Università di Firenze, Università di Perugia, INdAM consorziate nel CIAFM

DOTTORATO DI RICERCA IN MATEMATICA, INFORMATICA, STATISTICA

CURRICULUM IN MATEMATICA
CICLO XXXIII

Sede amministrativa Università degli Studi di Firenze
Coordinatore Prof. Paolo Salani

Solving systems of nonlinear equations via spectral residual methods

Settore Scientifico Disciplinare: MAT/08, Analisi Numerica

Dottoranda
Cristina Sgattoni

Tutor
Prof.ssa Benedetta Morini
Dr.ssa Margherita Porcelli

Coordinatore
Prof. Paolo Salani

Anni 2017/2020

Acknowledgments

I would like to express my sincere gratitude to my supervisors Prof.ssa Benedetta Morini and Dr.ssa Margherita Porcelli for their assistance at every stage of this thesis. Their immense knowledge and plentiful experience have encouraged me in all the time of my academic research.

I would also like to thank the examiners Prof.ssa Valeria Ruggiero and Prof.ssa Daniela di Serafino for their insightful comments and suggestions that helped me to improve my work.

My heartfelt thanks go to my colleagues, they have been a constant source of strength and inspiration. Without their support, it would be very hard for me to complete my study.

Finally, I would like to acknowledge my family, my boyfriend and my friends. I can honestly say that it was their enormous understanding that made it possible for me to see this project through to the end.

Contents

List of Figures

Chapter 1

Introduction

In this thesis we address the numerical solution of systems of nonlinear equations via *spectral residual methods.* Our problem takes the form

$$F(x) = 0, \tag{1.1}$$

with $F : \mathbb{R}^n \to \mathbb{R}^n$ continuously differentiable. We focus on the square case where the number of equations equals the number of variables and we assume that problem (1.1) admits a solution. Spectral residual methods are iterative procedures, they use the residual vector F evaluated at the current iterate as search direction and a spectral steplength, i.e., a steplength that is related to the spectrum of the average matrices associated to the Jacobian matrix of F. Such procedures are widely studied and employed since they are derivative-free and low-cost per iteration.

This chapter is devoted to an introduction to the problem of interest and to an overview of the methods proposed in literature in recent years. We close the chapter summarizing the contents of the thesis.

1.1 Problem overview

Systems of nonlinear equations (1.1) arise in many applications and require finding one vector $x \in \mathbb{R}^n$ that satisfies the relationships specified by the residual function F. Examples of applications are the Karush-Kuhn-Tucker conditions related to a nonlinear programming problem, the discretization of partial differential equations such as heat conduction or Navier-Stokes equations and physical or economical constraints such as consistency principles, conservation laws, equilibrium conditions [49]. In addition, many other applications such as the Kalker's rolling contact model [45] or natural gas distribution models [41] require the solution of a sequence of suitable nonlinear systems.

The numerical solution of (1.1) has been intensively investigated and a variety of iterative procedures has been proposed. The combination of efficiency, measured in terms of execution time and computational cost, and robustness, that is the ability to solve the problem successfully, is fundamental. In our context, methods are considered robust if they are able to solve problems arising from a large number of different areas and if the convergence does not depend critically on the choice of the starting point. Methods with the latter property are denoted as globally convergent methods. It is worth noting that a possible approach to (1.1) consists in solving the nonlinear least-squares problem written as the sum of the squares of the equations in (1.1):

$$\min_{x\in\mathbb{R}^n} f(x) = \min_{x\in\mathbb{R}^n} \frac{1}{2}\|F(x)\|^2, \tag{1.2}$$

with $f : \mathbb{R}^n \to \mathbb{R}$ known as merit or objective function and $\|\cdot\|$ being the Euclidean norm. Nonlinear least-squares problems have been a productive area of study and there exist many software packages to solve them [14, 22, 32, 49]. Nevertheless, well known important differences between nonlinear systems and optimization induce to study adequate algorithms for solving (1.1) in its original form [14, 22, 49]. In nonlinear equations we expect all equations to be satisfied at the solution rather than just minimizing the sum of squares, i.e. any solution of (1.1) is a global minimum for (1.2) but the viceversa is not true. This means that a local minimum of f in (1.2) could provide a point that is not a solution to our problem (1.1).

Concerning the solution of the original formulation (1.1), a wide class of globally convergent methods is based on the Newton method combined with linesearch or trust-region approaches, see e.g., [14, 49]. The main drawback of these methods is that they require the solution of a linear system of equations at each iteration where the coefficient matrix is the Jacobian of F or an approximation of it by finite differences. Such calculation might be quite expensive either when the problem is of medium or large size or when a sequence consisting of a large number of nonlinear systems has to be solved. For this reason classes of algorithms that approximate the Jacobian, reducing the computational cost without losing robustness and overall efficiency, are of special interest. Quasi-Newton methods belong to this class and are particularly attractive when the Jacobian matrix of F is not available analytically or its computation is not relatively easy. They showed to be effective both in the solution of one single nonlinear system and in the solution of sequences of nonlinear systems such as those arising in applications where sequences are generated by iterative refinement of parameters, see e.g., [6, 14, 28, 33, 34, 41, 44, 58]. In the next section we will focus on the issues arising in the context of Quasi-Newton methods and we will introduce the class of methods studied in this thesis.

1.2 Numerical methods

The most common approach for the solution of problem (1.1) consists in the use of Newton-based methods, as mentioned in the previous section. This means that, letting x_k be the current iterate, the next iterate x_{k+1} is computed solving the linear system

$$J(x_k)(x_{k+1} - x_k) = -F(x_k), \tag{1.3}$$

where $J(x_k)$ is the $n \times n$ Jacobian matrix of F at iteration k. We notice that these methods may become computationally expensive since both the computation of matrix J and the solution of a linear system are required at each iteration.

As for the solution of (1.3), direct methods such as Gaussian elimination may be too expensive if the system is medium or large size and the Jacobian matrix is either not structured or no sparse. Moreover, computing the solution of (1.3) at each iteration with a high accuracy may be not necessary when the current iterate x_k is far from the solution. Therefore, for large dimension problems, a possible approach for (1.1) is using Inexact Newton methods where the linear system (1.3) is solved inexactly by means of iterative solvers [12,17,42,55]. The inexactness comes from the fact that the iterative procedure for (1.3) is stopped prematurely, and consequently the linear system is solved approximately at a low computational cost per iteration. Inexact Newton methods are also matrix-free, i.e. they access the coefficient matrix $J(x_k)$ only evaluating matrix-vector products and avoid forming and storing the whole matrix $J(x_k)$. This class of methods is particularly convenient when the matrices are sparse but their efficiency generally depends on using a proper preconditioner for $J(x_k)$ and this calls for information on $J(x_k)$.

Quasi-Newton methods are adopted as an alternative approach replacing the matrix J with an approximation of it. The k-th iteration matrix, denoted as B_k, can be formed via least-change secant update strategies and may not involve derivatives at all [14,34,40]. In details, let us consider the following affine model for F around x_k

$$M_k(x) = F(x_k) + B_k(x - x_k), \tag{1.4}$$

satisfying $M_k(x_k) = F(x_k)$ for any matrix $B_k \in \mathbb{R}^{n\times n}$ and let x_{k+1} be such that $M_k(x_{k+1}) = 0$. We observe that this equation reduces to the Newton's equation (1.3) when $B_k = J(x_k)$. If $J(x_k)$ is not available or too expensive to compute, let us consider the secant equation stating that $M_k(x_{k-1}) = F(x_{k-1})$, that is

$$B_k(x_k - x_{k-1}) = F(x_k) - F(x_{k-1}). \tag{1.5}$$

If dimension n is larger than 1 then matrix B_k is not uniquely determined by (1.5) since there is an $n(n-1)$-dimensional affine subspace of matrices obeying such equation. The construction of a successful secant approximation consists in the selection of some matrices among all these possibilities. The choice of B_k should either retain as much information as possible from $J(x_k)$ and/or allow for a low cost solution of the linear system. A possible strategy could be to require the model (1.4) to interpolate $F(x)$ at other past points, but this leads to a poorly posed numerical problem and is not successful in practice [14]. The approach that leads to a successful secant approximation is the so called Broyden's update. It is based on the fact that we have no information either on the Jacobian or on the model (1.5) and its aim consists in preserving as much as possible of what is already available. Therefore, matrix B_k is chosen to minimize the change in the affine model. In details, it is proved that the Broyden's update represents the minimum change to B_{k-1} consistent with equation (1.5), measuring the change $B_k - B_{k-1}$ in the Frobenius norm [14, Lemma 8.1.1]. It turns out that B_k is not an approximation from scratch but it is a low rank update of B_{k-1}. As a consequence, the solution of the system $B_k(x_{k+1} - x_k) = -F(x_k)$ for x_{k+1} can take advantage of the availability of the factorization of a matrix at the previous iteration, e.g., if $B_{k-1}(x_k - x_{k-1}) = -F(x_{k-1})$

was solved for x_k using the QR factorization of B_{k-1} [48], such factorization can be updated at a low computational cost to get the QR factorization of B_k [14].

Many further successful updating techniques have been proposed, e.g., in the Inverse Column Update [43, 48] a column of the inverse of B_k^{-1} is updated at each iteration enforcing the secant equation (1.5). In so doing, the computation of the Quasi-Newton step $x_{k+1} - x_k$ only requires the product between B_k^{-1} and $F(x_k)$ avoiding the solution of a linear system. A further and particular case is given by the class of methods studied in this work where the Jacobian is approximated using a diagonal matrix. Summarizing, in Quasi-Newton methods the computational cost for building B_k is considerably lower than the cost for computing $J(x_k)$ and in many implementations the cost for solving the linear system $B_k(x_{k+1} - x_k) = -F(x_k)$ is low as previously described.

In this thesis we consider spectral residual methods which belong to the class of Quasi-Newton procedures. They are an extension of spectral gradient methods for large-scale optimization problems to systems of nonlinear equations. Spectral gradient methods, introduced by Barzilai and Borwein in [2], are low-cost schemes for minimizing a smooth function $f : \mathbb{R}^n \to \mathbb{R}$ and belong to the class of steepest descent methods, i.e., first-order iterative optimization algorithms which move at each iteration along $-\nabla f$ at the current iterate. Barzilai and Borwein showed in [2] that a suitable choice of the steplength greatly speeds up the convergence of the classical steepest descent method even if it does not guarantee descent in the objective function at each iteration. Spectral residual methods were first introduced by La Cruz and Raydan in [33] and starting from the proposal by La Cruz, Martinez and Raydan in [34] consist of iterative procedures for solving (1.1) without the use of derivatives. They use matrices B_k which are multiples of the identity matrix, i.e. $B_k = \beta_k^{-1} I$, with β_k being a nonzero steplength inspired by the Barzilai and Borwein method for unconstrained minimization problems [2]. Imposing condition (1.5) two steplengths $\beta_{k,1}$ and $\beta_{k,2}$ are derived as least-squares solutions of the following problems:

$$\beta_{k,1} = \operatorname*{argmin}_{\beta} \|\beta^{-1} p_{k-1} - y_{k-1}\|^2 = \frac{p_{k-1}^T p_{k-1}}{p_{k-1}^T y_{k-1}}, \tag{1.6}$$

$$\beta_{k,2} = \operatorname*{argmin}_{\beta} \|p_{k-1} - \beta y_{k-1}\|^2 = \frac{p_{k-1}^T y_{k-1}}{y_{k-1}^T y_{k-1}}, \tag{1.7}$$

where $p_{k-1} = x_k - x_{k-1}$ and $y_{k-1} = F(x_k) - F(x_{k-1})$.

Spectral residual methods have received a large attention since iterations are cheap and matrix-free, see e.g. [28, 33–35, 41, 48, 58]. In order to preserve robustness, such methods are combined with suitable globalization strategies that control the value of f in (1.2) at each iteration and use both $-\beta_k F(x_k)$ and $\beta_k F(x_k)$ as trial searches in a systematic way. In fact if $\nabla f(x_k)^T F(x_k) \neq 0$ then one of the two directions is a descent direction for f. The linesearch techniques adopted are tipically nonmonotone i.e., $\|F(x_k)\|$ is not monotonically decreasing [21, 36]. In the seminal paper [33] by La Cruz and Raydan a variant of the nonmonotone linesearch of Grippo, Lampariello and Lucidi [27] is used but such strategy requires the gradient of f and its computation is as costly as the computation of J being $\nabla f(x) = J(x)^T F(x)$. Since spectral residual methods do not require $J(x)$, it is appropriate to use a nonmonotone linesearch that does not involve derivatives; the first proposal was made in [34] by La Cruz, Martinez and Raydan and was based on derivative-free linesearch strategies for nonlinear systems.

Starting from an early contribution by Griewank [26], derivative-free linesearches for problem (1.1) were defined. Given x_k, let s_k be the trial step and suppose that either $s_k = -\beta_k F(x_k)$ or $s_k = \beta_k F(x_k)$ and that x_{k+1} takes the form $x_{k+1} = x_k + \gamma s_k$ with $\gamma \in (0, 1]$ chosen so that one of the nonmonotone linesearch conditions is met. Li and Fukushima [36] presented the derivative-free linesearch

$$\|F(x_k + \gamma s_k)\| \leq (1 + \eta_k)\|F(x_k)\| - \rho\gamma^2\|s_k\|^2, \tag{1.8}$$

with $\rho \in (0, 1)$ and η_k being a positive scalar such that $\{\eta_k\}$ satisfies

$$\sum_{k=0}^{\infty} \eta_k < \eta < \infty. \tag{1.9}$$

Note that (1.8) avoids the necessity of descent directions to guarantee that each iteration is well defined. By virtue of the continuity of F, condition (1.8) holds for all γ sufficiently small and it is called an *approximate norm descent* linesearch since it implies

$$\|F(x_k + \gamma s_k)\| \leq (1 + \eta_k)\|F(x_k)\|, \tag{1.10}$$

with $\eta_k \to 0$ as $k \to \infty$.

