure simple monotonicity is also the more efficient concept since usually less fevals and Jevals are necessary in the course of the algorithm. We are interested in the performance and robustness of the PNLF- and APNLF-algorithm in this context.

For all methods we count the number of steps until convergence according to the termination criteria is assumed. Additionally, for all methods the number of function evaluations (fevals) is considered. For the NLF- and PNLF-algorithm also the number of Jacobian evaluations (Jevals) is counted. In case of the APNLF-algorithm we count the number of descent updates (descUpd), purifying updates (purUpd) and also the number of adjoint and direct tangent evaluations ($w^T \cdot J$ and $J \cdot u$).

The results of the NLF-, PNLF- and APNLF-algorithm for the basic test set regarding these quantities are stated in Table 6.5 and 6.6.

In case of the APNLF-algorithm we are also interested in the performance of our angle estimator $\angle_{est}(\delta x, \Delta x)$. Additionally, we check whether the true angle $\angle(\delta x, \Delta x)$ is greater than the angle tolerance ψ in case of a descent update where no calculation of $\angle_{est}(\delta x, \Delta x)$ is performed. Such a situation may arise, see the discussion in Subsection 4.4.2 on our strategy when to consider a descent update or a purifying process. The angle related results are stated in Table 6.7.

Example		# fevals		# Jevals		# steps		
Dim n	Abbrev.	NLF	PNLF	NLF	PNLF	NLF	PNLF	sctype
2	Quadpoly_{50}	4	4	3	3	2	2	nosc
		11	11	7	6	6	6	adapt
2	Quadpoly_1	13	5	8	3	7	2	nosc
		23	22	12	13	12	12	adapt
2	Expsin	12	13	10	11	10	11	nosc
		13	13	11	11	11	11	adapt
3	5spheres	13	10	11	8	11	8	nosc
		13	10	11	8	11	8	adapt
6	Semicon*	13	12	7	7	7	7	nosc
		13	12	7	7	7	7	adapt
29	Hydro6	8	8	6	6	6	6	nosc
		7	7	5	6	5	5	adapt
31	Metha8	6	6	5	5	4	4	nosc
		6	6	4	4	4	4	adapt

*: $\lambda_0 = 10^{-4}$ and $\lambda_{min} = 10^{-8}$ instead of the default values are used.
Otherwise no convergence occurs.

Table 6.5: Results of the NLF- and PNLF-algorithm for the basic test set

Example		APNLF						
Dim n	Abbrev.	# fevals	# descUpd	# purUpd	# $w^T \cdot J$	# $J \cdot u$	# steps	sctype
2	Quadpoly_{50}	4	1	0	1	0	2	nosc
		10	1	3	7	6	5	adapt
2	Quadpoly_1	5	0	1	2	3	2	nosc
		24	4	9	22	19	14	adapt
2	Expsin	14	9	3	14	8	12	nosc
		14	7	5	16	12	12	adapt
3	5spheres	14	7	5	15	10	11	nosc
		13	6	8	18	13	11	adapt
6	Semicon*	12	4	3	9	6	7	nosc
		12	5	2	8	4	7	adapt
29	Hydro6	8	0	31	36	36	6	nosc
		8	0	26	31	31	6	adapt
31	Metha8	7	0	19	23	23	5	nosc
		8	0	8	13	13	6	adapt
For each example one Jeval occurs since $A_0 = F'(x_0)$ is chosen								

*: $\lambda_0 = 10^{-4}$ and $\lambda_{min} = 10^{-8}$ instead of the default values are used.
Otherwise no convergence occurs.

Table 6.6: Results of the APNLF-algorithm for the basic test set

Example	APNLF		
	# Est. Fail.	# $\angle > \psi$ and noest	sctype
Quadpoly_{50}	0	0	nosc
	0	0	adapt
Quadpoly_1	0	0	nosc
	0	0	adapt
Expsin	0	1	nosc
	0	0	adapt
5spheres	2	0	nosc
	1	0	adapt
Semicon	0	0	nosc
	0	0	adapt
Hydro6	0	0	nosc
	0	0	adapt
Metha8	0	0	nosc
	0	0	adapt

Est. Fail.: Estimation Failure, i.e.,
$\angle_{est}(\delta x, \Delta x) \leqslant \psi$ but $\angle(\delta x, \Delta x) > \psi$
$\angle > \psi$ and noest : $\angle(\delta x, \Delta x) > \psi$ while a descUpd is
taken where no calculation of
$\angle_{est}(\delta x, \Delta x)$ is considered

Table 6.7: $\angle(\delta x, \Delta x)$ and $\angle_{est}(\delta x, \Delta x)$ related results for the basic test set

6.1.2.1 Remarks on the results

The PNLF- and APNLF- algorithm show a very well performance in terms of robustness for all test problems: Just as the reference method convergence to the unique solution or in case of several solutions to the 'correct' root is achieved. Though we had to reduce the values of λ_0 and λ_{min} to $\lambda_0 = 10^{-4}$ and $\lambda_{min} = 10^{-8}$ for *all* methods, this is even the case for the extremely nonlinear problem *Semicon*.

The problems *Expsin(adapt)*, *Hydro6(adapt)*, *Metha8(adapt)* and *Semicon(adapt)* are also considered in [26] to test the code `NLEQ1`. For the first four problems our reference method shows nearly the same behavior as `NLEQ1`. The only difference is that our algorithm requires one less Jeval for *Hydro6(adapt)* and *Metha8(adapt)*. According to [26] convergence for the problem *Semicon(adapt)* is obtained if `NLEQ1` is restarted with $\lambda_{min} = 10^{-8}$, however, no fevals and Jevals are stated in [26].

Note that a discrepancy in Jacobian evaluations and number of taken steps as it may appear for one of the problems in Table 6.5 arises from a termination of the respective algorithm because (6.5), i.e., $\|\Delta x_l\|_2 \leqslant \text{XTOL}$ is true. Since this check is done at the very beginning of each step we do not consider the final step for counting if the aforementioned check holds and therefore the final iterate is simply computed via $x_l + \Delta x_l$ without invoking a step size control.

Considering the efficiency of the reference method and the PNLF-method Table 6.5 shows that the PNLF-algorithm performs marginally better in terms of fevals and Jevals, even if we neglect the problem $Quadpoly_1(nosc)$. It shall be noted that in general the chosen step sizes for the PNLF-algorithm are larger than the ones produced by the reference method as it is expected due to the projectional aspect of the PNLF. However, quite often the difference is very small such that no gain in terms of less iteration steps is obtained. This gives the impression that it is not uncommon that the local nonlinearity orthogonal to the Newton correction, i.e., $\chi_\perp(\lambda)$ as defined in Proposition 3.4 is of no substantial magnitude.

By means of the test problem *5spheres* an example is given where the larger step sizes of the PNLF-algorithm lead to a gain in efficiency. Both the reference method and the PNLF-algorithm rely on damping, however, the step sizes for the PNLF-algorithm quicker reach the 1-level which in turn leads to faster convergence, see Figure 6.6.

For the problem $Quadpoly_{50}(nosc)$ and $Quadpoly_{50}(nosc)$ the reference method and the PNLF-algorithm perform as expected from the discussion in Subsection 3.2.7. Adaptive scaling has an interesting effect on this problem. The rise in fevals and Jevals for both methods is not related to algorithmic aspects, it is a consequence of the underlying natural level function concept:

Recall from Subsection 3.2.7 that for the *unscaled* problem *Quadpoly* we have

$$\Delta x_0 = -x_0 = (-50, 1)^T$$
$$\chi_0(\lambda) = J(x_0)^{-1}\big(F(x_0 + \lambda\Delta x_0) - F(x_0) - \lambda J(x_0)\Delta x_0\big) = \lambda^2 \cdot (0, 625 \cdot a^{-1})^T$$

which implies that

$$\mu_0(\lambda) = -\frac{\Delta x_0^T \chi_0(\lambda)}{\|\Delta x_0\|_2^2} \approx \frac{1}{4} a^{-1}\lambda^2, \qquad \beta_0(\lambda) = \frac{\|\chi_{0,\perp}(\lambda)\|_2^2}{\|\Delta x_0\|_2^2} \approx 156 \cdot a^{-2}\lambda^4$$

and hence

$$\frac{T\big(x_0+\lambda\Delta x_0|P_{N_0}J(x_0)^{-1}\big)}{T\big(x_0|P_{N_0}J(x_0)^{-1}\big)} \approx \big(1-\lambda+\tfrac{1}{4}a^{-1}\lambda^2\big)^2$$
$$\frac{T\big(x_0+\lambda\Delta x_0|J(x_0)^{-1}\big)}{T\big(x_0|J(x_0)^{-1}\big)} \approx \big(1-\lambda+\tfrac{1}{4}a^{-1}\lambda^2\big)^2 + 156\cdot a^{-2}\lambda^4.$$

Due to the scaling strategy (6.3) the first scaling matrix is $D_0 = \mathrm{diag}(50,1)$. This leads to the scaled correction $\Delta x_0^{sc} = D_0^{-1}\Delta x_0 = -(1,1)^T$. However, $\chi_0(\lambda)$ does not change, i.e., $\chi_0^{sc}(\lambda) = \chi_0(\lambda)$. The scaled counterparts of $\mu_0(\lambda)$ and $\beta_0(\lambda)$ are

$$\mu_0^{sc} \approx 312\cdot a^{-1}\lambda^2 \qquad \text{and} \qquad \beta_0^{sc}(\lambda) \approx 9.8\cdot 10^4 a^{-2}\lambda^4.$$

Hence, a drastic reduction in step sizes is to be expected. That is just what the algorithms do. Note that in this scaled scenario the ordinary Newton method *still* converges within two steps. So by simple rescaling a former prime example for the efficiency of the natural level function concept turns into a rather disadvantageous example. However, this does not mean that in general scaling is bad. There are only minor performance changes for the other problems and in case of *Hydro6* and *Metha8* considering scaling turns out to be slightly more efficient. Furthermore, one should not forget that scaling invariance is obtained by the adaptive scaling strategy, see Subsection 3.4.3. Additionally, the algorithms terminate if an estimate for the componentwise *relative* error is small enough. This is especially advantageous if there is a considerable difference in the order of magnitude of the components of a solution. E.g., the solution to *Hydro6* contains components of order $\mathcal{O}(10^{-3})$ as well as components of order $\mathcal{O}(10^2)$.

That the APNLF is an approximation to the PNLF is confirmed by the number of the steps both associated algorithms require to converge. Throughout, if the APNLF-algorithm needs more steps the increase is only of small magnitude. This shows that our strategy to monitor angles and apply purifying updates if necessary works. An illustrative example is depicted in Figure 6.7. There, the problem *Expsin(adapt)* is considered for the default APNLF-algorithm, i.e., angle checks and purifying are active and also for the APNLF-algorithm where solely the descent update is employed.

If purifying is considered almost in any case the duophilic update is used. Only for the problem $Quadpoly_1(nosc)$ the one employed purifying update is a gradientphilic update. Throughout, no Newton-philic update is applied. This also implies that no singular Jacobian approximation or an approximation we consider to be ill-conditioned arises. Furthermore, no post-purifying occurs.

Except for the problems $Quadpoly_1(nosc)$, *Hydro6* and *Metha8* the APNLF-algorithm finally switches to the descent update. For $Quadpoly_1(nosc)$ after two steps each with step size one the exact solution is obtained. Regarding the two problems *Hydro6* and *Metha8* no descent update at all is employed. The convergence history for *Hydro6(adapt)* and *Metha8(adapt)* is depicted in Figure 6.8. In case of *Hydro6(adapt)* superlinear convergence is clearly identifiable. This is in contrast to the problem *Metha8(adapt)* where convergence occurs but not (yet) in a superlinear manner. However, *Hydro6(adapt)* shows that solely applying duophilic updates may also lead to a superlinear convergent sequence of iterates. Note that for the problems *Hydro6(adapt)* and *Metha8(adapt)* it turns out to be advantageous to apply adaptive scaling if we consider the number of computed purifying updates. The number is considerably reduced especially in case of *Metha8*.

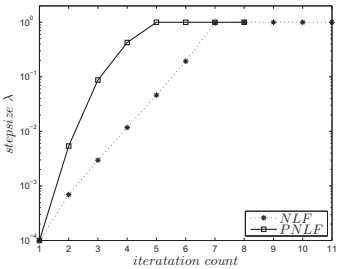
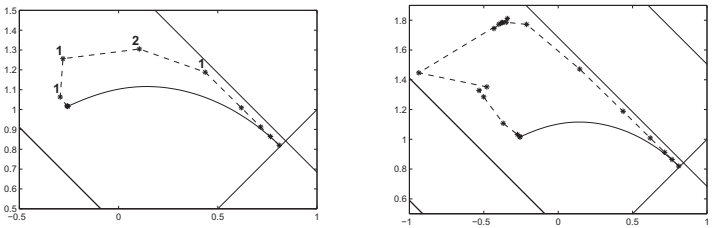


Figure 6.6: *5spheres* – Comparison of taken step sizes

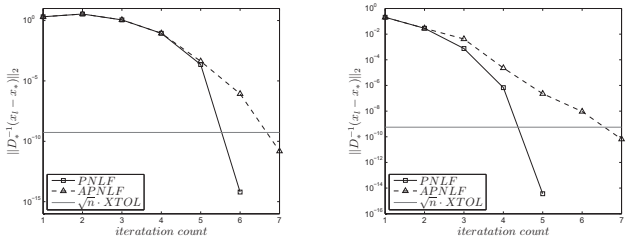


(a) Purifying: numbers represent # purUpd

(b) No purifying

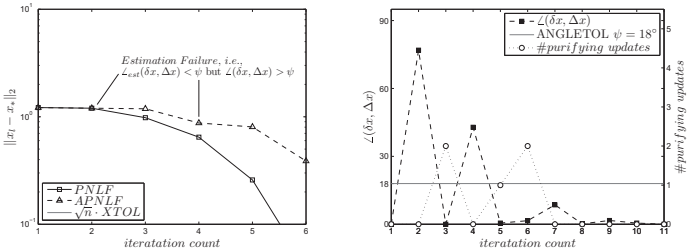
Figure 6.7: *Expsin* – Influence of purifying, iterates compared to Newton path for $x_0 = (0.81, 0.82)^T$

Table 6.7 shows that our angle estimator $\angle_{est}(\delta x, \Delta x)$ performs very well. For all problems except $Quadpoly_{50}(nosc)$ purifying updates are performed and hence $\angle_{est}(\delta x, \Delta x)$ is computed. There are only three overall estimation failures which occur only for the problem *5spheres*. Also, Table 6.7 shows that if $\angle_{est}(\delta x, \Delta x)$ is not considered due to a descent update only once the real angle $\angle(\delta x, \Delta x)$ is above the angle tolerance ψ. Regarding the estimation failures in the *5spheres* problem the APNLF-algorithm shows a very pleasant 'post-compensation' behavior. In Figure 6.9(b) and 6.10(b) it is shown that after an estimation failure a purifying process is initiated in the subsequent step. This way, the approximate correction δx is again aligned to the Newton correction Δx. The curves of convergence history in Figure 6.9(a) and 6.10(a) nicely reflect the benefit of this 'post-compensation' behavior. The curves are rather flat for a step where the estimator fails. On the other hand, for the subsequent step where purifying is invoked the curves drop considerably.



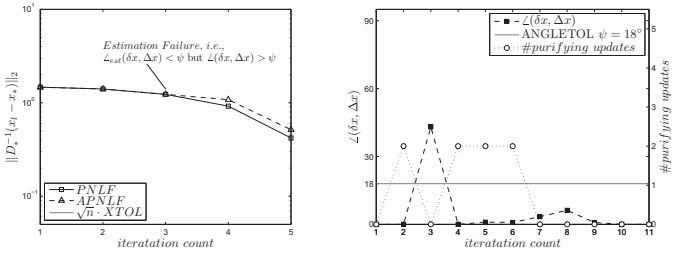
(a) *Hydro6(adapt)* (b) *Metha8(adapt)*

Figure 6.8: APNLF – Convergence history for example *Hydro6(adapt)* and *Metha8(adapt)*



(a) Convergency history – cut out (b) Angle/purifying relation

Figure 6.9: *5spheres*(nosc) – Purifying in the subsequent step compensates for failed estimate of $\angle(\delta x, \Delta x)$



(a) Convergency history – cut out (b) Angle/purifying relation

Figure 6.10: *5spheres*(adapt) – Purifying in the subsequent step compensates for failed estimate of $\angle(\delta x, \Delta x)$

6.1.3 Problems of variable dimension

In this subsection we will consider the problems *Trigo* and *Discint* for the dimensions $n = 2 \cdot 10^3$ and $n = 4 \cdot 10^3$ and in case of *Discint* combined with two initial guesses. A main purpose of testing these problems is to investigate the efficiency of the quasi-Newton based approach of the APNLF-algorithm. Therefore, we measure the run time of the APNLF-algorithm for the two considered problems and compare it to the run times of the NLF- and PNLF-algorithm. Derivative information will be provided by the AD-tool ADOL-C. We use our C++ implementation of the LU rank-1 update algorithm from [17]. Also, we employ our C++ codes which solve linear systems for a triangular matrix via forward and backward substitution.

Just like for the basic test set we additionally consider the performance of the angle estimator $\angle_{est}(\delta x, \Delta x)$ and also the value of the angle $\angle(\delta x, \Delta x)$ if no $\angle_{est}(\delta x, \Delta x)$ is computed. Furthermore, we count the number of performed descent and purifying updates (descUpd, purUpd).

No scaling is taken into account for the two test problems in this subsection. Regarding *Trigo* scaling is anyway not reasonable since the algorithms converge to the solution $x_* = 0 \in \mathbb{R}^n$.

The results of the run time tests are given in Table 6.8 and 6.10. Angle and descent/purifying update related data is provided by Table 6.9 and 6.11.

Trigo				
Dim n	NLF	PNLF	APNLF	data
$2 \cdot 10^3$	7	7	13	# steps
	11.34s	11.23s	5.7s	run time
$4 \cdot 10^3$	7	7	13	# steps
	71.43s	70.2s	29.37s	run time

$x_0 = \frac{3}{5} \cdot \hat{x}_0$ with $\hat{x}_0$ from [25]

Table 6.8: *Trigo* – Results of the NLF/PNLF/APNLF-algorithms

Trigo				
Dim n	# Est. Fail.	# $\angle > \psi$ and noest	# descUpd	# purUpd
$2 \cdot 10^3$	0	0	11	1
$4 \cdot 10^3$	0	0	11	1

Est. Fail.: Estimation Failure, i.e.,
$\angle_{est}(\delta x, \Delta x) \leqslant \psi$ but $\angle(\delta x, \Delta x) > \psi$

$\angle > \psi$ and noest : $\angle(\delta x, \Delta x) > \psi$ while a descUpd is taken
where no calculation of $\angle_{est}(\delta x, \Delta x)$ is considered

$x_0 = \frac{3}{5} \cdot \hat{x}_0$ with $\hat{x}_0$ from [25], $A_0 = F'(x_0)$

Table 6.9: *Trigo* – APNLF related quantities

Discint					
x_0	Dim n	NLF	PNLF	APNLF	data
$100 \cdot \hat{x}_0$	$2 \cdot 10^3$	9	9	15	# steps
		18.79s	18.9s	9.99s	run time
	$4 \cdot 10^3$	9	9	15	# steps
		105.58s	105.86s	40.99s	run time
$500 \cdot \hat{x}_0$	$2 \cdot 10^3$	14	13	13	# steps
		29.45s	27.76s	24.16s	run time
	$4 \cdot 10^3$	14	13	13	# steps
		164.42s	152.45s	98.61s	run time

$\hat{x}_0$ initial guess from [25]

Table 6.10: *Discint* – Results of the NLF/PNLF/APNLF-algorithms

Discint					
x_0	Dim n	# Est. Fail.	# $\angle > \psi$ and noest	# descUpd	# purUpd
$100 \cdot \hat{x}_0$	$2 \cdot 10^3$	0	1	11	8
	$4 \cdot 10^3$	0	1	11	8
$500 \cdot \hat{x}_0$	$2 \cdot 10^3$	0	0	5	62
	$4 \cdot 10^3$	0	0	5	63

Est. Fail.: Estimation Failure, i.e.,
$\angle_{est}(\delta x, \Delta x) \leqslant \psi$ but $\angle(\delta x, \Delta x) > \psi$

$\angle > \psi$ and noest : $\angle(\delta x, \Delta x) > \psi$ while a descUpd is taken
where no calculation of $\angle_{est}(\delta x, \Delta x)$ is considered

$\hat{x}_0$ initial guess from [25], $A_0 = F'(x_0)$

Table 6.11: *Discint* – APNLF related quantities

6.1.3.1 Remarks on the results

All methods for all combinations of considered dimensions and initial guesses converge. In case of *Trigo* it is the solution $x_* = 0 \in \mathbb{R}^n$ and in case of *Discint* the unique solution x_* characterized by (6.9).