La Cruz, Martinez and Raydan [34] proposed a combination and extension of the Grippo, Lampariello and Lucidi linesearch and of the Li and Fukushima linesearch in order to produce a robust nonmonotone linesearch that takes into account the advantages of both schemes; it has the form

$$\|F(x_k + \gamma s_k)\| \leq \max_{0\leq j\leq \min\{k,M\}} \|F(x_{k-j})\| + \eta_k - \rho\gamma^2\|F(x_k)\|, \tag{1.11}$$

with M nonnegative integer, ρ and $\{\eta_k\}$ as in the Li and Fukushima proposal. The first term on the right-hand side of (1.11) produces the nonmonotone behaviour of the norm of F, the second term guarantees that the strategy is well defined, and the third term is fundamental for proving global convergence. Condition (1.11) is also employed in [28] with $\eta_k = 0$ for all k and combined with a nonmonotone watchdog rule. An alternative proposal was made by Birgin, Krejic and Martinez [3] formulating the following linesearch:

$$\|F(x_k + \gamma s_k)\| \leq (1 - \rho\gamma)\|F(x_k)\| + \eta_k. \tag{1.12}$$

Moreover, in [35] the following acceptance condition inspired by [50] was introduced by La Cruz:

$$\|F(x_k + \gamma s_k)\|^2 \leq \|F(x_k)\|^2 + \eta_k - \rho\gamma^2\|s_k\|^2. \tag{1.13}$$

Finally, in [41,48] a new linesearch strategy based on a nonmonotone approximate norm descent property of the merit function (1.10) was adopted; such a strategy will be introduced and discussed in details in the next chapter.

1.3 Contents of the thesis

Similarly to the Barzilai and Borwein method for unconstrained optimization, spectral residual methods for (1.1) generate a nonmonotone sequence $\{\|F(x_k)\|\}$ and their effectiveness heavily relies on the steplengths β_k used.

It is well known that the performance of the Barzilai and Borwein method does not depend on the decrease of the objective function at each iteration but relies on the relationship between the steplengths used and the eigenvalues of the average Hessian matrix of the objective function [4, 19, 52]. Based on such feature, several strategies for steplength selection have been proposed to enhance the performance of the method, see e.g., [9–11, 15, 19, 20]. On the other hand, to our knowledge, an analogous study of the relationship between the steplengths originated by spectral residual methods and the eigenvalues of the average Jacobian matrix of F has not been carried out, and the impact of the choice of the steplenghts on the convergence history has not been investigated in details.

The first aim of this thesis is to analyze the properties of the spectral residual steplengths $\beta_{k,1}$, $\beta_{k,2}$ in (1.6) and (1.7) and study how they affect the performance of the methods. This aim is addressed both from a theoretical and experimental point of view. The main contributions of this work in this direction are: the theoretical analysis of the steplengths proposed in the literature and of their impact on the norm of F also with respect to the nonmonotone behaviour imposed by globalization strategies; the analysis of the performance of spectral methods with various rules for updating the steplengths. Rules based on adaptive strategies that suitably combine small and large steplengths result by far more effective than rules based on static choices of β_k and, inspired by the steplength rules proposed in the literature for unconstrained minimization problems, we propose and extensively test adaptive steplength strategies. Numerical experience is conducted on sequences of nonlinear systems arising from rolling contact models which play a central role in many important applications, such as rolling bearings and wheel-rail interaction [30, 31]. Solving these models gives rise to sequences which consist of a large number of medium-size nonlinear systems and represent a relevant benchmark test set for the purpose of this thesis. A first set of experiments was conducted using the globally convergent scheme proposed in [48] and later denoted as SRAND1, Spectral Residual Approximate Norm Descent method, version 1.

The second purpose of this thesis is to propose a variant of the derivative-free spectral residual method SRAND1 and obtain a scheme globally convergent under more general conditions. In [48] the sequence generated by SRAND1 was proved to be convergent under mild standard assumptions; moreover, sufficient conditions were provided to ensure that a limit point x^* of the generated sequence $\{x_k\}$ is also a solution of (1.1). These conditions relayed on the steplength $\beta_{k,1}$ and held for specific classes of problems. For example, $F(x^*) = 0$ is guaranteed in the case where $J(x^*)$ has positive (negative) definite symmetric part and suitably bounded condition number and in the case where $J(x^*)$ is strongly diagonal dominant with diagonal entries of constant sign. Inspired by [34], we propose a new linesearch strategy, which allows to obtain a more general and nontrivial convergence result and does not rely on the specific choice of β_k. The resulting method is denoted as SRAND2, Spectral Residual Approximate Norm Descent method, version 2. We prove that at every limit point x^* of the sequence $\{x_k\}$ generated by SRAND2, either $F(x^*) = 0$ or the gradient of the merit function f in (1.2) is orthogonal to the residual F:

$$\nabla f(x^*)^T F(x^*) = F(x^*)^T J(x^*) F(x^*) = 0. \tag{1.14}$$

Clearly this result gives $F(x^*) = 0$ as long as $F(x^*) \neq 0$ is not orthogonal to $J(x^*)^T F(x^*)$, and it is not related to a specific class of nonlinear systems. We further show that the improvement with respect to SRAND1 is not only theoretical; the performed numerical experiments show that the new linesearch has some positive impact also on the practical ability in solving nonlinear systems. Numerical experiments are conducted both on the previously discussed problems arising in rolling contact models and on a set of problems commonly used for testing solvers for nonlinear systems varying the updating rules for β_k.

Our original contribution in the development and analysis of spectral residual methods for solving problem (1.1) is contained in the works [45, 51].

The thesis is organized as follows. Chapter 2 is divided in three parts. First of all we introduce preliminaries on spectral residual methods; then in the second section we provide a theoretical analysis of the steplengths; finally, in the third section we present and study the algorithms SRAND1 and SRAND2. The experimental part is developed in Chapter 3 where we provide several strategies for selecting the steplength, introduce our test sets and discuss the numerical results obtained. Some conclusions and research perspectives are presented in Chapter 4. In Appendix A we detail the rolling contact model from which our first problem set derives, its discretization and the algorithm for its solution. Finally, complete results obtained with SRAND1 and SRAND2 are reported in Appendix B.

1.4 Notations

Throughout the thesis we use the following notation.

Unless explicitly stated, the symbol $\|\cdot\|$ denotes the Euclidean norm.

I denotes the identity matrix.

J denotes the Jacobian matrix of F.

Given a square matrix A, we let $A_S = \frac{1}{2}(A + A^T)$ be the symmetric part of A.

Given a symmetric matrix M, $\{\lambda_i(M)\}_{i=1}^n$ denotes the set of eigenvalues of M, $\lambda_{\min}(M)$ and $\lambda_{\max}(M)$ denote the minimum and maximum eigenvalue of M respectively, and $\{v_i\}_{i=1}^n$ denotes a set of associated orthonormal eigenvectors. Further, given a nonzero vector p, we let $q(M,p) = \dfrac{p^T M p}{p^T p}$ be the Rayleigh quotient.

Given a sequence of vectors $\{x_k\}$, for any function f we occasionally let $f_k = f(x_k)$.

Chapter 2

Spectral residual methods: stepsize selection and global convergence

This chapter contains the theoretical contribution of the thesis. In particular, in the first section we introduce the basic concepts and notation for spectral residual methods. In the second section we provide a theoretical analysis of the steplengths (1.6) and (1.7) including their impact on the behaviour of the norm of F and on a general scheme for nonmonotone linesearch. In the third section we present two linesearch strategies, their use in conjunction with spectral residual methods and discuss their convergence properties.

2.1 Preliminaries

In the seminal paper [2] Barzilai and Borwein proposed a gradient method for the unconstrained minimization

$$\min_{x\in\mathbb{R}^n} f(x), \tag{2.1}$$

where $f : \mathbb{R}^n \to \mathbb{R}$ is a given differentiable function. Given an initial guess $x_0 \in \mathbb{R}^n$, the Barzilai-Borwein (BB) iteration is defined by

$$x_{k+1} = x_k - \alpha_k \nabla f_k, \tag{2.2}$$

where α_k is a positive steplength inspired by Quasi-Newton methods for unconstrained optimization [14]. In Quasi-Newton methods, the step $p_k = x_{k+1} - x_k$ solves the linear system

$$B_k p_k = -\nabla f_k, \tag{2.3}$$

and, given $B_0 \in \mathbb{R}^{n\times n}$ as an initial data, $B_k \in \mathbb{R}^{n\times n}$, $k \geq 1$, satisfies the secant equation, i.e.,

$$B_k p_{k-1} = z_{k-1}, \quad \text{with} \quad p_{k-1} = x_k - x_{k-1}, \quad z_{k-1} = \nabla f_k - \nabla f_{k-1}. \tag{2.4}$$

Letting $B_k = \alpha^{-1} I$ and imposing condition (2.4), Barzilai and Borwein derived two steplengths which are the least-square solutions of the following problems:

$$\alpha_{k,1} = \operatorname*{argmin}_{\alpha} \|\alpha^{-1} p_{k-1} - z_{k-1}\|^2 = \frac{p_{k-1}^T p_{k-1}}{p_{k-1}^T z_{k-1}}, \tag{2.5}$$

$$\alpha_{k,2} = \operatorname*{argmin}_{\alpha} \|p_{k-1} - \alpha z_{k-1}\|^2 = \frac{p_{k-1}^T z_{k-1}}{z_{k-1}^T z_{k-1}}. \tag{2.6}$$

The second least-squares formulation is obtained from the first by symmetry. The final steplength α_k computed from (2.5) and (2.6) is then adjusted in order to be positive, bounded away from zero and not too large, i.e., $\alpha_k \in [\alpha_{\min}, \alpha_{\max}]$ for some positive $\alpha_{\min}$, $\alpha_{\max}$; in fact, one of the two scalars $\alpha_{k,1}$, $\alpha_{k,2}$ is used and the thresholds $\alpha_{\min}$, $\alpha_{\max}$ are applied to it, see e.g., [4, 15, 19].

Choosing $B_k = \alpha^{-1} I$ yields a low-cost iteration while the use of the steplengths $\alpha_{k,1}$, $\alpha_{k,2}$ yields a considerable improvement in the performance with respect to the classical steepest descent method [2, 19]. The BB method is commonly employed in the solution of large unconstrained optimization problems (2.1) and the behaviour of the sequence $\{f(x_k)\}$ is typically nonmonotone, possibly severely nonmonotone, in both the cases of quadratic and general nonlinear functions f [19, 23, 54]. The performance of the BB method depends on the relationship between the steplength α_k and the eigenvalues of the average Hessian matrix $\int_0^1 \nabla^2 f(x_{k-1} + t\, p_{k-1})\, dt$; hence this approach is also denoted as *spectral method* and an extensive investigation on steplength's selection has been carried on [9–11, 15, 19, 20].

The extension of this approach to the solution of nonlinear systems of equations (1.1) was firstly proposed by La Cruz and Raydan in [33]. Here we summarize such a proposal and the issues that were inherited by subsequent procedures falling into such framework and designed for both general nonlinear systems [28, 33–35, 41, 48, 58] and for monotone nonlinear systems* [1, 37, 38, 46, 57, 61]. Instead of applying the spectral method to the merit function

$$f(x) = \|F(x)\|^2, \tag{2.7}$$

the BB approach is specialized to the Newton equation yielding the so-called *spectral residual method.* Thus, let p_- satisfy the linear system

$$B_k p_- = -F_k, \tag{2.8}$$

and let $B_k = \beta^{-1} I$ satisfy the secant equation

$$B_k p_{k-1} = y_{k-1}, \quad \text{with} \quad p_{k-1} = x_k - x_{k-1}, \quad y_{k-1} = F_k - F_{k-1}.$$

*Nonlinear systems of the form (1.1) are *monotone* if $F : \mathbb{R}^n \to \mathbb{R}^n$ is monotone, i.e. $(F(x) - F(y))^T (x - y) \geq 0$ for any $x, y \in \mathbb{R}^n$, see e.g., [18].

Reasoning as in BB method, two steplengths are derived:

$$\beta_{k,1} = \frac{p_{k-1}^T p_{k-1}}{p_{k-1}^T y_{k-1}}, \tag{2.9}$$

$$\beta_{k,2} = \frac{p_{k-1}^T y_{k-1}}{y_{k-1}^T y_{k-1}}. \tag{2.10}$$

These scalars may be positive, negative or even null; moreover $\beta_{k,1}$ is not well defined if $p_{k-1}^T y_{k-1} = 0$ and $\beta_{k,2}$ is not well defined if $y_{k-1} = 0$. In practice, the steplength β_k is chosen equal either to $\beta_{k,1}$ or to $\beta_{k,2}$ as long as it results to be bounded away from zero and $|\beta_k|$ is not too large, i.e., $|\beta_k| \in [\beta_{\min}, \beta_{\max}]$ for some positive $\beta_{\min}$, $\beta_{\max}$. The step resulting from (2.8) turns out to be of the form $p_- = -\beta_k F_k$. But, once β_k is fixed, the kth iteration of the spectral residual method employs the residual directions $\pm F_k$ in a systematic way and tests both the steps

$$p_- = -\beta_k F_k \quad \text{and} \quad p_+ = +\beta_k F_k,$$

for acceptance using a suitable linesearch strategy. The use of both directions $\pm F_k$ is motivated by the fact that, contrary to $(-\alpha_k \nabla f_k)$, $\alpha_k > 0$, in (2.2), $(-\beta_k F_k)$ is not necessarily a descent direction for (2.7) at x_k; the value $\nabla f_k^T(-\beta_k F_k) = -2\beta_k F_k^T J_k F_k$ could be positive, negative or null. On the other hand, if $F_k^T J_k F_k \neq 0$, trivially either $(-\beta_k F_k)$ or $\beta_k F_k$ is a descent direction for f.

Analogously to the spectral method, the spectral residual method is characterized by a nonmonotone behaviour of $\{\|F_k\|\}$ and is implemented using nonmonotone linesearch strategies. The adaptation of the spectral method to nonlinear systems is low-cost per iteration since the computation of $\beta_{k,1}$ and $\beta_{k,2}$ is inexpensive and the memory storage is low, and turned out to be effective in the solution of medium and large nonlinear systems, see e.g., [28, 33–35, 48, 58].

Unlike the context of BB method for unconstrained optimization, to our knowledge a systematic analysis of the stepsizes $\beta_{k,1}$ and $\beta_{k,2}$ in the context of the solution of nonlinear systems and their impact on convergence history has not been carried out. The steplength $\beta_{k,1}$ has been used in most of the works on this subject [33–35, 41, 48]. On the other hand, in [28] it was observed experimentally that alternating $\beta_{k,1}$ and $\beta_{k,2}$ along iterations was beneficial for the performance and in [58] it was observed experimentally that using $\beta_{k,2}$ performed better in terms of robustness with respect to using $\beta_{k,1}$.