Comparing the run times of the algorithms Table 6.8 and 6.10 show the efficiency of the quasi-Newton approach of the APNLF-algorithm. Especially for higher dimensions, i.e., $n = 4 \cdot 10^3$ the run time of the Jacobian related approaches of the NLF- and PNLF-algorithm is dominated by the linear algebra operations of decomposing the Jacobian each step. Note that the minor difference in run time of the NLF- and PNLF- algorithm in case of *Trigo* and *Discint* with initial guess $100 \cdot \hat{x}_0$ is within MATLAB's measuring accuracy.

It seems that for the *Trigo* problem the APNLF-algorithm is indeed able to exploit the special structure (6.8) of the Jacobian. Only one purifying update (duophilic) for both considered dimensions occur. As it is seen from Table 6.9 this appears to be sufficient in order to keep the angle $\angle(\delta x, \Delta x) \leqslant \psi$ for the whole iteration. The convergence history depicted in Figure 6.11(a) confirms the dimension dependent r-order of convergence of the descent update which is greater or equal to the positive root ρ_n of $\rho^n(\rho - 1) - 1 = 0$, cf. (4.114) and Theorem 4.37. For $n = 2 \cdot 10^3$ it holds that $\rho_n \approx 1.0029$. This explains the nearly linear behavior of the error reduction in case of the APNLF-algorithm. Considering step sizes only marginal differences for the three considered algorithms appear, cf. Figure 6.11(b). A zoom reveals that the step sizes for the PNLF- and APNLF-algorithm are slightly larger than the ones for the NLF-algorithm. This confirms the impression from the basic test set. Even for the larger dimensions considered here it appears to be sufficient to consider the projected local nonlinearity onto the (approximate) Newton correction to determine reasonable step sizes.

In Figure 6.12 the taken step sizes for the problem *Discint* with $n = 2 \cdot 10^3$ are depicted. For $n = 4 \cdot 10^3$ the algorithms show a nearly identical behavior. The chosen step sizes reveal that the problem *Discint* is mildly nonlinear in general. However, there is some intermediate local nonlinearity which enforces a temporary but noticeable reduction of the step sizes. This rise in nonlinearity appears to be in a vicinity to the solution. Since $500 \cdot \hat{x}_0$ is further away from x_* then $100 \cdot \hat{x}_0$ the occurrence of a step size decrease is given for an iterate of higher index compared to the iteration for the initial guess $100 \cdot \hat{x}_0$. Due to the projectional aspect of the PNLF and APNLF there is less reduction in the taken step sizes of the PNLF- and APNLF-algorithm compared to the NLF-algorithm. However, it is remarkable that the intermediate rise in local nonlinearity is detected by the two new algorithms at all. Again, this confirms the impression that still sufficient information of the nonlinearity of F is taken into account though only its projection onto the (approximate) Newton correction is considered for the step size control.

The APNLF-algorithm reacts on the intermediate rise of local nonlinearity not only by reducing the step size but also by invoking purifying at and around the iterate where the step size reduction occurs, see Figure 6.13. For the initial guess $500 \cdot \hat{x}_0$ a considerable higher amount of purifying updates is applied compared to the initial guess $100 \cdot \hat{x}_0$. For the initial guess $100 \cdot \hat{x}_0$ it takes only four iterations until the step size reduction occurs. It appears that the Jacobian information

from the first iterate, recall that $A_0 = F'(100 \cdot \hat{x}_0)$ is chosen, still provides enough information for the critical iterates such that only minor purifying efforts are required. On the other hand, for $500^2 \cdot \hat{x}_0$ the initial Jacobian is too 'old' to provide significant information when the rise in local nonlinearity occurs. Therefore, purifying has to take care of providing this information.

In case of the initial guess $100 \cdot \hat{x}_0$ an additional purifying process at the eighth iterate is invoked. This is due to the fact that at the seventh iterate a descent update was chosen without checking $\angle_{est}(\delta x, \Delta x)$ but it holds that $\angle(\delta x, \Delta x) > \psi$ at this iterate, see Figure 6.14(b). This means like in the example *5spheres* 'post-compensating' purifying occurs. Figure 6.14(a) nicely reflects the benefit of this compensation. From the seventh iterate where $\angle(\delta x, \Delta x)$ is not within the tolerance to the eighth iterate the curve of the convergence history is rather flat whereas after the 'post-compensating' purifying at the eighth iterate the curve drops considerably. For the subsequent iterates only descent updates are considered. Like for the *Trigo* problem the curve of the convergence history reflects the expected r-order of convergence of the descent update.

For the initial guess $500 \cdot \hat{x}_0$ descent updates are only employed until the first six iterates. However, superlinear convergence is achieved, see Figure 6.15(a).

Throughout, for both initial guesses and considered dimensions only duophilic purifying updates are used. No singular Jacobian approximation or an approximation we consider to be ill-conditioned occurs and no post-purifying is invoked.

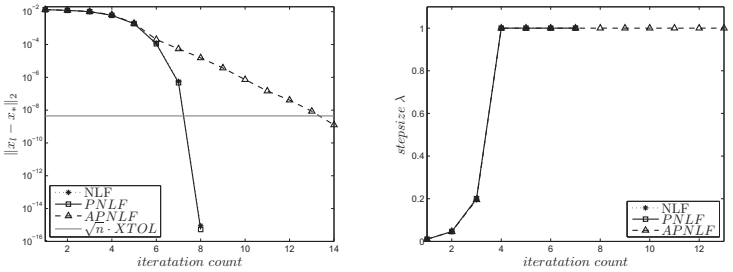
The angle estimator $\angle_{est}(\delta x, \Delta x)$ shows a very pleasant behavior. As it is seen from Table 6.11 no estimations failures occur at all.

To have an additional evidence of the benefit of the applied purifying concept for the APNLF-algorithm we run this algorithm without considering purifying, i.e., solely employing the descent update and compare it to the default algorithm. This is done for the problem *Discint* with the initial guess $500 \cdot \hat{x}_0$ and for $n = 2 \cdot 10^3$. The convergence history for both methods is depicted in Figure 6.16. Results in terms of required steps and run time are given in Table 6.12. An application of just one descent update per step is very cheap compared to a purifying process. However, the modified algorithm requires significantly more steps than the default algorithm so that the latter algorithm is more efficient in terms of run time.

Discint $n = 2 \cdot 10^3$ $x_0 = 500 \cdot \hat{x}_0$	APNLF	
	purif. applied	no purif.
# steps	13	67
run time	24.16s	28.44s

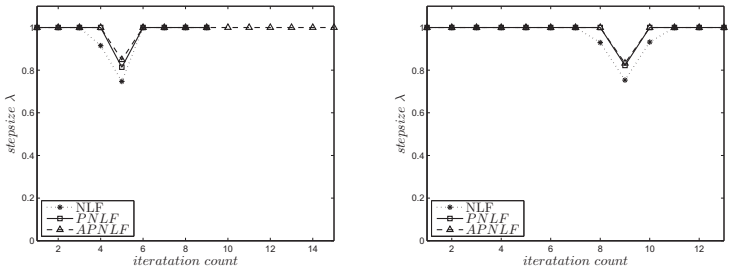
$\hat{x}_0$ initial guess from [25]

Table 6.12: *Discint* – Purifying vs. no purifying (i.e. solely descent updates are applied)



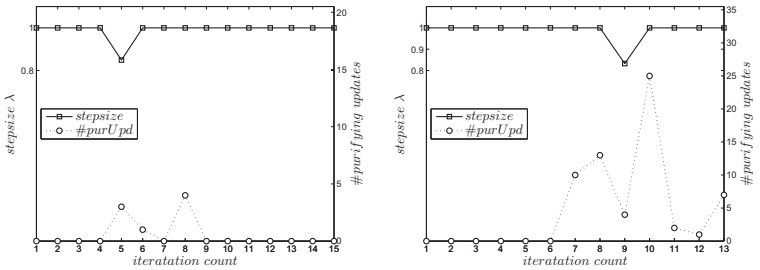
(a) Convergence history (b) Step sizes taken

Figure 6.11: *Trigo* – Convergence history and step sizes taken, $n = 2 \cdot 10^3$



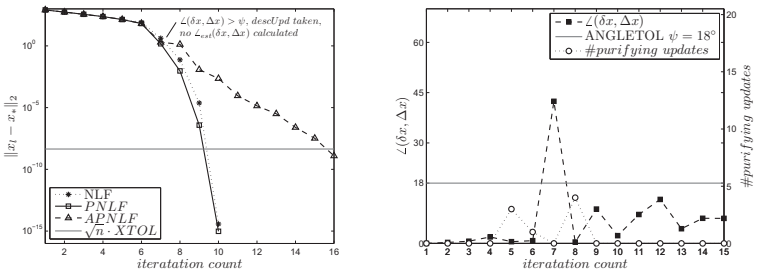
(a) $x_0 = 100 \cdot \hat{x}_0$. (b) $x_0 = 500 \cdot \hat{x}_0$.

Figure 6.12: *Discint* – step sizes taken, $n = 2 \cdot 10^3$



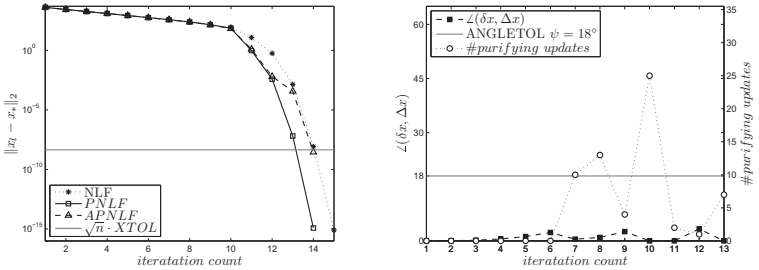
(a) $x_0 = 100 \cdot \hat{x}_0$ (b) $x_0 = 500 \cdot \hat{x}_0$

Figure 6.13: *Discint* – purifying per steps, $n = 2 \cdot 10^3$



(a) Convergence history (b) Angle/purifying relation

Figure 6.14: *Discint* – convergence history and angle/purifying relation, $n = 2 \cdot 10^3$, $x_0 = 100 \cdot \hat{x}_0$



(a) Convergence history (b) Angle/purifying relation

Figure 6.15: *Discint* – convergence history and angle/purifying relation, $n = 2 \cdot 10^3$, $x_0 = 500 \cdot \hat{x}_0$

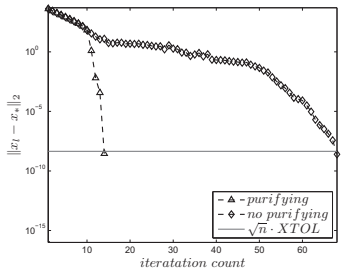


Figure 6.16: *Discint* – Convergence history for the APNLF-algorithm with purifying (default) and without purifying, $n = 2 \cdot 10^3$ and $x_0 = 500 \cdot \hat{x}_0$

6.2 Summary and Outlook

For the considered problems of the test set in Table 6.1 the new PNLF- and APNLF-algorithm perform very well. Convergence is achieved even for highly nonlinear problems. From the *Expsin* grid test we see that as expected the restricted monotonicity step size strategy turns out to be more reliable. However, the simple monotonicity strategy combined with the projected nonlinearity bound predictor also shows a sufficiently robust behavior as it is confirmed by the problems of the basic test set.

The basic approximation concept of the APNLF-algorithm to combine angle checks with purifying updates turns out to work very well. A failure of the estimator $\angle_{est}(\delta x, \Delta x)$ rarely occurs and if it is the case purifying compensates this failure in the subsequent step. Considering the two problems *Trigo* and *Discint* for higher dimensions the computational advantage of the APNLF-algorithm comes into play. In this scenario, updating a given LU-decomposition by means of rank-1 updates is more economic than to decompose a Jacobian each step.

The PNLF-algorithm may be seen as a refinement of our reference method which is based on the NLF. Though there is no significant boost in terms of efficiency for the considered problems in general, the overall performance of the PNLF-algorithm is slightly better than the performance of the reference method. The similar behavior of the reference method and the PNLF-algorithm gives rise to the impression that the local nonlinearity $\chi(\lambda)$ of F in the direction of the Newton correction at some iterate x, i.e.,

$$\chi(\lambda) = F'(x)^{-1}\big(F(x + \lambda\Delta x) - F(x) - \lambda F'(x)\Delta x\big)$$

is dominated by its projection onto the Newton correction – even for higher dimensional problems. Certainly, this behavior needs further investigation in order to describe it in a mathematically satisfying way.

The obtained results give good reason to hope that an adaption of the PNLF concept to the context of least squares problems (cf. Section 3.3) also provides satisfying results. Note that special care has to be taken to provide an adaption of the projected nonlinearity bound predictor since for its computation products of the form

$$\big(J(x_l)^- F(x_l)\big)^T J(x_l)^- \tag{6.10}$$

must be available where $J(x_l)^-$ is some generalized inverse of the Jacobian of F at x_l which is used to define the Gauß Newton correction $\Delta x_l = -J(x_l)^- F(x_l)$. The product (6.10) can be efficiently computed if $J(x_l)$ is decomposed as in (3.39). We omit details here.

Regarding the APNLF-algorithm, for future work the following modifications and adaptations may be considered:

- In the current implementation the angle conditions

$$\angle\big(\delta x_l, -\operatorname{grad} T(x_l|P_{l,k}A_{l,k}^{-1})^T\big) \leqslant \phi \qquad \text{and} \qquad \angle_{est}(\delta x_l, \Delta x_l) \leqslant \psi$$

  are checked for predefined fixed values of ϕ and ψ. However, the local behavior of F may change considerably in the course of the iteration. A too stringent choice of ϕ and ψ may cause an unnecessary amount of computational work per step whereas a too lax choice may result in too many iterations steps, or even worse, in no convergence. Hence, it sounds reasonable to develop a strategy for an *adaptive choice* of ϕ and ψ.

- Instead of quasi-Newton updates one may consider iterative methods in order to provide an approximation $\overline{H}_l$ (at least implicitly) to the inverse of the Jacobian at some iterate x_l. Note that for the APNLF-ansatz combined with the above stated angle checks the products (4.1) and (4.2), i.e.,

$$\overline{\delta x}_l := -\overline{H}_l F(x_l) \tag{6.11a}$$

  and

$$\overline{\delta x}_l^T \overline{H}_l \qquad \text{and} \qquad \overline{H}_l F'(x_l)\overline{\delta x}_l \tag{6.11b}$$

  must be available.

- If we formally substitute the projection onto the Newton correction by the identity matrix in the definition of the APNLF we obtain

$$\tfrac{1}{2}\|\overline{H}_l F(x)\|_2^2$$

  where as above $\overline{H}_l$ is an approximation to the inverse of the Jacobian $F'(x_l)$. For this *approximate natural level function* we can make use of the results from Chapter 5. With the choice $A = \overline{H}_l$ the polynomial model from Theorem 5.1 can be exploited to provide the basics for a step size control. In order to maintain sufficient quality of the approximations $\overline{H}_l$ this globalization approach can be combined with the purifying updates which we already employed in the context of the APNLF. As an alternative, one may consider iterative methods to provide the approximations $\overline{H}_l$. For this approach to be viable the iterative method only has to supply

$$\overline{\delta x}_l := -\overline{H}_l F(x_l) \qquad \text{and} \qquad \overline{H}_l F'(x_l)\overline{\delta x}_l$$

  compared to the three products (6.11).

Appendix

I: Proof of Theorem 4.39

The proof is basically an adaption of the proofs of Lemma 4.2.12 and Lemma 4.2.13 in [28]. Though, due to our affine covariant framework it requires some extra care. We split the proof into three parts. In the first part we will show that (4.120) is true for $j = l - 1$, i.e., $l = j + 1$ if the sequence $\{v_l\}$ fulfills the affine covariant residual property (4.102). In the second part we will prove (4.120) for the remaining indices $0 \leqslant j < l - 1$. Finally, in the third part we will show that (4.121) holds true.

We use the notation as introduced in Theorem 4.26.

I) From the definition of α_{l+1} we derive by the Cauchy-Schwarz inequality

$$\begin{aligned}
\alpha_{l+1} &= \frac{v_l^T(I - H_l J_{l+1})v_l}{v_l^T v_l} = \frac{v_l^T(I - H_l J_*)v_l}{v_l^T v_l} + \frac{v_l^T H_l(J_* - J_{l+1})v_l}{v_l^T v_l} \\
&= \frac{v_l^T(I - H_l J_*)v_l}{v_l^T v_l} + \frac{v_l^T J_*^{-1}(J_* - J_{l+1})v_l}{v_l^T v_l} \\
&\quad + \frac{v_l^T(H_l - J_*^{-1})J_* J_*^{-1}(J_* - J_{l+1})v_l}{v_l^T v_l} \\
&\leqslant \frac{v_l^T(I - H_l J_*)v_l}{v_l^T v_l} + \left[1 + \frac{\|v_l^T(I - H_l J_*)\|_2}{\|v_l\|_2}\right] \cdot \|J_*^{-1}(J_{l+1} - J_*)\|_2.
\end{aligned} \tag{12}$$

By the assumption that the transposed Dennis-Moré series is bounded the transposed Dennis-Moré property (4.94) holds. And since convergence is assumed there exist an index L and some constant φ such that for all $l \geqslant L$ it holds that

$$\frac{\|v_l^T E_l\|_2}{\|v_l\|_2} + \left[1 + \frac{\|v_l^T E_l\|_2}{\|v_l\|_2}\right] \cdot \|J_*^{-1}(J_{l+1} - J_*)\|_2 < \varphi < 1, \tag{13}$$

$$\alpha_{l,*} \leqslant \frac{\|v_l^T E_l\|_2}{\|v_l\|_2} < \frac{1}{2}. \tag{14}$$

From (12) and (13) it follows that $\alpha_{l+1} < \varphi$ and also,

$$\|H_{l+1} H_l^{-1}\|_2 \leqslant C_1 := \frac{1}{1 - \varphi}.$$

For $0 \leqslant l \leqslant L - 1$ we define C_2 via

$$C_2 := \max_{0 \leqslant l \leqslant L-1} \|H_{l+1} H_l^{-1}\|_2.$$

Thus, together

$$\|H_{l+1} H_l^{-1}\|_2 \leqslant C_3 := \max\{C_1, C_2\} < \infty \quad \forall l.$$

Now we show that

$$\frac{\|(I - H_{j+1} J_*)\delta x_j\|_2}{\|\delta x_j\|_2} \leqslant C_3(\psi\omega\|e_{j+1}\|_2 + c_j) \tag{15}$$

with $\lim_{j \to \infty} c_j = 0$ holds if the sequence $\{v_j\}$ has the affine covariant residual property. According to the definition in (4.82) and by means of the Sherman-Morrison-Woodbury

formula we obtain for H_{j+1} and δx_j

$$\begin{aligned}(I - H_{j+1}J_*)\delta x_j &= H_{j+1}(H_{j+1}^{-1} - J_*)\delta x_j \\ &= H_{j+1}\left[H_j^{-1}\left(I - \frac{v_j v_j^T}{v_j^T v_j}(I - H_j J_{j+1})\right)\delta x_j - J_*\delta x_j\right] \\ &= H_{j+1}\left[H_j^{-1}\left(I - \frac{v_j v_j^T}{v_j^T v_j}(I - H_j J_*)\right)\delta x_j - J_*\delta x_j\right] \\ &\quad + H_{j+1}H_j^{-1}\frac{v_j v_j^T}{v_j^T v_j}H_j J_* J_*^{-1}(J_{j+1} - J_*)\delta x_j.\end{aligned}$$

Since the sequence $\{v_j\}$ fulfills the affine covariant residual property (4.102) there is a $\xi_j \neq 0$ and a vector $r_j \in \mathbb{R}^n$ such that $\xi_j v_j = (I - H_j J_*)\delta x_j + r_j$ and

$$\|r_j\|_2 \leqslant c_j \|\delta x_j\|_2 \quad \text{with} \quad \lim_{j\to\infty} c_j = 0. \tag{16}$$

Substituting $\xi_j v_j - r_j$ for $(I - H_j J_*)\delta x_j$ yields

$$\begin{aligned}H_{j+1}\left[H_j^{-1}\left(I - \frac{v_j v_j^T}{v_j^T v_j}(I - H_j J_*)\right)\delta x_j - J_*\delta x_j\right] & \\ = H_{j+1}\left[H_j^{-1}\left(\delta x_j - \frac{v_j v_j^T}{v_j^T v_j}(\xi_j v_j - r_j)\right) - J_*\delta x_j\right] & \\ = H_{j+1}\left[H_j^{-1}\left(\delta x_j - \xi_j v_j + r_j - \Big(I - \frac{v_j v_j^T}{v_j^T v_j}\Big)r_j\right) - J_*\delta x_j\right] & \\ = H_{j+1}\left[H_j^{-1}\left(H_j J_*\delta x_j - \Big(I - \frac{v_j v_j^T}{v_j^T v_j}\Big)r_j\right) - J_*\delta x_j\right] & \\ = -H_{j+1}H_j^{-1}\Big(I - \frac{v_j v_j^T}{v_j^T v_j}\Big)r_j. &\end{aligned}$$

And hence, applying norms and the Lipschitz condition (4.66) we obtain

$$\|(I - H_{j+1}J_*)\delta x_j\|_2 \leqslant C_3(\psi\omega\|e_{j+1}\|_2\|\delta x_j\|_2 + \|r_j\|_2).$$

Dividing by $\|\delta x_j\|_2$ yields (15) because of (16). It remains to consider the cases where $j < l - 1$.