In the next two subsections we will analyze the two steplengths $\beta_{k,1}$ and $\beta_{k,2}$ and provide: their expression in terms of the spectrum of average matrices associated to the Jacobian matrix of F; their mutual relationship; their impact on the behaviour of $\|F_k\|$ and on a standard nonmonotone linesearch.

The matrices involved in our analysis are the following. Given a square matrix A, we let $A_S = \frac{1}{2}(A + A^T)$ be the symmetric part of A, G_{k-1} be the average matrix associated to the Jacobian J of F :

$$G_{k-1} \stackrel{\text{def}}{=} \int_0^1 J(x_{k-1} + t\, p_{k-1})\, dt, \tag{2.11}$$

and $(G_S)_{k-1}$ be the average matrix associated to the symmetric part J_S of J :

$$(G_S)_{k-1} \stackrel{\text{def}}{=} \int_0^1 J_S(x_{k-1} + t\, p_{k-1})\, dt. \tag{2.12}$$

Moreover, given a symmetric matrix M and a nonzero vector p, the Rayleigh quotient $q(M, p)$ introduced in Section 1.4 satisfies the following property [24, Theorem 8.1-2]

$$\lambda_{\min}(M) \leq q(M, p) \leq \lambda_{\max}(M). \tag{2.13}$$

2.2 Stepsize selection

2.2.1 Analysis of the steplengths $\beta_{k,1}$ and $\beta_{k,2}$

In this subsection we analyze the stepsizes $\beta_{k,1}$ and $\beta_{k,2}$ given in (2.9) and (2.10) making the following assumptions.

Assumption 2.2.1 *The scalars $\beta_{k,1}$ and $\beta_{k,2}$ are well defined and nonzero.*

Assumption 2.2.2 *Given x and p, F is continuously differentiable in an open convex set $D \subset \mathbb{R}^n$ containing $x + tp$ with $t \in [0, 1]$.*

We note that Assumption 2.2.1 holds whenever $p_{k-1}^T y_{k-1} \neq 0$.

In the following lemma we analyze the mutual relationship between the stepsizes $\beta_{k,1}$ and $\beta_{k,2}$ and give their characterization in terms of suitable Rayleigh quotients for the average matrices in (2.11) and (2.12). We will use repeatedly the property

$$p^T A p = p^T A_S p, \tag{2.14}$$

which holds for any square matrices A, $A_S = \frac{1}{2}(A + A^T)$, and any vector p of suitable dimension.

Lemma 2.2.3 *Let Assumption 2.2.1 hold and Assumption 2.2.2 hold with $x = x_{k-1}$, $p = p_{k-1}$. The steplengths $\beta_{k,1}$, $\beta_{k,2}$ are such that:*

P1) *they have the same sign and $|\beta_{k,2}| \leq |\beta_{k,1}|$;*

P2) *either it holds $\beta_{k,1} \leq \beta_{k,2} < 0$ or $0 < \beta_{k,2} \leq \beta_{k,1}$;*

P3) *they take the form*

$$\beta_{k,1} = \frac{1}{q\big((G_S)_{k-1}, p_{k-1}\big)}, \tag{2.15}$$

and

$$\beta_{k,2} = \frac{q\big((G_S)_{k-1}, p_{k-1}\big)}{q(G_{k-1}^T G_{k-1}, p_{k-1})}, \tag{2.16}$$

with $q(\cdot, \cdot)$ being the Rayleigh quotient, G_{k-1} and $(G_S)_{k-1}$ being the matrices in (2.11) and (2.12), respectively.

Proof. By (2.9) and (2.10), we can write

$$\begin{aligned}\beta_{k,2} &= \frac{p_{k-1}^T p_{k-1}}{p_{k-1}^T y_{k-1}} \frac{(p_{k-1}^T y_{k-1})^2}{(y_{k-1}^T y_{k-1})(p_{k-1}^T p_{k-1})} \\ &= \beta_{k,1} \frac{\|p_{k-1}\|^2 \|y_{k-1}\|^2 cos^2 \varphi_{k-1}}{\|p_{k-1}\|^2 \|y_{k-1}\|^2} \\ &= \beta_{k,1} \cos^2 \varphi_{k-1}, \end{aligned} \tag{2.17}$$

where φ_{k-1} is the angle between p_{k-1} and y_{k-1}, and P1) follows.

Property P2) follows as well since $\beta_{k,2} \neq 0$ by Assumption 2.2.1.

As for property P3), by the Mean Value Theorem [14, Lemma 4.1.9] and (2.11) we have

$$y_{k-1} = F_k - F_{k-1} = \int_0^1 J(x_{k-1} + t p_{k-1}) p_{k-1} \, dt = G_{k-1} p_{k-1}.$$

Then using (2.14) and the definition of the Rayleigh quotient, $\beta_{k,1}$ takes the form

$$\beta_{k,1} = \frac{p_{k-1}^T p_{k-1}}{p_{k-1}^T G_{k-1} p_{k-1}} = \frac{1}{q\big((G_S)_{k-1}, p_{k-1}\big)},$$

while $\beta_{k,2}$ takes the form

$$\beta_{k,2} = \frac{p_{k-1}^T (G)_{k-1} p_{k-1}}{p_{k-1}^T (G_{k-1}^T G_{k-1}) p_{k-1}} \frac{p_{k-1}^T p_{k-1}}{p_{k-1}^T p_{k-1}} = \frac{q\big((G_S)_{k-1}, p_{k-1}\big)}{q(G_{k-1}^T G_{k-1}, p_{k-1})}.$$

□

The above characterization P3) allows to derive bounds on the stepsizes $\beta_{k,1}$ and $\beta_{k,2}$ diversifying cases according to the spectral properties of the Jacobian matrix and the average matrices in (2.11) and (2.12). The relationship between $\beta_{k,1}$ and the spectral information of the symmetric part of average matrix (2.11) was observed in [33, 34, 48] but the following results are not contained in such references.

Lemma 2.2.4 *Let Assumption 2.2.1 hold and Assumption 2.2.2 hold with $x = x_{k-1}$, $p = p_{k-1}$. Then, the steplengths $\beta_{k,1}$ and $\beta_{k,2}$ are such that:*

(i) *if the Jacobian J is symmetric and positive definite on the line segment in between x_{k-1} and $x_{k-1} + p_{k-1}$ then $\beta_{k,1}$ and $\beta_{k,2}$ are positive and*

$$\frac{1}{\lambda_{\max}(G_{k-1})} \leq \beta_{k,2} \leq \beta_{k,1} \leq \frac{1}{\lambda_{\min}(G_{k-1})}; \tag{2.18}$$

(ii) *if $(G_S)_{k-1}$ in (2.12) is positive definite, then $\beta_{k,1}$ and $\beta_{k,2}$ are positive and*

$$\max\left\{\frac{1}{\lambda_{\max}\big((G_S)_{k-1}\big)}, \beta_{k,2}\right\} \leq \beta_{k,1} \leq \frac{1}{\lambda_{\min}\big((G_S)_{k-1}\big)}, \tag{2.19}$$

$$\frac{\lambda_{\min}\big((G_S)_{k-1}\big)}{\lambda_{\max}(G_{k-1}^T G_{k-1})} \leq \beta_{k,2} \leq \min\left\{\frac{\lambda_{\max}\big((G_S)_{k-1}\big)}{\lambda_{\min}(G_{k-1}^T G_{k-1})}, \beta_{k,1}\right\}; \tag{2.20}$$

(iii) *if* $(G_S)_{k-1}$ *in (2.12) is indefinite and* G_{k-1} *in* (2.11) *is nonsingular, then*

(iii.1) $\beta_{k,1}$ *satisfies either*

$$\beta_{k,1} \leq \min\left\{\frac{1}{\lambda_{\min}\left((G_S)_{k-1}\right)}, \beta_{k,2}\right\} \quad \text{or} \quad \beta_{k,1} \geq \max\left\{\frac{1}{\lambda_{\max}\left((G_S)_{k-1}\right)}, \beta_{k,2}\right\}; \tag{2.21}$$

(iii.2) $\beta_{k,2}$ *satisfies either*

$$0 < \beta_{k,2} \leq \min\left\{\frac{\lambda_{\max}\big((G_S)_{k-1}\big)}{\lambda_{\min}(G_{k-1}^T G_{k-1})}, \beta_{k,1}\right\}, \tag{2.22}$$

or

$$\max\left\{\frac{\lambda_{\min}\big((G_S)_{k-1},\big)}{\lambda_{\max}(G_{k-1}^T G_{k-1})}, \beta_{k,1}\right\} \leq \beta_{k,2} < 0. \tag{2.23}$$

Proof. Consider properties P1), P2) and P3) from Lemma 2.2.3.

(i) Steplengths $\beta_{k,1}$ and $\beta_{k,2}$ are positive due to (2.15), (2.16). The rightmost inequality of (2.18) follows from (2.15) and (2.13). The remaining part of (2.18) is proved observing that (2.16) yields

$$\beta_{k,2} = \frac{p_{k-1}^T G_{k-1}^{1/2} G_{k-1}^{1/2} p_{k-1}}{p_{k-1}^T G_{k-1}^{1/2} G_{k-1} G_{k-1}^{1/2} p_{k-1}} = \frac{1}{q(G_{k-1}, G_{k-1}^{1/2} p_{k-1})}, \tag{2.24}$$

and using P2) and (2.13).

(ii) Using (2.15),(2.13) and P2) we get positivity of $\beta_{k,1}$ and (2.19). Consequently, $\beta_{k,2}$ is positive by property P1), and bounds (2.20) can be derived using (2.16), (2.13) and item P2) of Lemma 2.2.3.

(iii) If $(G_S)_{k-1}$ is indefinite then its extreme eigenvalues have opposite sign, i.e., $\lambda_{\min}\big((G_S)_{k-1}\big) < 0$ and $\lambda_{\max}\big((G_S)_{k-1}\big) > 0$. Hence, (2.15), (2.13) and P2) give (2.21). Moreover, since $G_{k-1}^T G_{k-1}$ is symmetric and positive definite, we can use, as before, P1) and (2.13) and get (2.22) and (2.23).

□

Lemma 2.2.4 easily extends to the case where matrices are negative definite.

Item (i) in Lemma 2.2.4 includes the case where F is *strictly monotone*, i.e., $(F(x) - F(y))^T(x-y) > 0$ for any $x, y \in \mathbb{R}^n$ with $x \neq y$, see e.g. [18]. In fact, if the Jacobian is positive definite in $\mathbb{R}^n$ then F is strictly monotone in $\mathbb{R}^n$ [18, Preposition 2.3.2].

2.2.2 On the impact of the steplength β_k on $\|F_{k+1}\|$, case J symmetric

In this subsection we investigate how the choice of the steplength β_k may affect $\|F_{k+1}\|$ in a spectral residual method when the Jacobian J is symmetric. Results are first derived using a generic β_k and discussed thereafter with respect to the choice of either $\beta_{k,1}$ or $\beta_{k,2}$.

Next result analyzes the residual vector F_{k+1} componentwise. It heavily relies on the existence of a set of orthonormal eigenvectors for the average matrix G_k.

Lemma 2.2.5 *Suppose that Assumption 2.2.2 holds with $x = x_k$ and $p = p_k$ and that the Jacobian J is symmetric. Let $p_k = p_- = -\beta_k F_k \neq 0$, $x_{k+1} = x_k + p_k$, $\big\{\lambda_i\big(G_k\big)\big\}_{i=1}^n$ be the eigenvalues of matrix G_k in (2.11) and $\{v_i\}_{i=1}^n$ be a set of associated orthonormal eigenvectors. Let F_k and F_{k+1} be expressed as*

$$F_k = \sum_{i=1}^n \mu_k^i v_i, \qquad F_{k+1} = \sum_{i=1}^n \mu_{k+1}^i v_i,$$

where μ_k^i, μ_{k+1}^i, $i = 1, \ldots, n$, are scalars. Then

$$F_{k+1} = (I - \beta_k G_k) F_k, \tag{2.25}$$
$$\mu_{k+1}^i = \mu_k^i \big(1 - \beta_k \lambda_i(G_k)\big), \qquad i = 1, \ldots, n. \tag{2.26}$$

Moreover, it holds:

(a) *if $\beta_k \lambda_i(G_k) = 1$, then $\mu_{k+1}^i = 0$;*

(b) *if $0 < \beta_k \lambda_i(G_k) < 2$, then $|\mu_{k+1}^i| < |\mu_k^i|$; otherwise $|\mu_{k+1}^i| \geq |\mu_k^i|$.*

Proof. The Mean Value Theorem [14, Lemma 4.1.9] gives

$$F_{k+1} \;=\; F_k + \int_0^1 J(x_k + t p_k) p_k \, dt,$$

and $p_k = -\beta_k F_k$ and (2.11) yield (2.25). Moreover, since $\{v_i\}_{i=1}^n$ are orthonormal we have for $i = 1, \ldots, n$

$$\begin{aligned} \mu_{k+1}^i &= (v_i)^T F_{k+1} \\ &= (v_i)^T (I - \beta_k G_k) F_k \\ &= \mu_k^i \big(1 - \beta_k \lambda_i(G_k)\big), \end{aligned}$$

i.e., equation (2.26). Consequently, Item (a) follows trivially; Item (b) follows noting that $\big|1 - \beta_k \lambda_i(G_k)\big| < 1$ if and only if $0 < \beta_k \lambda_i(G_k) < 2$.

□

Lemma 2.2.5 trivially extends to the case where $p_k = p_+ = \beta_k F_k$.

If the nonlinear system (1.1) represents the first-order optimality condition of the optimization problem (2.1) where $f(x) = \frac{1}{2}x^T Ax - b^T x$ is quadratic and A is symmetric and positive definite, then the previous lemma reduces to well known results on the behaviour of the gradient method in terms of the spectrum of the Hessian matrix A, see [52]. In fact, we get $F(x) = \nabla f(x) = Ax - b = 0$ and its Jacobian is constant $J(x) = A$, $\forall x$. Then the following strict relationship between F_k and the ith eigenvalue $\lambda_i(A)$ of the Jacobian holds throughout the iterations

$$\mu_{k+1}^i = \mu_k^i(1 - \beta_k \lambda_i(A)) = \mu_0^i \prod_{j=0}^{k} (1 - \beta_j \lambda_i(A)),$$

where μ_{k+1}^i and μ_k^i, $i = 1, \ldots n$, are the eigencomponents of F_{k+1} and F_k respectively, with respect to the eigendecomposition of A. As a consequence, a small steplength β_k, i.e., close to $1/\lambda_{\max}(A)$, can significantly reduce the values $|\mu_{k+1}^i|$ corresponding to large eigenvalues $\lambda_i(A)$ while a small reduction is expected for the scalars $|\mu_{k+1}^i|$ corresponding to small eigenvalues $\lambda_i(A)$. On the contrary, a large steplength β_k, i.e., close to $1/\lambda_{\min}(A)$, can significantly reduce the values $|\mu_{k+1}^i|$ corresponding to small eigenvalues $\lambda_i(A)$ while tends to increase the scalar $|\mu_{k+1}^i|$ corresponding to large eigenvalues $\lambda_i(A)$. This offers some intuition for choosing the steplengths by alternating in a balanced way small and large steplengths in order to reduce the eigencomponents, see e.g., [15, p. 178].