II) First assume that $j + 1 < l \leqslant L$. This is a finite number of cases. Thus, there exists $C_4 > 0$ such that

$$\frac{\|(I - H_l J_*)\delta x_j\|}{\|\delta x_j\|} \leqslant C_4 \sum_{k=j+1}^{l} \|e_k\|_2 + C_3 c_j \tag{17}$$

for $j+1 < l \leqslant L$ and c_j as given in (16). Now we consider the cases where $j+1 < l$ and $l > L$.

For ease of writing we abbreviate $m := l-1$. From (4.73) we obtain

$$\begin{aligned} E_{m+1}\delta x_j = &\left[I - \frac{1}{(1-\alpha_{m,*})}\cdot\frac{v_m v_m^T}{v_m^T v_m}\right] E_m \delta x_j \\ &+ \frac{1}{(1-\alpha_{m,*})}\cdot\frac{v_m v_m^T}{v_m^T v_m}(I - H_m J_*)E_m\delta x_j \\ &+ \frac{1}{(1-\alpha_{m,*})(1-\alpha_{m+1})}\cdot\frac{v_m^T H_m J_* J_*^{-1}(J_{m+1}-J_*)v_m}{v_m^T v_m}\cdot\frac{v_m v_m^T}{v_m^T v_m}H_m J_* E_m\delta x_j \\ &+ \frac{1}{(1-\alpha_{m+1})}\cdot\frac{v_m v_m^T}{v_m^T v_m}H_m J_* J_*^{-1}(J_{m+1}-J_*)H_m J_*\delta x_j. \end{aligned} \tag{18}$$

Since $m \geqslant L$ we can exploit (14) to obtain the estimate

$$\left\| I - \frac{1}{1-\alpha_{m,*}}\cdot\frac{v_m v_m^T}{v_m^T v_m}\right\|_2 \leqslant 1.$$

Thus,

$$\begin{aligned} &\left\|\left[I - \frac{1}{(1-\alpha_{m,*})}\cdot\frac{v_m v_m^T}{v_m^T v_m}\right] E_m\delta x_j\right\|_2 \\ &\qquad + \left\|\frac{1}{(1-\alpha_{m,*})}\cdot\frac{v_m v_m^T}{v_m^T v_m}(I - H_m J_*)E_m\delta x_j\right\|_2 \leqslant \Big(1 + 2\frac{\|v_m^T E_m\|_2}{\|v_m\|_2}\Big)\|E_m\delta x_j\|_2. \end{aligned}$$

Recall that by the assumptions $\|H_m J_*\| \leqslant \psi$. Together with the derived bounds for α_{m+1} and $\alpha_{m,*}$ and with the Lipschitz condition (4.66) we obtain for the norms of the third and fourth summand on the right hand side of (18) the estimate

$$\begin{aligned} &\left\|\frac{1}{(1-\alpha_{m,*})(1-\alpha_{m+1})}\cdot\frac{v_m^T H_m J_* J_*^{-1}(J_{m+1}-J_*)v_m}{v_m^T v_m}\cdot\frac{v_m v_m^T}{v_m^T v_m}H_m J_* E_m\delta x_j\right\|_2 \\ &\qquad + \left\|\frac{1}{(1-\alpha_{m+1})}\cdot\frac{v_m v_m^T}{v_m^T v_m}H_m J_* J_*^{-1}(J_{m+1}-J_*)H_m J_*\delta x_j\right\|_2 \\ &\qquad\qquad \leqslant 2\frac{\psi^2(\frac{3}{2}+\psi)\omega}{1-\varphi}\|e_{m+1}\|_2\|\delta x_j\|_2. \end{aligned}$$

Hence, with

$$C_5 := 2\frac{\psi^2(\frac{3}{2}+\psi)\omega}{1-\varphi}$$

we obtain for $\|E_{m+1}\delta x_j\|_2$ the upper bound

$$\|E_{m+1}\delta x_j\|_2 \leqslant \Big(1 + 2\frac{\|v_m^T E_m\|_2}{\|v_m\|_2}\Big)\|E_m\delta x_j\|_2 + C_5\|e_{m+1}\|_2\|\delta x_j\|_2. \tag{19}$$

Set $s = \max\{j+1, L\}$. Applying (19) recursively yields

$$\begin{aligned} \|E_{m+1}\delta x_j\|_2 \leqslant &\left[\prod_{t=s}^{m}\Big(1 + 2\frac{\|v_t^T E_t\|_2}{\|v_t\|_2}\Big)\right]\|E_s\delta x_j\|_2 \\ &+ C_5\sum_{k=s+1}^{m+1}\left[\prod_{t=k}^{m}\Big(1 + 2\frac{\|v_t^T E_t\|_2}{\|v_t\|_2}\Big)\right]\|e_k\|_2\|\delta x_j\|_2. \end{aligned} \tag{20}$$

Before we proceed we show that there is a C_6 so that

$$\prod_{t=0}^{\infty}\Big(1+2\frac{\|v_t^T E_t\|_2}{\|v_t\|_2}\Big) \leqslant C_6 < \infty. \tag{21}$$

It is readily seen that there exists some positive constant $C_{6,1}$ such that we can write the left hand side of (21) as

$$C_{6,1}\cdot\prod_{t=L}^{\infty}\Big(1+2\frac{\|v_t^T E_t\|_2}{\|v_t\|_2}\Big).$$

For arbitrary but fixed $\hat{t} \geqslant L$ we consider the partial product

$$\prod_{t=L}^{\hat{t}}\Big(1+2\frac{\|v_t^T E_t\|_2}{\|v_t\|_2}\Big).$$

Applying the natural logarithm gives

$$\sum_{t=L}^{\hat{t}}\ln\Big(1+2\frac{\|v_t^T E_t\|_2}{\|v_t\|_2}\Big).$$

From (14) we have for all $t \geqslant L$ that $\|v_t^T E_t\|_2/\|v_t\|_2 < \frac{1}{2}$ and hence by the natural logarithm series for $\ln(1+q)$, $q \in (-1,1)$, and by the Leibniz criterion we obtain

$$\sum_{t=L}^{\hat{t}}\ln\Big(1+2\frac{\|v_t^T E_t\|_2}{\|v_t\|_2}\Big) \leqslant 2\sum_{t=L}^{\hat{t}}\frac{\|v_t^T E_t\|_2}{\|v_t\|_2}.$$

And for $\hat{t}\to\infty$,

$$\sum_{t=L}^{\infty}\ln\Big(1+2\frac{\|v_t^T E_t\|_2}{\|v_t\|_2}\Big) \leqslant 2\sum_{t=L}^{\infty}\frac{\|v_t^T E_t\|_2}{\|v_t\|_2} < \infty$$

by the assumption that the transposed Dennis-Moré series is bounded. Hence, there exists a positive constant $C_{6,2}$ such that

$$\prod_{t=L}^{\infty}\Big(1+2\frac{\|v_t^T E_t\|_2}{\|v_t\|_2}\Big) \leqslant C_{6,2}.$$

Defining C_6 as $C_{6,1}\cdot C_{6,2}$ yields (21). Inserting this result into (20) leads to

$$\|E_{m+1}\delta x_j\|_2 \leqslant C_6\|E_s\delta x_j\|_2 + C_5C_6\sum_{k=s+1}^{m+1}\|e_k\|_2\|\delta x_j\|_2. \tag{22}$$

If $s = j+1$ we obtain by means of (15),

$$\begin{aligned}\|E_{m+1}\delta x_j\|_2 &\leqslant C_3C_6c_j\|\delta x_j\|_2\\ &\quad + C_3C_6\psi\omega\|e_{j+1}\|_2\|\delta x_j\|_2 + C_5C_6\sum_{k=j+2}^{m+1}\|e_k\|_2\|\delta x_j\|_2.\end{aligned} \tag{23}$$

And for $s = L$ from (17),

$$\begin{aligned}\|E_{m+1}\delta x_j\|_2 &\leqslant C_3C_6c_j\|\delta x_j\|_2\\ &\quad + C_4C_6\sum_{k=j+1}^{L}\|e_k\|_2\|\delta x_j\|_2 + C_5C_6\sum_{k=L+1}^{m+1}\|e_k\|_2\|\delta x_j\|_2.\end{aligned} \tag{24}$$

Combining the results of (23) and (24) by defining

$$\tilde{c}_j := C_3 C_6 c_j, \quad C := C_6 \cdot \max\{C_3\psi\omega, C_4, C_5\},$$

dividing by $\|\delta x_j\|_2$ and by undoing the substitution $m = l - 1$ finally yields

$$\frac{\|E_l \delta x_j\|_2}{\|\delta x_j\|_2} \leqslant C \sum_{k=j+1}^{l} \|e_k\|_2 + \tilde{c}_j, \quad 0 \leqslant j < l,$$

which is just (4.120).

III) If $v_j = -H_j F_{j+1}$ then from (4.102) and Proposition 4.34 it follows that c_j may be chosen as

$$c_j = \psi \frac{\omega}{2}(\|e_j\|_2 + \|e_{j+1}\|_2).$$

For $\|e_{j+1}\|_2 \leqslant \|e_j\|_2$ we have

$$\tilde{c}_j = C_3 C_6 c_j \leqslant C_3 C_6 \psi\omega \|e_j\|_2 \leqslant C\|e_j\|_2$$

and (4.121) is true for $\tilde{C} := C$. In the opposed case we define $\tilde{C} := C + C_3 C_6 \psi \frac{\omega}{2}$ and hence,

$$\frac{\|E_l \delta x_j\|_2}{\|\delta x_j\|_2} \leqslant \tilde{C} \sum_{k=j}^{l} \|e_k\|_2.$$

This concludes the proof.