On the other hand, if F is a general nonlinear mapping then G_k changes at each iteration and Lemma 2.2.5 suggests the expected change of F from iteration k to iteration $k+1$ and the following guidelines. The first guideline concerns the case where J is symmetric and positive definite. A nonmonotone behaviour of the sequence $\{\|F_k\|\}$ is expected. By Item (i) of Lemma 2.2.4, both $\beta_{k,1}$ or $\beta_{k,2}$ are positive and $\beta_k \lambda_i(G_k)$ lies in the interval $\left[\dfrac{\lambda_i(G_k)}{\lambda_{\max}(G_{k-1})}, \dfrac{\lambda_i(G_k)}{\lambda_{\min}(G_{k-1})}\right]$ for $i = 1, \ldots, n$. Assuming without loss of generality that the eigenvalues are numbered in nondecreasing order, by standard arguments on perturbation theory for the eigenvalues it holds

$$|\lambda_i(G_k) - \lambda_i(G_{k-1})| \leq \|G_k - G_{k-1}\|,$$

$i = 1, \ldots, n$, [24, Theorem 8.1-6]. Thus, if the Jacobian is Lipschitz continuous in an open convex set containing $x_{k-1} + tp_{k-1}$ and $x_k + tp_k$ with constant $L_J > 0$, it follows

$$\|G_k - G_{k-1}\| \leq \frac{L_J}{2} \Big(\|p_{k-1}\| + \|p_k\| \Big).$$

Hence, if $\|p_{k-1}\|$ and/or $\|p_k\|$ are large, by Item (b) of Lemma 2.2.5 no decrease of μ_{k+1}^i may occur. On the contrary, for small values of $\|p_{k-1}\|$ and $\|p_k\|$, as occurs if $\{x_k\}$ is convergent, G_k undergoes small changes with respect to G_{k-1} and the behaviour of μ_{k+1}^i shows similarities with the case where J is constant. Thus, a small steplength β_k close to $1/\lambda_{\max}(G_{k-1})$ can significantly reduce the scalars $|\mu_{k+1}^i|$ corresponding to large eigenvalues $\lambda_i(G_k)$, while a small reduction is expected for the values $|\mu_{k+1}^i|$ corresponding to small eigenvalues $\lambda_i(G_k)$. A large steplength β_k close to $1/\lambda_{\min}(G_{k-1})$ can significantly reduce the scalars $|\mu^i_{k+1}|$ corresponding to small eigenvalues $\lambda_i(G_k)$ while tends to increase the eigencomponents $|\mu^i_{k+1}|$ corresponding to large eigenvalues $\lambda_i(G_k)$. As for the case of a constant Jacobian, these features suggest to choose the steplengths by alternating in a balanced way small and large steplengths in order to reduce the eigencomponents.

The second guideline concerns the case where J is symmetric and indefinite and $\lambda_{\min}(G_k) < 0 < \lambda_{\max}(G_k)$. If $\beta_k > 0$, from Item (b) of Lemma 2.2.5 it follows that $|\mu^i_{k+1}|$ corresponding to positive $\lambda_i(G_k)$ are smaller than $|\mu^i_k|$ if $\beta_k\lambda_i(G_k)$ is small enough while all $|\mu^i_{k+1}|$ corresponding to negative eigenvalues increase with respect to $|\mu^i_k|$ and the amplification depends on the magnitude of $\beta_k\lambda_i(G_k)$. If $\beta_k < 0$ similar conclusions hold. In general, a nonmonotone behaviour of the sequence $\{\|F_k\|\}$ is expected and the smaller $\{|\beta_k\lambda_i(G_k)|\}_{i=1,\dots,n}$ are, the smaller $\|F_{k+1}\|/\|F_k\|$ is. Since a small value of $\{|\beta_k\lambda_i(G_k)|\}_{i=1,\dots,n}$ might be induced by a small value of $|\beta_k|$, the use of $\beta_{k,2}$ might be advisable taking into account that $|\beta_{k,2}| \le |\beta_{k,1}|$ and $\beta_{k,1}$ can arbitrarily grow in the indefinite case (see Lemma 2.2.4).

2.2.3 On the impact of the steplength β_k in the approximate norm descent linesearch

In this subsection we embed the spectral residual method in a general globalization scheme based on the so-called approximate norm descent condition in (1.10), which is repeated here for the sake of clarity:

$$\|F(x_k + p_k)\| \le (1 + \eta_k)\|F(x_k)\|, \tag{2.27}$$

with $\eta_k \to 0$ as $k \to \infty$ [36]. Intuitively, large values of η_k allow a highly nonmonotone behaviour of $\|F_k\|$ while small values of η_k promote the decrease of $\|F\|$. Several line-search strategies in the literature fall in this scheme, see e.g., [25, 36, 41, 48]. The main idea is that, given x_k, the trial steps take the form

$$p_- = -\gamma_k\beta_k F_k \quad \text{or} \quad p_+ = +\gamma_k\beta_k F_k, \tag{2.28}$$

with $\gamma_k \in (0, 1]$. The steps in (2.28) are tested in a systematic way with γ_k computed by a backtracking process so that (2.27) is satisfied. Enforcing condition (2.27) ensures the convergence of the sequence $\{\|F_k\|\}$ [36, Lemma 2.4].

We now analyse the properties of $\|F_{k+1}\|$ as a function of the stepsize $\gamma_k\beta_k$ and determine conditions on $\gamma_k\beta_k$ which enforce (2.27). First of all we observe that by the Mean Value Theorem [14, Lemma 4.1.9] and (2.28) we have

$$F_{k+1} = (I \pm \gamma_k\beta_k G_k)F_k. \tag{2.29}$$

Using this equation we can write

$$\|F_{k+1}\|^2 = \|F_k\|^2 \pm 2\gamma_k\beta_k F_k^T (G_S)_k F_k + \gamma_k^2\beta_k^2 F_k^T G_k^T G_k F_k, \tag{2.30}$$

and analyze the fulfillment of either the decrease of $\|F\|$ or (2.27) as given below.

Theorem 2.2.6 *Suppose that Assumptions 2.2.1 and 2.2.2 hold with $x = x_k$ and $p = p_k$. Suppose $F_k^T J_k F_k \neq 0$ and $F_k^T G_k F_k \neq 0$ with G_k given in (2.11). Let $\Delta = q\big((G_S)_k, F_k\big)^2 + (\eta_k^2 + 2\eta_k) q(G_k^T G_k, F_k)$, then*

(1) If $x_{k+1} = x_k + p_k$, $p_k = p_- = -\gamma_k \beta_k F_k$, $\gamma_k \in (0, 1]$, we have that $\|F_{k+1}\| < \|F_k\|$ when

$$\beta_k q\big((G_S)_k, F_k\big) > 0 \qquad \text{and} \quad \gamma_k |\beta_k| < 2\, \frac{\big|q\big((G_S)_k, F_k\big)\big|}{q(G_k^T G_k, F_k)}. \tag{2.31}$$

Condition (2.27) is satisfied when

$$\frac{q\big((G_S)_k, F_k\big) - \sqrt{\Delta}}{q(G_k^T G_k, F_k)} \leq \gamma_k \beta_k \leq \frac{q\big((G_S)_k, F_k\big) + \sqrt{\Delta}}{q(G_k^T G_k, F_k)}. \tag{2.32}$$

(2) If $x_{k+1} = x_k + p_k$, $p_k = p_+ = \gamma_k \beta_k F_k$, $\gamma_k \in (0, 1]$, we have that $\|F_{k+1}\| < \|F_k\|$ when

$$\beta_k q\big((G_S)_k, F_k\big) < 0 \qquad \text{and} \quad \gamma_k |\beta_k| < 2\, \frac{\big|q\big((G_S)_k, F_k\big)\big|}{q(G_k^T G_k, F_k)}. \tag{2.33}$$

Condition (2.27) is satisfied when

$$\frac{-q\big((G_S)_k, F_k\big) - \sqrt{\Delta}}{q(G_k^T G_k, F_k)} \leq \gamma_k \beta_k \leq \frac{-q\big((G_S)_k, F_k\big) + \sqrt{\Delta}}{q(G_k^T G_k, F_k)}. \tag{2.34}$$

Proof. Concerning Item *(1)*, using (2.29) we get

$$\begin{aligned} \|F_{k+1}\|^2 &= \Big(1 - 2\gamma_k \beta_k \frac{F_k^T (G_S)_k F_k}{\|F_k\|^2} + \gamma_k^2 \beta_k^2 \frac{F_k^T G_k^T G_k F_k}{\|F_k\|^2}\Big) \|F_k\|^2 \\ &= \Big(1 - 2\gamma_k \beta_k q\big((G_S)_k, F_k\big) + \gamma_k^2 \beta_k^2 q(G_k^T G_k, F_k)\Big) \|F_k\|^2. \end{aligned}$$

Noting that by assumption $q\big((G_S)_k, F_k\big) \neq 0$ and $q(G_k^T G_k, F_k) > 0$, hence $\|F_{k+1}\| < \|F_k\|$ holds if

$$\beta_k q\big((G_S)_k, F_k\big) > 0 \quad \text{and} \quad - 2\gamma_k \beta_k q\big((G_S)_k, F_k\big) + \gamma_k^2 \beta_k^2 q(G_k^T G_k, F_k) < 0,$$

and these conditions can be rewritten as in (2.31). Condition (2.32) follows trivially.

Item *(2)* follows analogously. From (2.29) and imposing $\|F_{k+1}\| < \|F_k\|$ we get the condition

$$\beta_k q\big((G_S)_k, F_k\big) < 0 \quad \text{and} \quad 2\gamma_k \beta_k q\big((G_S)_k, F_k\big) + \gamma_k^2 \beta_k^2 q(G_k^T G_k, F_k) < 0$$

which is equivalent to (2.33). Condition (2.34) follows trivially. □

We remark that, since G_k and $(G_S)_k$ depend on $\gamma_k \beta_k$, conditions (2.31)–(2.34) are implicit in $\gamma_k \beta_k$. The above theorem supports testing the two steps (2.28) systematically because of the following fact. At k-th iteration, β_k, $q\big(J_k, F_k\big)$ and $q(J_k^T J_k, F_k)$ are given and by continuity of the Jacobian, the Rayleigh quotients $q\big((G_S)_k, F_k\big)$ and $q(G_k^T G_k, F_k)$ tend to $q\big(J_k, F_k\big)$ and $q(J_k^T J_k, F_k)$ respectively as γ_k tends to zero. Hence, given $\epsilon < \frac{1}{2}\min\{q\big(J_k, F_k\big),\, q(J_k^T J_k, F_k)\}$, if γ_k is sufficiently small then

$$\frac{q\big(J_k, F_k\big) - \epsilon}{q\big(J_k^T J_k, F_k\big) + \epsilon} \leq \frac{q\big((G_S)_k, F_k\big)}{q\big(G_k^T G_k, F_k\big)} \leq \frac{q\big(J_k, F_k\big) + \epsilon}{q\big(J_k^T J_k, F_k\big) - \epsilon},$$

and $\dfrac{q\big((G_S)_k, F_k\big)}{q\big(G_k^T G_k, F_k\big)}$ has the same sign as $\dfrac{q\big(J_k, F_k\big)}{q\big(J_k^T J_k, F_k\big)}$. Consequently, for γ_k sufficiently small, either condition (2.31) or (2.33) is fulfilled. Analogous considerations can be made for conditions (2.32) and (2.34).

As a final comment, the previous theorem suggests that a small $|\beta_k|$ promotes the fulfillment of conditions (2.31) and (2.33) or (2.32) and (2.34). Again, by Lemma 2.2.4, the use of $\beta_{k,2}$ may be advisable taking into account that $|\beta_{k,2}| \leq |\beta_{k,1}|$ and that $\beta_{k,1}$ can arbitrarily grow in the indefinite case; taking the steplength equal to $\beta_{k,1}$ may cause a large number of backtracks and an erratic behaviour of $\{\|F_k\|\}$ as long as η_k is sufficiently large.

2.3 Globalization strategies

In this section we introduce two spectral residual algorithms which implement a linesearch along $\pm F_k$ and enforce the approximate norm descent condition (2.27) in the framework discussed in the previous section. The two algorithms are denoted as Srand1 and Srand2, Spectral Residual Approximate Norm Descent method, version 1 and version 2 respectively. Srand1 is originated by the Projected Approximate Norm Descent algorithm with Spectral Residual step (Pand-SR) developed in [48] for solving convexly constrained nonlinear systems. Among its variants proposed in [41, 48] and based on Quasi-Newton methods, we consider the spectral residual implementation for unconstrained nonlinear systems. Srand2 is a variant of Srand1 and represents one of the contribution of this thesis.

2.3.1 The Srand1 algorithm

The Srand1 algorithm employs a nonmonotone linesearch strategy based on the approximate norm descent property in (2.27). The idea behind such a condition is to allow a highly nonmonotone behaviour of $\|F_k\|$ for (initial) large values of η_k while promoting a decrease of $\|F\|$ for small (final) values of η_k. A nonmonotone behavior of the norm of F is crucial to avoid practical stagnation of methods based on spectral stepsizes (see e.g. [19, 34, 54]); at the same time condition (2.27) ensures the sequence $\{\|F_k\|\}$ to be bounded (see [36, Lemma 2.1]).

In details, given the current iterate x_k, a new iterate x_{k+1} is computed as $x_{k+1} = x_k + p_k$ with p_k given by either $(-\gamma_k\beta_k F_k)$ or $(+\gamma_k\beta_k F_k)$, $\gamma_k \in (0, 1]$.