II: Pseudo-codes of some algorithmic aspects of the quasi-Newton iteration from Section 4.4

Algorithm A.1 (Determine whether a descent update or a purifying process will be executed)

---

given: $A_{l,0} \in \mathbb{R}^{n\times n}$ nonsingular, $\overline{\delta x}_{l,0} = -A_{l,0}^{-1}F_l$ with $F_l := F(x_l) \neq 0$, $J_l := F'(x_l)$,
  $\bar{\varepsilon} \ll 1$, $\tilde{\phi}$ with $\frac{\pi}{2} > \tilde{\phi} \geqslant \phi$ for ϕ from (4.124),
  $\lambda_{0,0}$ if $l = 0$ or λ_{l-1} if $l > 0$
set $\lambda\text{pred} = \text{true}$
set $A_w = A_{l,0}$
determine $\hat{g}_{l,0}^T = \overline{\delta x}_{l,0}^T A_{l,0}^{-1}$
determine $g_{l,0}^T = \hat{g}_{l,0}^T J_l$ ▷ i.e. $-\operatorname{grad} T(x_l | P_{l,0} A_{l,0}^{-1})$
determine $h_{l,0} = g_{l,0}^T \overline{\delta x}_{l,0}$
determine $1 - \alpha_{l,0} = h_{l,0} / \|\overline{\delta x}_{l,0}\|_2^2$
if $|1 - \alpha_{l,0}| < \bar{\varepsilon}$ **then** ▷ descent update of $A_{l,0}$ close to singular
  determine $t_{l,0}^T = \overline{\delta x}_{l,0}^T - g_{l,0}^T$ ▷ i.e. $\overline{\delta x}_{l,0}^T (I - A_{l,0}^{-1} J_l)$
  determine $u_{l,0} = J_l \overline{\delta x}_{l,0}$
  set $y_{l,0} = -F_l$
  determine $d_{l,0} = \|\overline{\delta x}_{l,0}\|_2^2 - h_{l,0}$ ▷ i.e. $\overline{\delta x}_{l,0}^T (I - A_{l,0}^{-1} J_l) \overline{\delta x}_{l,0}$
  set $\text{purswitch}_{l,0} = \text{'duophilic'}$
  go to purifying process ▷ Algorithm A.2
else
  determine $s_{l,0} = (1 - \alpha_{l,0})^{-1}$
  determine $a_{l,0} = \|\overline{\delta x}_{l,0} - s_{l,0} g_{l,0}\|_2 / \|\overline{\delta x}_{l,0}\|_2$
  if $\left(\angle(s_{l,0}\overline{\delta x}_{l,0}, g_{l,0}) \leqslant \tilde{\phi}\right)$ && $(a_{l,0} < 1)$ **then**
    if $l > 0$ **then**
      if $\lambda_{l-1} = 1$ **then**
        determine $[\hat{\omega}]_l = 2\frac{|\hat{g}_{l,0}^T F_{l-1} + \|\overline{\delta x}_{l,0}\|_2^2 + \lambda_{l-1} g_{l,0}^T \delta x_{l-1})|}{\lambda_{l-1}^2 \|\delta x_{l-1}\|_2^2 \cdot \|\overline{\delta x}_{l,0}\|_2}$
        ▷ i.e. $2\frac{\|P_{l,0} A_{l,0}^{-1}(F_{l-1} - F_l + \lambda_{l-1} J_l \delta x_{l-1})\|_2}{\lambda_{l-1}^2 \|\delta x_{l-1}\|_2^2}$
        determine $\lambda_{l,0} = \min\left(\frac{|1-\alpha_{l,0}|^2}{[\hat{\omega}]_l \|\overline{\delta x}_{l,0}\|_2}, 1\right)$
        ▷ i.e. $\min\left(\frac{1}{(|1-\alpha_{l,0}|)^{-1}[\hat{\omega}]_l \|\delta x_l\|_2}, 1\right)$
        if $\lambda_{l,0} = 1$ **then**
          use_decent_update = true
        else
          use_decent_update = false
        end if
      else
        use_decent_update = false
      end if
    else if $\lambda_{0,0} = 1$ **then**
      use_decent_update = true

else
  use_decent_update = false
end if
else
  use_decent_update = false
end if
end if
if use_decent_update **then**
  set $\delta x_l = \frac{1}{1-\alpha_{l,0}}\overline{\delta x}_{l,0}$, $\bar{k}_l = 0$
  set directang$_l$ = false
  set predtype$_l$ = 'nlb'
  use $T(x|P_{l,0}A_{l,0})$ for step size control
  go to step size control
else
  go to purifying check ▷ Algorithm A.3
end if

Algorithm A.2 (Purifying process at purifying index k)

---

given: all well defined quantities from Algorithm A.1 or A.3, respectively,
  for current purifying index k,
  J_l and F_l as in Algorithm A.1,
  large constant K, $\varepsilon_{sing} \ll 1$, $\varepsilon \geqslant \tilde{\varepsilon}$ where $\tilde{\varepsilon}$ from Algorithm A.1
switch purswitch$_{l,k}$
  case 'duophilic'
    $A_{l,k+1} = A_{l,k} - \frac{(y_{l,k}-u_{l,k})t_{l,k}^T}{d_{l,k}}$
    ▷ i.e. $A_{l,k+1} = A_{l,k} - \frac{(A_{l,k}-J_l)\overline{\delta x}_{l,k}\overline{\delta x}_{l,k}^T(I-A_{l,k}^{-1}J_l)}{\overline{\delta x}_{l,k}^T(I-A_{l,k}^{-1}J_l)\overline{\delta x}_{l,k}}$
  case 'Newton-philic'
    $A_{l,k+1} = A_{l,k} - \frac{(y_{l,k}-u_{l,k})t_{l,k}^T}{\|r_{l,k}\|_2^2}$
    ▷ i.e. $A_{l,k+1} = A_{l,k} - \frac{(A_{l,k}-J_l)\overline{\delta x}_{l,k}\left(A_w^{-1}(A_{l,k}-J_l)\overline{\delta x}_{l,k}\right)^T A_w^{-1}(A_{l,k}-J_l)}{\|A_w^{-1}(A_{l,k}-J_l)\overline{\delta x}_{l,k}\|_2^2}$
  case 'gradientphilic'
    $A_{l,k+1} = A_{l,k} - \frac{(A_{l,k}t_{l,k}-u_{l,k})t_{l,k}^T}{\|t_{l,k}\|_2^2}$
    ▷ i.e. $A_{l,k+1} = A_{l,k} - \frac{(A_{l,k}-J_l)(I-A_{l,k}^{-1}J_l)^T\overline{\delta x}_{l,k}\overline{\delta x}_{l,k}^T(I-A_{l,k}^{-1}J_l)}{\|\overline{\delta x}_{l,k}^T(I-A_{l,k}^{-1}J_l)\|_2^2}$
end switch
set $k = k+1$
determine an LU-decomposition of $A_{l,k}$: $PA_{l,k} = LU$
let $u_{11}, \ldots, u_{nn}$ be the diagonal elements of U
if at least one u_{ii} is zero **then**

20: determine $\overline{\delta x}_{l,k} \in \ker(U) \setminus \{0\}$
21: set badly_conditioned_or_singular = true
22: **else if** $\max_{i,j} \frac{|u_{ii}|}{|u_{jj}|} > K$ **then**
23: determine $\overline{\delta x}_{l,k}$ as an approximation to a singular vector w.r.t. the smallest
24: singular value of $A_{l,k}$
25: set badly_conditioned_or_singular = true
26: **else**
27: determine $\overline{\delta x}_{l,k} = -A_{l,k}^{-1} F_l$
28: set badly_conditioned_or_singular = false
29: **end if**
30: **if** badly_conditioned_or_singular **then**
31: determine $u_{l,k} = J_l \overline{\delta x}_{l,k}$
32: determine $y_{l,k} = A_{l,k} \overline{\delta x}_{l,k}$
33: determine $r_{l,k} = A_w^{-1}(y_{l,k} - u_{l,k})$
34: **if** $\|r_{l,k}\|_2 / \|\overline{\delta x}_{l,k}\|_2 < \varepsilon_{sing}$ **then** ▷ i.e. $\frac{\|A_w^{-1}(A_{l,k} - J_l)\overline{\delta x}_{l,k}\|_2}{\|\overline{\delta x}_{l,k}\|_2} < \varepsilon_{sing}$
35: set exit_cond = J_l deemed (nearly) singular
36: abort algorithm
37: **else if** $k = n$ **then**
38: set exit_cond = too many purifying updates
39: abort algorithm
40: **else** ▷ push towards Newton direction
41: determine $t_{l,k}^T = r_{l,k}^T A_w^{-1} A_{l,k} - r_{l,k}^T A_w^{-1} J_l$ ▷ i.e. $(A_w^{-1}(A_{l,k} - J_l)\overline{\delta x}_{l,k})^T A_w^{-1}(A_{l,k} - J_l)$
42: set purswitch$_{l,k}$ = 'Newton-philic'
43: **go to** line 5
44: **end if**
45: **else** ▷ current $A_{l,k}$ nonsingular
46: set $A_w = A_{l,k}$
47: determine $\hat{g}_{l,k}^T = \overline{\delta x}_{l,k}^T A_{l,k}^{-1}$
48: determine $g_{l,k}^T = \hat{g}_{l,k}^T J_l$ ▷ i.e. $-\operatorname{grad} T(x_l | P_{l,k} A_{l,k}^{-1})$
49: determine $h_{l,k} = g_{l,k}^T \overline{\delta x}_{l,k}$
50: determine $1 - \alpha_{l,k} = h_{l,k} / \|\overline{\delta x}_{l,k}\|_2^2$
51: **if** $|1 - \alpha_{l,k}| < \varepsilon$ **then** ▷ descent update of $A_{l,k}$ close to singular
52: **if** $k < n$ **then**
53: determine $t_{l,k}^T = \overline{\delta x}_{l,k}^T - g_{l,k}^T$ ▷ i.e. $\overline{\delta x}_{l,k}^T (I - A_{l,k}^{-1} J_l)$
54: determine $u_{l,k} = J_l \overline{\delta x}_{l,k}$
55: set $y_{l,k} = -F_l$
56: determine $d_{l,k} = \|\overline{\delta x}_{l,k}\|_2^2 - h_{l,k}$ ▷ i.e. $\overline{\delta x}_{l,k}^T (I - A_{l,k}^{-1} J_l) \overline{\delta x}_{l,k}$
57: set purswitch$_{l,k}$ = 'duophilic'
58: **go to** line 5
59: **else**
60: set exit_cond = too many purifying updates