The main phases of the algorithm are as follows. First, the scalar β_k is chosen so that $|\beta_k| \in [\beta_{\min}, \beta_{\max}]$. Second, the scalar $\gamma_k \in (0, 1]$ is fixed using a backtracking strategy. Starting from $\gamma_k = 1$, it is progressively reduced by a factor $\sigma \in (0, 1)$ until one of the following conditions is satisfied:

$$\|F(x_{k+1})\| \le (1 - \rho(1 + \gamma_k))\|F(x_k)\|, \tag{2.35}$$

or

$$\|F(x_{k+1})\| \le (1 + \eta_k - \rho\gamma_k)\|F(x_k)\|, \tag{2.36}$$

where $\rho \in (0, 1)$ is intended to be a small scalar which plays the same role as the Armijo constant [14], and $\{\eta_k\}$ is a positive sequence satisfying (1.9). The first condition (2.35) promotes at each iteration a sufficient decrease in $\|F\|$ which can be accomplished for suitable values of $\pm\gamma\beta_k F_k$, as long as $F_k^T J_k F_k \neq 0$, and is crucial for establishing results on the convergence of $\{\|F_k\|\}$ to zero. On the other hand, the second condition (2.36) allows for an increase of $\|F\|$ depending on the magnitude of η_k. Trivially, (2.35) implies (2.36) and both imply the approximate norm descent condition (2.27). Conditions (2.35) and (2.36) differ from (1.8), (1.11), (1.12), (1.13) in two aspects. First, they are independent of the norm of the trial step which may be very large or small because of the spectral coefficient β_k. Second, η_k appears as a multiplicative term for $\|F_k\|$ while the contribution of η_k is unpredictable in (1.12) and (1.13) because it is not adjusted to reflect the size of $\|F_k\|$.

The formal description of the method is reported in Algorithm 2.3.1 where we deliberately do not specify the form of the stepsize β_k.

Algorithm 2.3.1: The SRAND1 algorithm

Given $x_0 \in \mathbb{R}^n$, $0 < \beta_{\min} < \beta_{\max}$, $\beta_0 \in [\beta_{\min}, \beta_{\max}]$, ρ, $\sigma \in (0, 1)$, a positive sequence $\{\eta_k\}$ satisfying (1.9).

If $\|F_0\| = 0$ stop.
For $k = 0, 1, 2, \ldots$ do

1. Set $\gamma = 1$.
2. Repeat
    - 2.1 Set $p_- = -\gamma\beta_k F_k$ and $p_+ = \gamma\beta_k F_k$.
    - 2.2 If p_- satisfies (2.35), set $p_k = p_-$ and go to Step 3.
    - 2.3 If p_+ satisfies (2.35), set $p_k = p_+$ and go to Step 3.
    - 2.4 If p_- satisfies (2.36), set $p_k = p_-$ and go to Step 3.
    - 2.5 If p_+ satisfies (2.36), set $p_k = p_+$ and go to Step 3.
    - 2.6 Otherwise set $\gamma = \sigma\,\gamma$.
3. Set $\gamma_k = \gamma$, $x_{k+1} = x_k + p_k$.
4. If $\|F_{k+1}\| = 0$ stop.
5. Choose β_{k+1} such that $|\beta_{k+1}| \in [\beta_{\min}, \beta_{\max}]$.

The acceptance cycle of the trial steps in Step 2 terminates in a finite number of steps [48]. Indeed, from the continuity of F and the positivity of η_k, there exists a scalar $\bar{\gamma} > 0$ such that

$$\|F(x_k \pm \gamma\beta_k F(x_k))\| \leq \|F(x_k)\| + (\eta_k - \rho\gamma)\|F(x_k)\|,$$

with $\gamma \in (0, \bar{\gamma}]$. Trivially the above inequality implies that (2.36) holds for γ small enough, see also Theorem 2.2.6.

The following theorem collects the main convergence properties of SRAND1 method given in [48].

Theorem 2.3.1 *Let $\{\eta_k\}$ be a positive sequence satisfying (1.9), $\{x_k\}$ and $\{\gamma_k\}$ be the sequences of iterates and of linesearch stepsizes generated by the* SRAND1 *algorithm. Then,*

(i) the sequence $\{\|F_k\|\}$ is convergent.

(ii) $\lim_{k\to\infty} \gamma_k \|F_k\| = 0$.

(iii) $\liminf_{k\to\infty} \gamma_k > 0$ implies that $\lim_{k\to\infty} \|F_k\| = 0$.

(iv) If (2.44) *is satisfied for infinitely many k, then $\lim_{k\to\infty} \|F_k\| = 0$.*

(v) If $\|F_k\| \leq \|F_{k+1}\|$ for infinitely many iterations, then $\liminf_{k\to\infty} \gamma_k = 0$.

(vi) If $\|F_k\| \leq \|F_{k+1}\|$ for all k sufficiently large, then $\{\|F_k\|\}$ does not converge to 0.

(vii) The sequence $\{x_k\}$ is convergent and, if x^* is the limit point and x_0 is the starting guess, then

$$\|x_0 - x^*\| \leq \beta_{\max}\left(\frac{1}{\rho} + \frac{\eta}{\rho}e^{\eta}\right)\|F_0\|. \tag{2.37}$$

Proof. Items $(i) - (vi)$ are proved in [48, Theorem 4.2]. Item (vii) is proved in [48, Theorem 4.3]. $\square$

The result in Item (vii) of the theorem above has an important consequence. In particular, the bound on $\|x_0 - x^*\|$ implies that if a solution $\bar{x}$ of (1.1) is such that $\|x_0 - \bar{x}\|$ does not satisfy (2.37), then $\{x_k\}$ cannot converge to $\bar{x}$. Namely SRAND1 method is globally convergent but the limit point of $\{x_k\}$ belongs to a specified neighborhood of the initial point and may not be a zero of F.

Under specific assumptions on the Jacobian J at the limit point x^* and assuming that $\beta_k = \beta_{k,1}$ as in (2.9) at Step 5 of Algorithm 2.3.1, the next two theorems are proved in [48]. The first result concerns the case when $J_S(x^*)$ is positive (negative) definite and ensures that $\lim_{k\to\infty} \|F_k\| = 0$ when the 2-norm condition number of J_S is of order $\mathcal{O}(\rho^{-1})$.

Theorem 2.3.2 *Let $\{\eta_k\}$ be a positive sequence satisfying (1.9) and $\{x_k\}$ be the sequence of iterates generated by the* SRAND1 *algorithm. Suppose $\beta_k = \beta_{k,1}$ with $\beta_{k,1}$ given in (2.9) and $p_k = \pm\gamma_k \beta_k F_k$ with $|\beta_k| \in (\beta_{\min}, \beta_{\max})$. Assume F continuously differentiable and J Lipschitz continuous. Moreover assume that the symmetric part J_S of J is positive (negative) definite at the limit point x^* of $\{x_k\}$, and that the 2-norm condition number $\mathcal{K}(J_S(x^*))$ satisfies*

$$\mathcal{K}(J_S(x^*)) < \frac{\omega}{\rho}, \tag{2.38}$$

for some $\omega \in (0,1)$, and $\rho \in (0,1)$ as in (2.35)-(2.36). Then $F(x^*) = 0$.

Proof. See [48, Theorem 5.2]. □

The second result concerns problems where J is strongly diagonally dominant and the diagonal entries have constant sign. We use the following notation:

$$\zeta_i(x) \stackrel{\text{def}}{=} \frac{1}{|(J(x))_{ii}|} \sum_{\substack{j=1 \\ j \neq i}}^{n} |(J(x))_{ij}| \quad i = 1, \ldots, n, \tag{2.39}$$

$$m(x) \stackrel{\text{def}}{=} \min_{1 \le i \le n} (J(x))_{ii}, \qquad M(x) \stackrel{\text{def}}{=} \max_{1 \le i \le n} (J(x))_{ii}, \tag{2.40}$$

$$\widetilde{m}(x) \stackrel{\text{def}}{=} \min_{1 \le i \le n} |(J(x))_{ii}|, \qquad \widetilde{M}(x) \stackrel{\text{def}}{=} \max_{1 \le i \le n} |(J(x))_{ii}|. \tag{2.41}$$

Observe that all these quantities only depend on the Jacobian matrix at x. The value of $\zeta_i(x)$ measures the degree of diagonal dominance of the i-th row of $J(x)$, $m(x)$ and $M(x)$ measure the signed range of its diagonal elements while $\widetilde{m}(x)$ and $\widetilde{M}(x)$ measure the diagonals' absolute values' range. If $J(x)$ has positive diagonal entries, then $\widetilde{m}(x) = m(x) = |m(x)|$ and $\widetilde{M}(x) = M(x) = |M(x)|$. If the diagonal elements are negative, then $\widetilde{m}(x) = -M(x) = |M(x)|$ and $\widetilde{M}(x) = -m(x) = |m(x)|$. The conditions used are

$$\max\left[\frac{\widetilde{M}(x^*)}{|m(x^*)|}, \frac{\widetilde{M}(x^*)}{|M(x^*)|}\right] \sum_{i=1}^{n} \zeta_i(x^*) \le \frac{1-\nu}{1+\nu}, \tag{2.42}$$

and

$$\frac{\widetilde{M}(x^*)}{\widetilde{m}(x^*)} < \left(\frac{\nu}{2-\nu}\right)\left(\frac{1-\nu}{1+\nu}\right)\frac{1}{\rho}, \tag{2.43}$$

for some $\nu \in (0,1)$ and $\rho \in (0,1)$ being the constant in (2.35)-(2.36). Such conditions are satisfied by matrices which are close to being diagonal and have a condition number of order ρ^{-1}. In fact, for decreasing values of $\max_{1 \le i \le n} \zeta_i$, the ratio $\widetilde{M}/\widetilde{m}$ approaches $\mathcal{K}(J(x^*))$ and (2.43) implies a bound on such a condition number in terms of ρ^{-1}. For example, if $\nu = 1/2$, the right-hand side of (2.42) is $1/3$ and that of (2.43) is $1/(9\rho)$.

Theorem 2.3.3 *Let $\{\eta_k\}$ be a positive sequence satisfying (1.9) and $\{x_k\}$ be the sequence of iterates generated by the* SRAND1 *algorithm. Suppose $\beta_k = \beta_{k,1}$ with $\beta_{k,1}$*

given in (2.9) *and* $p_k = \pm\gamma_k\beta_k F_k$ *with* $|\beta_k| \in (\beta_{\min}, \beta_{\max})$. *Assume* F *continuously differentiable and* J *Lipschitz continuous. Suppose that* $J(x^*)$ *is nonsingular where* x^* *is the limit point of* $\{x_k\}$. *Suppose in addition that* $J(x^*)$ *has diagonal entries of constant sign and satisfies (2.42) and (2.43), for some* $\nu \in (0,1)$ *and* $\rho \in (0,1)$ *being the constant in (2.35)-(2.36). Then* $F(x^*) = 0$.

Proof. See [48, Theorem 5.3]. □

2.3.2 SRAND2: a new spectral residual algorithm

In light of the previous discussion we consider a variant of the linesearch conditions (2.35) and (2.36) which gives rise to the SRAND2 method, i.e., Spectral Residual Approximate Norm Descent method, version 2. The SRAND2 algorithm can be sketched as SRAND1 algorithm except for the acceptance conditions of x_{k+1}. In SRAND2 conditions (2.35) and (2.36) are respectively replaced by

$$\|F(x_{k+1})\| \leq (1 - \rho(1 + \gamma_k^2))\|F(x_k)\|, \tag{2.44}$$

and

$$\|F(x_{k+1})\| \leq (1 + \eta_k - \rho\gamma_k^2)\|F(x_k)\|. \tag{2.45}$$

Still these conditions are derivative-free and both imply the approximate norm descent condition (2.27).

We observe that the change in conditions (2.44)-(2.45) with respect to (2.35)-(2.36) amounts to the term γ_k^2 in the right hand side of (2.44)-(2.45). This squared term is common to other linesearch strategies as e.g. (1.8) and (1.11). This small change in the linesearch conditions has a considerable impact on global convergence results as shown below. The formal description of the method is reported in Algorithm 2.3.2.

Analogously to SRAND1 (see [48]), we observe that the repeat loop at Step 2 terminates in a finite number of steps: indeed, from the continuity of F and the positivity of η_k, there exists $\bar{\gamma} > 0$ such that

$$\|F(x_k \pm \gamma\beta_k F(x_k))\| \leq \|F(x_k)\| + (\eta_k - \rho\gamma^2)\|F(x_k)\|,$$

with $\gamma \in (0, \bar{\gamma}]$; therefore, inequality (2.45) holds for small enough values of γ_k, see also Theorem 2.2.6.

We now provide the convergence analysis of the SRAND2 algorithm. Theorems 2.3.4 and 2.3.5 analyze the sequences $\{\gamma_k\}$ and $\{\|F_k\|\}$; they state general results which derive from the linesearch strategy and are analogous to Theorem 2.3.1; their proofs follow the lines of [48, Theorem 4.2]. Theorem 2.3.6 constitutes the main contribution. It is related both to the linesearch strategy and to the choice of the spectral residual steps, and it is independent of the specific choice of β_k.

Algorithm 2.3.2: The SRAND2 algorithm

Given $x_0 \in \mathbb{R}^n$, $0 < \beta_{\min} < \beta_{\max}$, $\beta_0 \in [\beta_{\min}, \beta_{\max}]$, ρ, $\sigma \in (0, 1)$, a positive sequence $\{\eta_k\}$ satisfying (1.9).

If $\|F_0\| = 0$ stop.
For $k = 0, 1, 2, \ldots$ do
1. Set $\gamma = 1$.
2. Repeat
   - 2.1 Set $p_- = -\gamma\beta_k F_k$ and $p_+ = \gamma\beta_k F_k$.
   - 2.2 If p_- satisfies (2.44), set $p_k = p_-$ and go to Step 3.
   - 2.3 If p_+ satisfies (2.44), set $p_k = p_+$ and go to Step 3.
   - 2.4 If p_- satisfies (2.45), set $p_k = p_-$ and go to Step 3.
   - 2.5 If p_+ satisfies (2.45), set $p_k = p_+$ and go to Step 3.
   - 2.6 Otherwise set $\gamma = \sigma\,\gamma$.
3. Set $\gamma_k = \gamma$, $x_{k+1} = x_k + p_k$.
4. If $\|F_{k+1}\| = 0$ stop.
5. Choose β_{k+1} such that $|\beta_{k+1}| \in [\beta_{\min}, \beta_{\max}]$.