61: abort algorithm
62: **end if**
63: **end if**
64: determine $s_{l,k} = (1 - \alpha_{l,k})^{-1}$
65: determine $a_{l,k} = \|\overline{\delta x}_{l,k} - s_{l,k} g_{l,k}\|_2 / \|\overline{\delta x}_{l,k}\|_2$
66: **go to** purifying check ▷ Algorithm A.3
67: **end if**

Algorithm A.3 (Purifying check)

1: given: all well defined quantities from Algorithm A.1 or A.2, respectively,
2: additionally, k_{saved}, post_purifying from Algorithm A.4 or A.5 if invoked from there,
3: $\lambda_{0,0}$, J_l and F_l as in Algorithm A.1,
4: $\varepsilon_2, \varepsilon_3 \ll 1$, ϕ from (4.124) with $\phi \leqslant \tilde{\phi}$, ▷ $\tilde{\phi}$ from Algorithm A.1
5: $\psi \ll \pi/2$ from (4.124)
6: **if** $(\angle(s_{l,k} \cdot \overline{\delta x}_{l,k}, g_{l,k}) \leqslant \phi)$ && $(a_{l,k} < 1)$ **then** ▷ gradient part good enough
7: $u_{l,k} = J_l \overline{\delta x}_{l,k}$
8: determine $\hat{u}_{l,k} = A_{l,k}^{-1} u_{l,k}$
9: determine $\widetilde{r}_{l,k} = \overline{\delta x}_{l,k} - s_{l,k} \hat{u}_{l,k}$ ▷ i.e. $(I - \frac{1}{1-\alpha_{l,k}} A_{l,k}^{-1} J_l) \overline{\delta x}_{l,k}$
10: **if** $(\|\widetilde{r}_{l,k}\|_2 / \|\overline{\delta x}_{l,k}\|_2 < 1)$ && $(\angle_{est}(s_{l,k} \cdot \overline{\delta x}_{l,k}, \Delta x_l) \leqslant \psi)$ **then**
11: ▷ Newton part good enough
12: set $\delta x_l = \frac{1}{1-\alpha_{l,k}} \overline{\delta x}_{l,k}$, $\bar{k}_l = k$
13: set directang$_l$ = true
14: ▷ check where we come from to know what to do
15: **if** post_purifying && $k = k_{saved}$ **then**
16: ▷ no post purifying necessary though ϕ and ψ are more restrictive
17: **return**
18: **else if** λpred && $k = k_{saved}$ **then** ▷ predictor was bad, handle it outside
19: **return**
20: **else if** $l > 0$ **then** ▷ determine a predictor
21: **if** use_nlb_pred **then**
22: determine $[\hat{\omega}]_l$ from Algorithm A.1, line 23
23: where the index $(l, 0)$ is substituted by (l, k)
24: determine $\lambda_{l,0} = \min\left(\frac{|1-\alpha_{l,k}|^2}{[\hat{\omega}]_l \|\overline{\delta x}_{l,k}\|_2}, 1\right)$
25: ▷ i.e. $\min\left(\frac{1}{(|1-\alpha_{l,k}|)^{-1} [\hat{\omega}]_l \|\delta x_l\|_2}, 1\right)$
26: set predtype$_l$ = 'nlb'
27: **else**
28: **if** directang$_{l-1}$ && use_dflh_pred **then** ▷ Deuflhard-like predictor
29: $[\mathring{\omega}]_l = \frac{|g_{l,k}^T \delta x_{l-1} - (1-\alpha_{l-1,k_{l-1}})^{-1} g_{l,k}^T u_{l-1,k_{l-1}}|}{\lambda_{l-1} \|\delta x_{l-1}\|_2^2 \|\overline{\delta x}_{l,k}\|_2}$

▷ i.e. $\frac{\|P_{l,k}A_{l,k}^{-1}(J_l - J_{l-1})\delta x_{l-1}\|_2}{\lambda_{l-1}\|\delta x_{l-1}\|_2^2}$

set predtype$_l$ = 'qdflh'

else ▷ simple predictor

determine $[\hat{\omega}]_l = [\omega]_{l-1}(\lambda_{l-1})$

set predtype$_l$ = 'simple'

end if

determine $\lambda_{l,0} = \min\left(\frac{|1-\alpha_{l,k}|^2}{[\omega]_l\|\overline{\delta x}_{l,k}\|_2}, 1\right)$ ▷ i.e. $\min\left(\frac{1}{([1-\alpha_{l,k}|)^{-1}[\omega]_l\|\delta x_l\|_2}, 1\right)$

end if

else

set $\lambda_{l,0} = \lambda_{0,0}$ ▷ user given choice for $l = 0$

end if

set λpred = true

use $T(x|P_{l,k}A_{l,k})$ for step size control

go to step size control ▷ if post_purifying was set it shall remain set!

else if $k = n$ **then**

set exit_cond = too many purifying updates

abort algorithm

else

determine $t_{l,k}^T = \overline{\delta x}_{l,k}^T - g_{l,k}^T$ ▷ i.e. $\overline{\delta x}_{l,k}^T(I - A_{l,k}^{-1}J_l)$

determine $d_{l,k} = \|\overline{\delta x}_{l,k}\|_2^2 - h_{l,k}$ ▷ i.e. $\overline{\delta x}_{l,k}^T(I - A_{l,k}^{-1}J_l)\overline{\delta x}_{l,k}$

if $\alpha_{l,k} < \varepsilon_2$ && $\frac{|d_{l,k}|}{\|t_{l,k}\|_2\|\overline{\delta x}_{l,k}\|_2} < \varepsilon_3$ **then**

▷ $\left|\cos\left[\angle\left(\overline{\delta x}_{l,k}, (\overline{\delta x}_{l,k}^T(I - A_{l,k}^{-1}J_l))^T\right)\right]\right| < \varepsilon_3$

determine $r_{l,k} = \overline{\delta x}_{l,k} - \hat{u}_{l,k}$ ▷ i.e. $(I - A_{l,k}^{-1}J_l)\overline{\delta x}_{l,k}$

set $t_{l,k}^T = r_{l,k}^T - r_{l,k}^T A_{l,k}^{-1}J_l$ ▷ i.e. $((I - A_{l,k}^{-1}J_l)\overline{\delta x}_{l,k})^T(I - A_{l,k}^{-1}J_l)$

set purswitch$_{l,k}$ = 'Newton-philic'

else

set purswitch$_{l,k}$ = 'duophilic'

end if

set $y_{l,k} = -F_l$

go to purifying process ▷ Algorithm A.2

end if

else if $k = n$ **then**

set exit_cond = too many purifying updates

abort algorithm

else ▷ no good gradient

determine $t_{l,k}^T = \overline{\delta x}_{l,k}^T - g_{l,k}^T$ ▷ i.e. $\overline{\delta x}_{l,k}^T(I - A_{l,k}^{-1}J_l)$

if $\alpha_{l,k} < \varepsilon_2$ **then**

determine $u_{l,k} = J_l t_{l,k}$

set purswitch$_{l,k}$ = 'gradientphilic'

else

determine $u_{l,k} = J_l\overline{\delta x}_{l,k}$

71: determine $d_{l,k} = \|\overline{\delta x}_{l,k}\|_2^2 - h_{l,k}$ ▷ i.e. $\overline{\delta x}_{l,k}^T (I - A_{l,k}^{-1} J_l) \overline{\delta x}_{l,k}$
72: set $y_{l,k} = -F_l$
73: set purswitch$_{l,k}$ = 'duophilic'
74: **end if**
75: **go to** purifying process ▷ Algorithm A.2
76: **end if**

Algorithm A.4 (Failed step size $\lambda_{l,j}$, simple monotonicity)

1: given: corrector step size $\lambda_{l,j}^c = 1/|1 - \alpha_{l,k}|^{-1} [\omega]_{l,k}(\lambda_{l,j}) \|\delta x_l\|_2$, cf. (4.158),
2: $\frac{1}{2} <$ PRED_BADTOL $\leqslant 1$, $0 < \lambda_{min} \leqslant \lambda_{thresh} \ll 1$,
3: ϕ and ψ from (4.124),
4: λ_{safe}, ▷ if not empty it is a step size which provides descent
5: set $k_{saved} = k$ ▷ save current purifying index
6: **if** $\lambda_{l,j} \leqslant \lambda_{thresh}$ && directang$_l$ && ¬post_purifying && isempty(λ_{safe}) **then**
7: set post_purifying = true
8: set $\phi = \phi/2$, $\psi = \psi/2$ ▷ recover old values after step size control
9: **go to** purifying check ▷ if purifying index changes start anew
10: **if** $\lambda_{l,j} \leqslant \lambda_{min}$ **then** ▷ else proceed as usual: next comes line 39
11: set exit_cond = too small step size
12: abort algorithm
13: **end if**
14: **else if** $\lambda_{l,j} \leqslant \lambda_{min}$ **then**
15: set exit_cond = too small step size
16: abort algorithm
17: **end if**
18: **if** ¬isempty(λ_{safe}) **then**
19: set $\lambda_l = \lambda_{safe}$
20: terminate step size control (restore old values of ϕ and ψ if necessary)
21: **end if**
22: **if** λpred && $l > 0$ **then** ▷ $j = 0$, give other predictors a try
23: set $\lambda_{bad} = \lambda_{l,j}$
24: **if** ¬directang$_l$ **then** ▷ true implies that $k = 0$
25: **go to** purifying check ▷ if purifying index changes start anew
26: **end if**
27: determine the two predictors λ_I and λ_{II} (Deuflhard-like predictor only
28: if directang$_{l-1}$ is true) which were not chosen to determine $\lambda_{l,0}$
29: set $\hat{\lambda} = 0$
30: **if** $\lambda_{bad} \cdot$ PRED_BADTOL $> \lambda_I$ **then**

31: set $\hat{\lambda} = \lambda_I$
32: **end if**
33: **if** $\lambda_{bad} \cdot \text{PRED\_BADTOL} > \lambda_{II}$ **then**
34: set $\hat{\lambda} = \max(\hat{\lambda}, \lambda_{II})$
35: **end if**
36: set $\lambda_{l,0} = \max\big(\max(\lambda^c_{l,j}, \hat{\lambda}), \lambda_{min}\big)$
37: set λpred = false
38: **else**
39: set $\lambda_{l,j+1} = \max(\lambda^c_{l,j}, \lambda_{min})$
40: **end if**