Theorem 2.3.4 *Let $F : \mathbb{R}^n \to \mathbb{R}^n$ be a continuous map, and let $\{x_k\}$ and $\{\gamma_k\}$ be the sequences of iterates and of linesearch stepsizes generated by the* SRAND2 *algorithm. Then the sequence $\{\|F_k\|\}$ is convergent and bounded by*

$$\|F_k\| \le e^{\eta}\|F_0\|, \ \textit{for all}\ k \ge 0, \tag{2.46}$$

where $\eta > 0$ is given in (1.9). Moreover

$$\lim_{k\to\infty} \gamma_k^2 \|F_k\| = 0. \tag{2.47}$$

Proof. Convergence of $\{\|F_k\|\}$ follows from (2.27), recalling that any positive sequence $\{a_k\}$ satisfying

$$a_{k+1} \le (1 + \eta_k)a_k + \eta_k,$$

with $\eta_k > 0$ and $\sum_{k=0}^{\infty} \eta_k < \infty$, is convergent (see [13, Lemma 3.3]). Further, applying (2.27) recursively, we get

$$\|F_{k+1}\| \le \prod_{i=0}^{k}(1 + \eta_i)\|F_0\|, \quad \forall k \ge 0.$$

Then (2.46) easily follows by observing that if $\{\eta_k\}$ is a sequence of positive scalars that satisfies (1.9),

$$\prod_{i=0}^{k}(1 + \eta_i) \le e^{\eta}, \quad \forall k \ge 0 \tag{2.48}$$

(see [36, Lemma 2.1]). Finally, the limit in (2.47) is easily verified by rewriting (2.45) as

$$0 \le \rho\gamma_k^2 \|F_k\| \le (1+\eta_k)\|F_k\| - \|F_{k+1}\|,$$

and letting k go to infinity, since $\lim_{k\to\infty} \eta_k = 0$ and the sequence $\{\|F_k\|\}$ is convergent. $\square$

Theorem 2.3.5 in particular identifies situations where $\{\|F_k\|\}$ may or may not converge to zero.

Theorem 2.3.5 *Let $F : \mathbb{R}^n \to \mathbb{R}^n$ be a continuous map, and let $\{x_k\}$ and $\{\gamma_k\}$ be the sequences of iterates and of linesearch stepsizes generated by the* SRAND2 *algorithm. Then*

1. $\liminf_{k\to\infty} \gamma_k^2 > 0$ *implies that* $\lim_{k\to\infty} \|F_k\| = 0$.
2. *If* (2.44) *is satisfied for infinitely many k, then* $\lim_{k\to\infty} \|F_k\| = 0$.
3. *If $\|F_k\| \le \|F_{k+1}\|$ for infinitely many iterations, then* $\liminf_{k\to\infty} \gamma_k^2 = 0$.
4. *If $\|F_k\| \le \|F_{k+1}\|$ for all k sufficiently large, then $\{\|F_k\|\}$ does not converge to 0.*

Proof.

1. The statement follows directly from (2.47).

2. If the sufficient decrease condition (2.44) is attained for infinitely many k, there exists a subsequence $\{\|F_{k_j}\|\}, 1 \le k_0 < k_1 < \ldots$, such that

$$\|F_{k_j}\| \le (1-\rho-\rho\gamma_{k_j}^2)\|F_{k_j-1}\| \le (1-\rho)\|F_{k_j-1}\|.$$

Furthermore, from (2.27) we obtain

$$\|F_{k_j-1}\| \le (1+\eta_{k_j-2})\|F_{k_j-2}\| \le \prod_{i=k_{j-1}}^{k_j-2} (1+\eta_i)\|F_{k_{j-1}}\|.$$

Consequently,

$$\begin{aligned}
\|F_{k_j}\| &\le (1-\rho)\|F_{k_j-1}\| \\
&\le (1-\rho) \prod_{i=k_{j-1}}^{k_j-2} (1+\eta_i)\|F_{k_{j-1}}\| \\
&\le (1-\rho)^2 \prod_{i=k_{j-1}}^{k_j-2} (1+\eta_i)\|F_{k_{j-1}-1}\| \\
&\le \ldots \\
&\le (1-\rho)^{j+1} \prod_{i=k_0}^{k_j-2} (1+\eta_i)\|F_{k_0-1}\| \\
&\le (1-\rho)^{j+1} \prod_{i=0}^{k_j-2} (1+\eta_i)\|F_0\| \\
&\le (1-\rho)^{j+1} e^{\eta}\|F_0\|,
\end{aligned}$$

where in the last inequality we used (2.48). Thus $\lim_{j\to\infty} \|F_{k_j}\| = 0$, and since $\{\|F_k\|\}$ converges we also have $\lim_{k\to\infty} \|F_k\| = 0$.

3. Let us now consider the case that $\|F\|$ does not decrease at infinitely many iterations; then there exists a subsequence $\{\|F_{k_j}\|\}$ such that

$$\|F_{k_j}\| \leq \|F_{k_j+1}\| \leq (1 + \eta_{k_j} - \rho\gamma_{k_j}^2)\|F_{k_j}\|.$$

This means that

$$0 \leq \rho\gamma_{k_j}^2 \leq \eta_{k_j}.$$

Since $\lim_{k\to\infty} \eta_k = 0$, we have that $\liminf_{k\to\infty} \gamma_k^2 = 0$.

4. If $\|F_k\| \leq \|F_{k+1}\|$ for all k sufficiently large, then trivially $\{\|F_k\|\}$ cannot converge to 0. □

We now provide the main convergence result, that is at every limit point x^* of the sequence $\{x_k\}$ generated by the SRAND2 algorithm, either $F(x^*) = 0$ or $F(x^*) \neq 0$ and the gradient of the merit function f in (1.2) is orthogonal to the residual F at x^*.

Theorem 2.3.6 *Let F be continuously differentiable. Let $\{x_k\}$ be the sequence generated by the* SRAND2 *algorithm and let x^* be a limit point of $\{x_k\}$. Then either*

$$F(x^*) = 0,$$

or

$$\nabla f(x^*)^T F(x^*) = F(x^*)^T J(x^*) F(x^*) = 0. \tag{2.49}$$

Proof. Let K be an infinite subset of indices such that $\lim_{k\in K} x_k = x^*$. By Theorem 2.3.4 we know that $\lim_{k\in K} \gamma_k^2 \|F_k\| = 0$. Hence there are two possibilities:

$$\text{either} \quad \liminf_{k\in K} \gamma_k^2 > 0 \quad \text{or} \quad \liminf_{k\in K} \gamma_k^2 = 0.$$

The first one implies $\lim_{k\in K} \|F_k\| = 0$. Then using the continuity of F it follows easily that

$$\lim_{k\in K} \|F(x_k)\| = \|F(x^*)\| = 0.$$

In the second case we have $\liminf_{k\in K} \gamma_k^2 = \liminf_{k\in K} \gamma_k = 0$. Let $\underline{\gamma}_k = \gamma_k/\sigma$ denote the last attempted value for the linesearch parameter before γ_k is accepted during the backtracking phase. Hence for sufficiently large values of $k \in K$ we have

$$\|F(x_k - \underline{\gamma}_k \beta_k F_k)\| > (1 + \eta_k - \rho\underline{\gamma}_k^2)\|F(x_k)\|,$$

$$\|F(x_k + \underline{\gamma}_k \beta_k F_k)\| > (1 + \eta_k - \rho\underline{\gamma}_k^2)\|F(x_k)\|.$$

Being $\eta_k > 0$, and by virtue of (2.46), there is a positive constant c_1 such that

$$\|F(x_k \pm \underline{\gamma}_k \beta_k F_k)\| - \|F(x_k)\| > (\eta_k - \rho\underline{\gamma}_k^2)\|F(x_k))\| > -\rho\underline{\gamma}_k^2\|F(x_k)\| > -c_1\rho\underline{\gamma}_k^2, \quad (2.50)$$

and multiplying both sides of (2.50) by $\|F(x_k \pm \underline{\gamma}_k \beta_k F_k)\| + \|F(x_k)\|$, we obtain

$$\|F(x_k \pm \underline{\gamma}_k \beta_k F_k)\|^2 - \|F(x_k)\|^2 > -c_1\rho\underline{\gamma}_k^2\big(\|F(x_k \pm \underline{\gamma}_k \beta_k F_k)\| + \|F(x_k)\|\big). \quad (2.51)$$

Now we observe that $x_k \pm \gamma_k\beta_k F_k$ is bounded $\forall k \in K$; indeed, by hypothesis $\gamma_k \in (0, 1]$, $|\beta_k| \leq \beta_{\max}$, the subsequence $\{x_k\}_{k\in K}$ is convergent to x^* and hence bounded, and $\|F_k\|$ is bounded by Theorem 2.3.4. Then recalling the definition of $\underline{\gamma}_k = \gamma_k/\sigma$ and the continuity of F, we have

$$\|F(x_k \pm \underline{\gamma}_k \beta_k F_k)\| + \|F(x_k)\| \leq c_2, \quad k \in K, \quad (2.52)$$

for some positive constant c_2. Consequently, from (2.51)–(2.52), there exists a constant $c > 0$ such that

$$\|F(x_k \pm \underline{\gamma}_k \beta_k F_k)\|^2 - \|F(x_k))\|^2 > -c\rho\underline{\gamma}_k^2, \quad (2.53)$$

for sufficiently large values of $k \in K$.

Now, we suppose that $\beta_k > 0$ for infinitely many indices $k \in K_1 \subseteq K$, and we consider the two steps $-\gamma_k\beta_k F_k$ and $+\gamma_k\beta_k F_k$ separately.

- Firstly, we consider $-\gamma\beta_k F_k$. By virtue of the Mean Value Theorem and (2.53), there exists $\xi_k \in [0, 1]$ such that

  $$\big\langle \nabla f(x_k - \xi_k\underline{\gamma}_k\beta_k F_k), -\underline{\gamma}_k\beta_k F_k\big\rangle > -c\rho\underline{\gamma}_k^2,$$

  for sufficiently large $k \in K$. Hence, for all large $k \in K_1$ we have that:

  $$\big\langle \nabla f(x_k - \xi_k\underline{\gamma}_k\beta_k F_k), F_k\big\rangle < c\rho\frac{\underline{\gamma}_k}{\beta_k} \leq c\rho\frac{\underline{\gamma}_k}{\beta_{\min}}. \quad (2.54)$$

- Now we consider $+\gamma\beta_k F_k$. Similarly there exists $\xi'_k \in [0, 1]$ such that for all large $k \in K_1$

  $$\big\langle \nabla f(x_k + \xi'_k\underline{\gamma}_k\beta_k F_k), F_k\big\rangle > -c\rho\frac{\underline{\gamma}_k}{\beta_k} \geq -c\rho\frac{\underline{\gamma}_k}{\beta_{\min}}. \quad (2.55)$$

Since $\liminf_{k\in K} \gamma_k = 0$, taking limits in (2.54) and (2.55) we get

$$\big\langle \nabla f(x^*), F(x^*)\big\rangle = 0.$$

We proceed in a quite similar way if $\beta_k < 0$ for infinitely many indices. □

Corollary 2.3.7 *The orthogonality condition (2.49) implies $F(x^*) = 0$ in the following cases:*

(a) $J(x^*)$ *is positive (negative) definite;*

(b) $v^T J(x^*) v \neq 0$, *for all* $v \in \mathbb{R}^n$, $v \neq 0$.

Case (a) in Corollary 2.3.7 includes the class of strictly monotone nonlinear systems of equations of the form (1.1).

A general result similar to Theorem 2.3.6 was not proved for SRAND1. As reported in Theorem 2.3.2 and Theorem 2.3.3 conditions guaranteeing $F(x^*) = 0$, with x^* being the limit point of $\{x_k\}$, were obtained for SRAND1 using β_k as in (2.9) and in the case where $J(x^*)$ has positive (negative) definite symmetric part and suitably bounded condition number, or where $J(x^*)$ is strongly diagonal dominant with diagonal entries of constant sign.

In the forthcoming chapter we show that SRAND2 corresponds in practice to an algorithm potentially more robust than SRAND1. We cannot expect strong difference in the performance of the two methods, given the small change between the two. Nevertheless, the new linesearch is able to recover some runs where SRAND1 does not converge to a zero of the nonlinear system.

Chapter 3

Numerical experiments

This chapter is devoted to the experimental part of the thesis. The aim is twofold:

- verify the impact of the use of different updating rules for β_k on the practical behaviour of both SRAND1 and SRAND2. Regarding SRAND1, though sufficient conditions for the convergence of the sequence cover a limited number of cases, see Theorems 2.3.2 and 2.3.3, we remark that it has the potential to compute zeros of F for any choice of β_k, see Theorem 2.3.1, Items $(iii) - (iv)$;
- investigate numerically if SRAND2 algorithm is more robust than SRAND1 in practice.

In the first section we give some details on the implemented algorithms and set the parameters used in all the experiments. In the second section we propose some steplength selection rules and in the third section we test them on a sequence of nonlinear systems of equations arising from rolling contact models. In the fourth section we analyze the numerical performance of the new linesearch strategy.

3.1 Implementation issues

SRAND1 and SRAND2 methods given in Algorithms 2.3.1 and 2.3.2 were implemented in Matlab and the parameters were set as follows

$$\beta_0 = 1,\ \beta_{\min} = 10^{-10},\ \beta_{\max} = 10^{10},\ \rho = 10^{-4},\ \sigma = 0.5,\ \eta_k = 0.99^k(100+\|F_0\|^2)\ \forall k \geq 0,$$

see [48]. A maximum number of iterations and F-evaluations equal to 10^5 was imposed and a maximum number of backtracks equal to 40 allowed at each iteration. The procedures were declared successful when

$$\|F_k\| \leq 10^{-6}. \tag{3.1}$$

A failure was declared either because the assigned maximum number of iterations or F-evaluations or backtracks was reached, or because $\|F\|$ was not reduced for 500 consecutive iterations. Such occurrences are denoted in the forthcoming tables as $\mathtt{F_{it}}$, $\mathtt{F_{fe}}$, $\mathtt{F_{bt}}$, $\mathtt{F_{in}}$, respectively.

The solvers were run using MATLAB R2019b and the experiments carried out on a Intel Core i7-9700K CPU @ 3.60GHz x 8, 16 GB RAM, 64-bit.