Algorithm A.5 (Failed step size $\lambda_{l,j}$, restricted monotonicity)

1: given: corrector step size $\lambda^c_{l,j} = 1/|1 - \alpha_{l,k}|^{-1}[\omega]_{l,k}(\lambda_{l,j})\|\delta x_l\|_2$, cf. (4.158),
2: $\frac{1}{2} <$ PRED_BADTOL $\leqslant 1$, $0 < \lambda_{min} \leqslant \lambda_{thresh} \ll 1$,
3: ϕ and ψ from (4.124),
4: set $k_{saved} = k$ ▷ save current purifying index
5: **if** $\lambda_{l,j} \leqslant \lambda_{thresh}$ && directang$_l$ && ¬post_purifying **then**
6: set post_purifying = true
7: set $\phi = \phi/2$, $\psi = \psi/2$ ▷ recover old values after step size control
8: **go to** purifying check ▷ if purifying index changes start anew
9: **if** $\lambda_{l,j} \leqslant \lambda_{min}$ **then** ▷ else proceed as usual: next comes line 34
10: set exit_cond = too small step size
11: abort algorithm
12: **end if**
13: **else if** $\lambda_{l,j} \leqslant \lambda_{min}$ **then**
14: set exit_cond = too small step size
15: abort algorithm
16: **end if**
17: **if** λpred && $l > 0$ **then** ▷ $j = 0$, give other predictors a try
18: set $\lambda_{bad} = \lambda_{l,j}$
19: **if** ¬directang$_l$ **then** ▷ true implies that $k = 0$
20: **go to** purifying check ▷ if purifying index changes start anew
21: **end if**
22: determine the two predictors λ_I and λ_{II} (Deuflhard-like predictor only
23: if directang$_{l-1}$ is true) which were not chosen to determine $\lambda_{l,0}$
24: set $\hat{\lambda} = 0$
25: **if** $\lambda_{bad} \cdot \text{PRED\_BADTOL} > \lambda_I$ **then**
26: set $\hat{\lambda} = \lambda_I$

end if
if $\lambda_{bad} \cdot \text{PRED\_BADTOL} > \lambda_{II}$ **then**
set $\hat{\lambda} = \max(\hat{\lambda}, \lambda_{II})$
end if
set $\lambda_{l,0} = \max\big(\max(\lambda_l^c, \hat{\lambda}), \lambda_{min}\big)$
set $\lambda\text{pred} = \text{false}$
else
set $\lambda_{l,j+1} = \max(\lambda_{l,j}^c, \lambda_{min})$
end if

III: Chosen values of some additional constants for the NLF/PNLF/APNLF-algorithm

Here we state concrete values of some of the constants which control the behavior of the NLF-, PNLF- and APNLF-algorithm. For details about the conditions in which these constants appear consider the references in the following table.

Constant	Value	Context	Reference
$\underline{\eta}$ BADTOL	$\frac{1}{2}$ $\frac{85}{100}$	Increasing step sizes (simple monotonicity)	(3.101)
PRED_BADTOL	$\frac{7}{10}$	PNLF: consideration of unprojected predictors APNLF: consideration of other predictors	Paragraph 3.4.1.7 Appendix II, Alg. A.4, A.5
PRED_RED	$\frac{5}{12}$	reduction factor if projected predictor fails and no unprojected is available	Paragraph 3.4.1.7
ε_2	10^{-10}	decision duo- or gradient/Newton-philic purifying update	(4.137) Appendix II, Alg. A.3
ε_3	10^{-10}	decision duo- or Newton-philic purifying update	(4.142) Appendix II, Alg. A.3
K	10^{14}	condition check for U where $PA_{l,k} = LU$	(4.143) Appendix II, Alg. A.2
ε_{sing}	10^{-12}	singularity check for $A_{l,k}$	(4.144) Appendix II, Alg. A.2
λ_{thresh}	λ_{min}	consideration of post-purifying	Paragraph 4.4.6.4

Table A.1: Values of additional constants for the NLF/PNLF/APNLF-algorithm

In case of post-purifying the more restrictive values of ϕ and ψ for the angle checks (6.4) are chosen to be one half of the default values.

Bibliography

[1] ADOL-C. `http://www.coin-or.org/projects/ADOL-C.xml`. Project manager: Andrea Walther.

[2] U. M. Ascher and M. R. Osborne. A note on solving nonlinear equations and the 'natural' criterion function. *Journal of Optimization Theory and Applications*, 55:147–152, 1987.

[3] Åke Björck. *Numerical Methods for Least Squares Problems*. SIAM, Philadelphia, 1996.

[4] H. G. Bock. Recent advances in parameter identification techniques for ODE. In P. Deuflhard and E. Hairer, editors, *Numerical Treatment of Inverse Problems in Differential and Integral Equations*, pages 95–121. Birkhäuser, Boston, 1983.

[5] H. G. Bock. *Randwertproblemmethoden zur Parameteridentifizierung in Systemen nichtlinearer Differentialgleichungen*, volume 183 of *Bonner Mathematische Schriften*. Universität Bonn, Bonn, 1987.

[6] H. G. Bock, E. A. Kostina, and J. P. Schlöder. On the role of natural level functions to achieve global convergence for damped Newton methods. In M.J.D. Powell and S. Scholtes, editors, *System Modelling and Optimization. Methods, Theory and Applications*. Kluwer, 2000.

[7] M. Brookes. The Matrix Reference Manual. `http://www.ee.ic.ac.uk/hp/staff/dmb/matrix/intro.html`, 2011.

[8] C. G. Broyden, J. E. Dennis, Jr., and Jorge J. Moré. On the local and superlinear convergence of quasi-Newton methods. *Journal of the Institute of Mathematics and Its Applications*, 12(1):223–245, 1973.

[9] J. E. Dennis, Jr. and Robert B. Schnabel. *Numerical Methods for Unconstrained Optimization and Nonlinear Equations*. Prentice-Hall Series in Computational Mathematics, Cleve Moler, Advisor. 1983.

[10] P. Deuflhard. *Ein Newton-Verfahren bei fastsingulärer Funktionalmatrix zur Lösung von nichtlinearen Randwertaufgaben mit der Mehrzielmethode*. Dissertation in mathematics, Math. Institute University of Cologne, Dec. 1972.

[11] P. Deuflhard. *Newton Methods for Nonlinear Problems. Affine Invariance and Adaptive Algorithms*. Springer Series in Computational Mathematics. 2004.

[12] P. Deuflhard and G. Heindl. Affine Invariant Convergence theorems for Newton's Method and Extensions to related Methods. *SIAM Journal on Numerical Analysis*, 16(1):1–10, Feb. 1979.

[13] A. Griewank, D. Juedes, and J. Utke. ADOL-C, A Package for the Automatic Differentiation of Algorithms Written in C/C++. *ACM Transactions on Mathematical Software*, 22(2):131–167, 1996.

[14] A. Griewank and A. Walther. On Constrained Optimization by Adjoint based quasi-Newton Methods. *Optimization Methods and Software*, 17(5):869–889, 2002.

[15] Andreas Griewank. *Evaluating Derivatives, Principles and Techniques of Algorithmic Differentiation*. Number 19 in Frontiers in Applied Mathematics. SIAM, Philadelphia, PA, 2000.

[16] Herbert B. Keller. Newton's Method under Mild Differentiability Conditions. *Journal of Computer and System Sciences*, 4(1):15–28, 1970.

[17] A. Kiełbasiński and H. Schwetlick. *Numerische lineare Algebra. Eine computerorienierte Einführung*. Verlag Harri Deutsch, Thun-Frankfurt, 1988.

[18] Richard F. King. An improved pegasus method for root finding. *BIT Numerical Mathematics*, 13(4):423–427, December 1973.

[19] MATLAB. `http://www.mathworks.com/products/matlab/`.

[20] L. Miranian and M. Gu. Strong rank revealing LU factorizations. *Linear Algebra and its Applications*, 367:1–16, July 2003.

[21] J. Molenaar and P. W. Hemker. A multigrid approach for the solution of the 2D semiconductor equations. *IMPACT of Computing in Science and Engineering*, 2(3):219–243, 1990.

[22] Jorge J. Moré. The Levenberg-Marquardt algorithm: Implementation and theory. In G. A. Watson, editor, *Numerical Analysis*, volume 630 of *Lecture Notes in Mathematics*, pages 105–116. Springer Verlag, New York, 1978.

[23] Jorge J. Moré. A Collection of Nonlinear Model Problems. In Eugene L. Allgower and Kurt Georg, editors, *Computational Solution of Nonlinear Systems of Equations*, volume 26 of *Lectures in Applied Mathematics*, pages 723–762. AMS, 1990.

[24] Jorge J. Moré and Michel Y. Cosnard. Numerical Solution of Nonlinear Equations. *ACM Transactions on Mathematical Software*, 5(1):64–85, 1979.

[25] Jorge J. Moré, Burton S. Garbow, and Kenneth E. Hillstrom. Testing Unconstrained Optimization Software. *ACM Transactions on Mathematical Software*, 7(1):17–41, 1981.

[26] U. Nowak and L. Weimann. A Family of Newton Codes for Systems of Highly Nonlinear Equations. Technical Report TR 91-10, Zuse Institute Berlin (ZIB), 1991.

[27] J. M. Ortega and W. C. Rheinboldt. *Iterative Solution of Nonlinear Equations in Several Variables*. Academic Press, New York, NY, USA, 1970.

[28] Sebastian Schlenkrich. *Adjoint-based Quasi-Newton Methods for Nonlinear Equations*. Sierke Verlag, 2007.

[29] Sebastian Schlenkrich, Andrea Walther, and Andreas Griewank. Application of AD-based Quasi-Newton Methods to Stiff ODEs. In H. M. Bücker, G. Corliss, P. Hovland, U. Naumann, and B. Norris, editors, *Automatic Differentiation: Applications, Theory, and Implementations*, volume 50 of *Lecture Notes in Computational Science and Engineering*. Springer, 2006.

[30] P. Stange, A. Griewank, and M. Bollhöfer. On the Efficient Update of Rectangular LU Factorizations subject to Low Rank Modifications. *Electronic Transactions on Numerical Analysis*, 26:161–177, 2007.

[31] J. Stoer. On the Numerical Solution of Constrained Least-Squares Problems. *SIAM Journal on Numerical Analysis*, 8(2):382–411, 1971.

[32] J. Stoer and R. Bulirsch. *Introduction to Numerical Analysis*, volume 12 of *Texts in Applied Mathematics*. Springer, 3rd edition, 2002.