3.2 Steplength selection

In view of our theoretical analysis and guidelines on steplength selection given in Chapter 2, we attempt to tailor Barzilai and Borwein rules for unconstrained optimization to spectral residual methods. In this section we discuss several steplength rules for spectral residual methods which will be tested in conjunction with SRAND1 algorithm in Section 3.3 and with SRAND2 algorithm in Section 3.4.

Let us consider different rules for the choice of β_k at Step 5 in the SRAND1 algorithm. Besides the straightforward choice of one of the two steplengths $\beta_{k,1}$, $\beta_{k,2}$, along all iterations, we consider adaptive strategies that suitably combine them and parallel those used for quadratic and nonlinear optimization problems. Below, given a scalar β, $T(\beta)$ is the thresholding rule which projects $|\beta|$ onto $I_\beta \stackrel{\text{def}}{=} [\beta_{\min}, \beta_{\max}]$, i.e.,

$$T(\beta) = \min\left\{\beta_{\max}, \max\left\{\beta_{\min}, |\beta|\right\}\right\}. \tag{3.2}$$

BB1 rule. By [28, 33, 35, 48], at each iteration let

$$\beta_k = \begin{cases} \beta_{k,1} & \text{if } \ |\beta_{k,1}| \in I_\beta \\ T(\beta_{k,1}) & \text{otherwise.} \end{cases} \tag{3.3}$$

BB2 rule. At each iteration let

$$\beta_k = \begin{cases} \beta_{k,2} & \text{if } \ |\beta_{k,2}| \in I_\beta \\ T(\beta_{k,2}) & \text{otherwise.} \end{cases} \tag{3.4}$$

ALT rule. Following [9, 28], at each iteration let us alternate between $\beta_{k,1}$ and $\beta_{k,2}$:

$$\beta_k^{\text{ALT}} = \begin{cases} \beta_{k,1} & \text{for } k \text{ odd} \\ \beta_{k,2} & \text{otherwise,} \end{cases} \tag{3.5}$$

$$\beta_k = \begin{cases} \beta_k^{\text{ALT}} & \text{if } \ |\beta_k^{\text{ALT}}| \in I_\beta \\ \beta_{k,1} & \text{if } k \text{ even, } \ |\beta_{k,1}| \in I_\beta, \ |\beta_{k,2}| \notin I_\beta \\ \beta_{k,2} & \text{if } k \text{ odd, } \ |\beta_{k,2}| \in I_\beta, \ |\beta_{k,1}| \notin I_\beta \\ T(\beta_k^{\text{ALT}}) & \text{otherwise.} \end{cases} \tag{3.6}$$

ABB rule. Following [62] and ABB rule in [20], we define the Adaptive Barzilai-Borwein (ABB) rule as follows. Given $\tau \in (0,1)$, let

$$\beta_k^{\mathrm{ABB}}(\xi_1,\xi_2) = \begin{cases} \xi_2 & \text{if } \dfrac{\xi_2}{\xi_1} < \tau \\ \xi_1 & \text{otherwise} \end{cases} \tag{3.7}$$

for some given ξ_1, ξ_2. Then

$$\beta_k = \begin{cases} \beta_k^{\mathrm{ABB}}(\beta_{k,1},\beta_{k,2}) & \text{if } |\beta_{k,1}|, |\beta_{k,2}| \in I_\beta \\ \beta_{k,1} & \text{if } |\beta_{k,1}| \in I_\beta,\ |\beta_{k,2}| \notin I_\beta \\ \beta_{k,2} & \text{if } |\beta_{k,2}| \in I_\beta,\ |\beta_{k,1}| \notin I_\beta \\ \beta_k^{\mathrm{ABB}}(T(\beta_{k,1}),T(\beta_{k,2})) & \text{otherwise.} \end{cases} \tag{3.8}$$

Observe that a large value of τ promotes the use of $\beta_{k,2}$ with respect to $\beta_{k,1}$. The rule allows to switch between the steplengths $\beta_{k,1}$ and $\beta_{k,2}$ and was originally motivated by the behaviour of the Barziali and Borwein method applied to convex and quadratic minimization problems (see [20,62] and our discussion below Lemma 2.2.5).

ABBm rule. This rule elaborates the ABBminmin rule given in [20], taking into account that $\beta_{k,2}$ may be negative along iterations. Let m be a nonnegative integer, and

$$\begin{aligned} \widetilde{\beta}_{k,2} &= \begin{cases} \beta_{k,2} & \text{if } |\beta_{k,2}| \in I_\beta \\ T(\beta_{k,2}) & \text{otherwise,} \end{cases} \\ j^* &= \operatorname{argmin}\{|\widetilde{\beta}_{j,2}| : j = \max\{1, k-m\}, \dots, k\}. \end{aligned} \tag{3.9}$$

Given $\tau \in (0,1)$, we fix β_k as follows

$$\beta_k^{\mathrm{ABBm}}(\xi_1,\xi_2) = \begin{cases} \widetilde{\beta}_{j^*,2} & \text{if } \dfrac{\xi_2}{\xi_1} < \tau \\ \xi_1 & \text{otherwise,} \end{cases} \tag{3.10}$$

$$\beta_k = \begin{cases} \beta_k^{\mathrm{ABBm}}(\beta_{k,1},\beta_{k,2}) & \text{if } |\beta_{k,1}|, |\beta_{k,2}| \in I_\beta \\ \beta_{k,1} & \text{if } |\beta_{k,1}| \in I_\beta,\ |\beta_{k,2}| \notin I_\beta \\ \beta_{k,2} & \text{if } |\beta_{k,2}| \in I_\beta,\ |\beta_{k,1}| \notin I_\beta \\ \beta_k^{\mathrm{ABBm}}(T(\beta_{k,1}),T(\beta_{k,2})) & \text{otherwise.} \end{cases} \tag{3.11}$$

Again, a large value of τ promotes the use of a step from BB2 rule instead of $\beta_{k,1}$. In case $|\beta_{k,1}|, |\beta_{k,2}| \in I_\beta$ and $\dfrac{\beta_{k,2}}{\beta_{k,1}} < \tau$, $\widetilde{\beta}_{j,2}$ with the smallest absolute value over the last $m+1$ iterations is taken; consequently, in general smaller steplengths are taken with respect to ABB rule.

DABBm rule. Following [5,7], a dynamic threshold $\tau_k \in (0,1)$ can be used in place of the prefixed threshold τ in (3.10). Given $\widetilde{\beta}_{k,2}$ and j^* in (3.9), we propose the rule defined as

$$\beta_k^{\mathrm{DABBm}}(\xi_1, \xi_2) = \begin{cases} \widetilde{\beta}_{j^*,2} & \text{if } \dfrac{\xi_2}{\xi_1} < \tau_k \\ \xi_1 & \text{otherwise,} \end{cases} \tag{3.12}$$

$$\beta_k = \begin{cases} \beta_k^{\mathrm{DABBm}}(\beta_{k,1}, \beta_{k,2}) & \text{if } |\beta_{k,1}|, |\beta_{k,2}| \in I_\beta \\ \beta_{k,1} & \text{if } |\beta_{k,1}| \in I_\beta,\ |\beta_{k,2}| \notin I_\beta \\ \beta_{k,2} & \text{if } |\beta_{k,2}| \in I_\beta,\ |\beta_{k,1}| \notin I_\beta \\ \beta_k^{\mathrm{DABBm}}(T(\beta_{k,1}), T(\beta_{k,2})) & \text{otherwise} \end{cases} \tag{3.13}$$

with the dynamic threshold set as

$$\tau_k = \min\left\{\tau, \|F_k\|^{1/(2+b_t^2)}\right\}, \tag{3.14}$$

$$b_t = \max\{b_j : j = \max\{1, k-w\}, \dots, k\}. \tag{3.15}$$

Here $\tau \in (0,1)$ is an upper bound on the value of τ_k, w is a nonnegative integer and b_j denotes the number of backtracks performed at iteration j (see Step 2 of SRAND1 algorithm). If $\|F_k\|$ is getting small and the number of performed backtracks in the last $w+1$ iterations is small, then (3.14) promotes the use of steplengths from BB1 rule, i.e., larger steplengths which can speed convergence to a zero of F. On the other hand, when the number of backtracks performed along previous iterations is large and τ is large, the use of smaller steplengths from BB2 rule is encouraged.

The steplength rules and parameters used in our experiments are summarized in Table 3.1. We tested different dynamic thresholds τ in (3.14) for DABBm rule and here we report results obtained with the best one in terms of efficiency and robustness.

Rule	β_k
BB1	β_k in (3.3)
BB2	β_k in (3.4)
ALT	β_k in (3.5), (3.6)
ABB01	β_k in (3.7), (3.8) with $\tau = 0.1$
ABB08	β_k in (3.7), (3.8) with $\tau = 0.8$
ABBm01	β_k in (3.9)-(3.11) with $\tau = 0.1$, $m = 5$
ABBm08	β_k in (3.9)-(3.11) with $\tau = 0.8$, $m = 5$
DABBm	β_k in (3.9), (3.12)-(3.15) with $\tau = 0.8$, $m = 5$, $w = 20$

Table 3.1: Steplength's rules in SRAND1 implementation.

3.3 Numerical analysis of the steplength selection

In this section we present an extensive numerical validation of the steplength rules summarized in Table 3.1. SRAND1 algorithm is applied in conjunction to such rules for solving sequences of nonlinear systems arising from rolling contact problems. Further, a comparison between the best performing SRAND1 variant and a standard Newton trust-region method is made.

3.3.1 Nonlinear systems arising from rolling contact models

Rolling contact is a fundamental issue in mechanical engineering and plays a central role in many important applications such as rolling bearings and wheel-rail interaction [30, 31]. In order to perform simulations of complex mechanical systems with a good tradeoff between accuracy and efficiency, three working hypotheses are usually made in modelling rolling contact: non-conformal contact, i.e., the typical dimensions of the contact area are negligible if compared to the curvature radii of the contact body surfaces; planar contact, i.e., the contact area is contained in a plane; half-space contact, i.e., locally, the contact bodies are viewed as three-dimensional half-spaces [30, 31]. In this framework, we focus on the Kalker's rolling contact model which represents a relevant and general model in contact mechanics.

The solution of Kalker's rolling contact model can be performed using different approaches. The approach in [59, 60] calls for the solution of constrained optimization problems while the so-called CONTACT algorithm [31] gives rise to sequences of nonlinear systems. Our problem set derives from the application of CONTACT algorithm; here we describe in which phase of the Kalker's model solution they arise and give some of their features. We refer to Appendix A for a sketch of Kalker's model, its discretization, and the Kalker's CONTACT algorithm.

Kalker's CONTACT algorithm determines the normal pressure, the tangential pressure, the contact area, the adhesion area and the sliding area in the contact between two elastic bodies and relies on the elastic decoupling between the normal contact problem and the tangential contact problem. Such problems are solved separately; first the normal problem is solved via the the so-called NORM algorithm, second the tangential problem is solved via the so-called TANG algorithm. Algorithms NORM and TANG are expected to identify the elements in the contact area and in the adhesion-sliding areas, respectively. These algorithms are applied sequentially and repeatedly until the values of the computed pressures undergo a sufficiently small change that suggests their reliable approximation; in general, a few repetitions of NORM and TANG algorithms are required. Each repetition of NORM algorithm calls for the solution of a sequence of linear systems while each repetition of TANG algorithm calls for the solution of a sequence of linear and nonlinear systems. Computationally, the major bottleneck is the numerical solution of the sequence of nonlinear systems generated in the TANG phase. Importantly, each CONTACT iteration requires few repetitions of TANG algorithm but the CONTACT algorithm is performed for several time instances*.

Our tests were made on wheel-rail contact in railway systems. The benchmark vehicle is a driverless subway vehicle, designed by Hitachi Rail on MLA platform (Light Automatic Metro). The vehicle is a fixed-length train composed of four carbodies and five bogies (four motorized and one, the third, trailer), see Figure 3.1. The multibody model has been realized in the Simpack Rail environment [56]. We considered a train route of length $400m$ including a typical railway curved track characterized by three significant parts: two straight lines (from $0m$ to $70m$ and from $233m$ to $400m$), the curve (from $116m$ to $186m$) and two cycloids (from $70m$ to $116m$ and from $186m$ to $233m$) which smoothly connect the straight lines and the curve in terms of curvature radius. The radius of the curve is $500m$. In this analysis, we focused on the contact between the first vehicle wheel and the rail; since the vehicle length is equal to $45.7m$, at the beginning of the dynamic simulation the considered wheel starts in the position $45.7m$ along the track. We performed a simulation in an interval of 10 seconds using 500 time steps, which amounts to 500 calls to CONTACT algorithm, for train speeds with magnitude v taking the values: $v = 10\, m/s$ and $v = 16\, m/s$. Accordingly, during the whole simulation the considered wheel travels along the track a distance equal to $100m$ and $160m$, respectively. The traveling velocities considered give a realistic lateral acceleration along the curve according to the current regulation in force in the railway field.

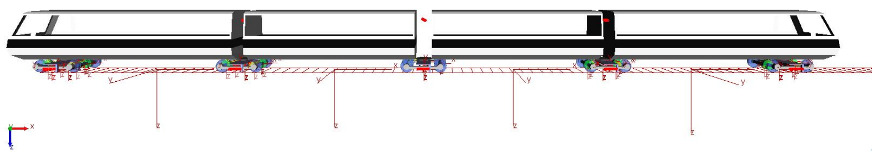


Figure 3.1: Multibody model of the benchmark vehicle.

The set of test problems was generated implementing the CONTACT algorithm in Matlab and using a standard trust-region Newton method† for solving the arising nonlinear systems. Afterwards, a representative subset of the nonlinear systems was selected to form our problem set. Specifically, six sequences of nonlinear systems generated by the CONTACT algorithm and corresponding to six consecutive time instances for each track section (straight line, cycloid and curve) and for each velocity were selected. Such sequences are representative of the systems arising throughout the whole simulation and allow a fair analysis of SRAND1 on nonlinear systems from a real application. Table 3.2 summarizes the features of the sequences: magnitude of the train velocity v, section of the route, time instances, number of nonlinear systems in the sequence, dimension n of the systems (proportional to the number of mesh nodes in the potential contact area).

*In Appendix A see: (A.1) for the form of normal contact problem and tangential contact problem, (A.5) for the form of the nonlinear systems to be solved, Figure A.2 for the flow of Kalker's CONTACT algorithm.

†The code in [47] was applied using the default setting and dropping bound constraints on the unknown.

A typical feature of the contact model is that n increases as the velocity increases and when the train curves along the route (i.e. the track curvature increases). The total number of systems associated to $v = 10\,m/s$ and $v = 16\,m/s$ is 121 and 153 respectively and forms the problem set denoted as SET-CONTACT.

$v(m/s)$	Track Section	Time Instances	Number of Systems	n
10	Straight line	100-105	10	156
	Cycloid	300-305	56	897
	Curve	450-455	55	1394
16	Straight line	50-55	8	156
	Cycloid	150-155	63	1120
	Curve	350-355	82	1394

Table 3.2: Sequences of nonlinear systems forming the SET-CONTACT.

3.3.2 Experimental study

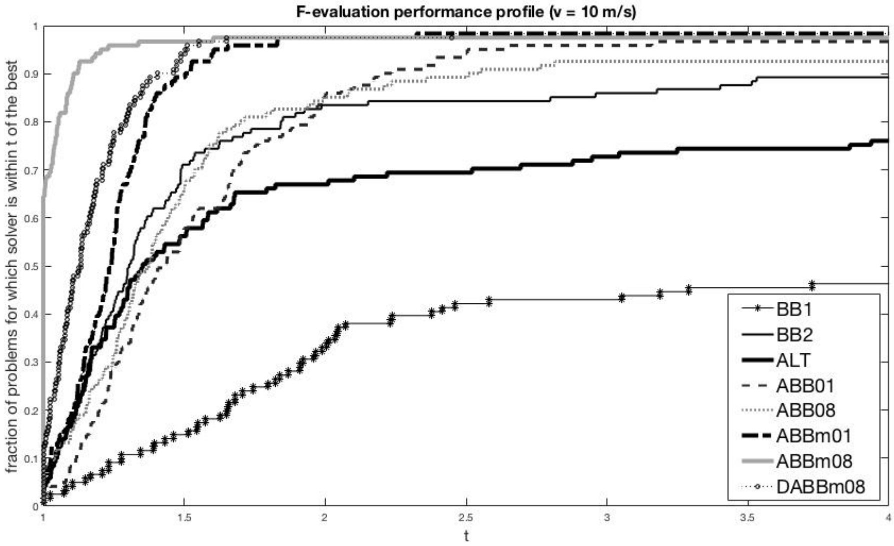
We now test the performance of all the variants of SRAND1 method in the solution of the sequences of nonlinear systems in Table 3.2. Further, in light of the theoretical investigation presented in this work, we analyze in details the results obtained with BB1 and BB2 rule and support the use of rules that switch between the two steplengths.

Figure 3.2 shows the performance profiles [16] in terms of F-evaluations employed by the SRAND1 variants for solving the sequence of systems generated both with $v = 10m/s$ (121 systems) (upper) and with $v = 16\,m/s$ (153 systems) (lower) and highlights that the choice of the steplength is crucial for both efficiency and robustness. The complete results are reported in Appendix B.

The performance profile is a tool proposed by Dolan and Moré [16] for comparing a group of algorithms. For each test T and algorithm A, let feT_A denote the number of F-evaluations required to solve test T by algorithm A, and feT be the lowest number of F-evaluations required by the algorithms under comparison to solve test T. Then, for algorithm A the performance profile is defined as

$$\pi(\tau) = \frac{\#\text{ tests s. t.} \dfrac{feT\_A}{feT} \le \tau}{\#\text{ tests}}, \quad \tau \ge 1.$$

We start observing that BB2 rule outperformed BB1 rule; in fact the latter shows the worst behaviour both in terms of efficiency and in terms of number of problems solved. Alternating $\beta_{k,1}$ and $\beta_{k,2}$ in ALT rule without taking into account the magnitude of the two scalars improves performance over BB1 rule but is not competitive with BB2 rule. On the other hand, the variants of SRAND1 using adaptive strategies are the most robust, i.e., they solve the largest number of problems, and efficient. Specifically, comparing ABB, ABBm and DABBm rules, the most effective steplength selections are ABBm and



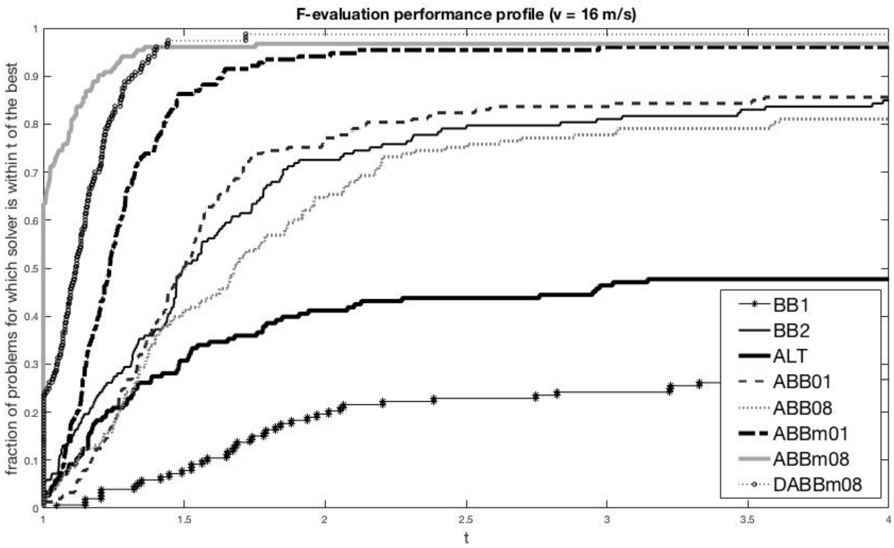


Figure 3.2: SET-CONTACT: F-evaluation performance profiles of SRAND1. Upper: $v = 10\,m/s$, lower: $v = 16\,m/s$.

DABBm. Using ABBm01 rule, 97.5% (2 failures) and 94.1% (6 failures) out of the total number of systems were solved successfully for $v = 10\,m/s$ and $v = 16\,m/s$ respectively; using ABBm08 rule, 97.5% (1 failures) and 96.7% (5 failures) of the total number of systems were solved successfully with $v = 10\,m/s$ and $v = 16\,m/s$ respectively; using the dynamic selection DABBm, the largest number of systems was solved successfully, i.e., 97.5% (1 failure) and 98.7% (2 failures) out the total number of systems with

$v = 10\,m/s$ and $v = 16\,m/s$ respectively. Overall, ABBm08 rule gives rise to the most efficient algorithm for both velocity values; the profile related to BB2 rule is within a factor 2 of it in roughly the 80% and the 70% of the runs for $v = 10m/s$ and $v = 16\,m/s$, respectively.

Let us now focus on the performance of SRAND1 coupled with BB1 and BB2 rules. As a representative run of our numerical experience reported in Appendix B, we consider the nonlinear system arising with $v = 16\,m/s$, at time $t = 150$, iteration 2 of the CONTACT algorithm and iteration 2 of the TANG algorithm (system 150_2_2 in Table B.4).

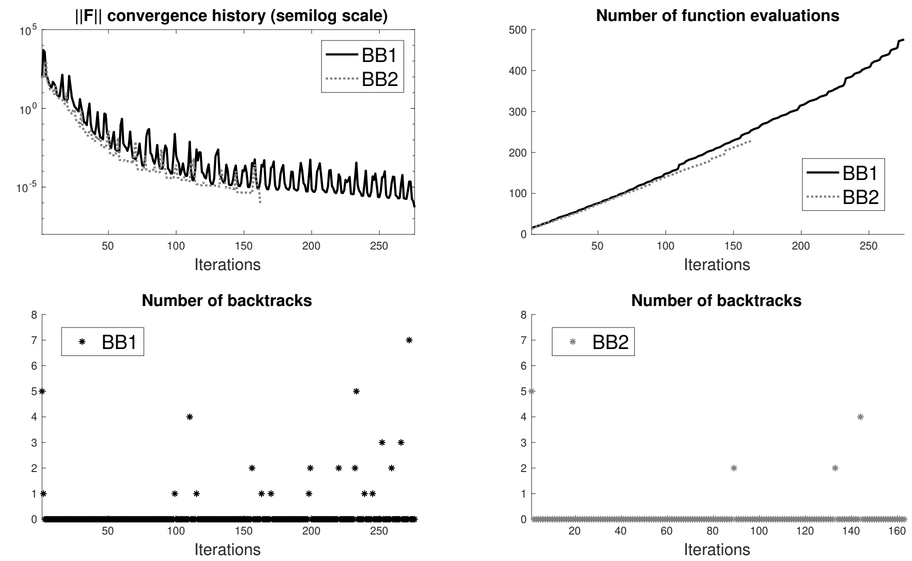


Figure 3.3: SET-CONTACT: SRAND1 with BB1 rule vs SRAND1 with BB2 rule on a single nonlinear system.

In the upper part of Figure 3.3 we display $\|F\|$ along iterations and the number of F-evaluations performed. We note that using the stepsize $\beta_{k,1}$ causes a highly nonmonotone behavior of $\|F\|$ and such behaviour is not productive for convergence; using BB1 rule 276 iterations and 476 F-evaluations are performed while using BB2 rule 163 iterations and 228 F-evaluations are required. The distinguishing feature of these runs is the high number of backtracks performed at some iterations where $\beta_{k,1}$ is used, see the bottom part of the figure where the number of backtracks versus iterations is reported for both SRAND1 variants. This behaviour is in accordance with the analysis in Subsection 2.2.3: since $\beta_{k,1}$ can be arbitrarily larger than $\beta_{k,2}$ in the indefinite case, the need to perform a large number of backtracks to enforce approximate norm decrease is likely to occur in case $\beta_{k,1}$ is taken as the initial steplength. Such observation supports the use of $\beta_{k,2}$; the benefit from using shorter steps is further shown by the performance of ABBm over ABB, the former tends to take shorter steps than the latter by exploiting the iteration history and results to be more effective.

We conclude our experimental analysis using a spectral residual method in the CONTACT algorithm. To this purpose, we compare two implementations of CONTACT algorithm which differ only in the nonlinear solver for the nonlinear systems arising in the TANG algorithm. The first implementation (CONTACT-NTR) uses a standard Newton trust-region method and the second one (CONTACT-DABBm) uses DABBm which turned out to be the more robust SRAND1 version in the analysis above (see Figure 3.2). As a standard Newton trust-region method, we used the Matlab code proposed in [47]; default parameters were used and bound constraints on the unknown were dropped using the setting indicated in the code. The Jacobian matrix of F was approximated by finite differences.

As a preliminary issue, we observe that the Jacobian matrices of F are dense through the iterations; thus they cannot be formed as a low computational cost by finite difference procedures for sparse matrices [8]. We also observed in the experiments that the Jacobian matrices are nonsymmetric, do not have dominant diagonals and they are not close to diagonal matrices. For example, let us consider the Jacobian matrix of the system corresponding to speed $v = 16\, m/s$, curve track section, instant $t = 355$, iteration 2 of the CONTACT and iteration 4 of the TANG algorithm (355_2_4 in Table B.6). It has dimension 292×292 and, evaluated at the final iterate computed using ABBm08 rule, 96.18% of its elements are nonzero. The structure of the Jacobian can be observed in Figure 3.4 where the absolute values of its elements are plotted in a logarithmic scale (the surface of the full matrix on the left and a plot of the row 146 on the right). This structure is observed along all the iterations of the nonlinear system solvers and is common to all sequences generated by the CONTACT algorithm.

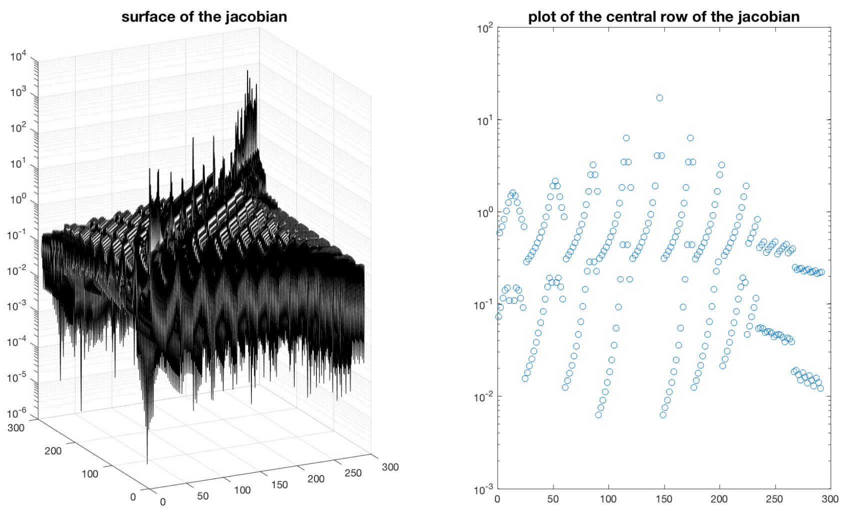


Figure 3.4: Jacobian matrix: surface of the full matrix and plot of the central row (base 10 logarithm of the absolute values).

In our implementation, CONTACT algorithm terminated when the relative error between two successive values of the computed pressures dropped below 10^{-4} or a maximum of 20 alternating cycles between NORM and TANG was reached. Both nonlinear solvers were run until the stopping rule (3.1) is met. We ran CONTACT-NTR and CONTACT-DABBm over the whole track for both velocities, that is we considered the whole sequence of 500 time steps. CONTACT-NTR generated 3759 and 5353 nonlinear systems for $v = 10\, m/s$ and $v = 16\, m/s$, respectively and CONTACT-DABBm generated 4496 and 5494 nonlinear systems for the two velocities.

As a first remark, both procedures successfully solved the contact model described above and were reliable and accurate in the numerical simulation of wheel-rail interaction. Secondly, the use of the spectral residual method yields a gain in terms of time with respect to the use of a standard Newton method where finite difference approximation of Jacobian matrices is employed; this feature derives from the fact that spectral residual method is derivative-free and does not ask for the solution of linear systems. Figures 3.5 and 3.6 show the comparison of the two CONTACT implementations in terms of number of F-evaluations (excluding those needed to approximate the Jacobian matrices) and execution elapsed time. From the plots we observe that CONTACT-DABBm takes a larger number of F-evaluations than CONTACT-NTR but it is faster. Over the whole time interval, CONTACT-DABBm employed 1 hour, 19 mins and 2 hours, 28 mins to solve the generated nonlinear systems with $v = 10\, m/s$ and $v = 16\, m/s$, while CONTACT-NTR took 7 hours and 49 mins and 12 hours and 41 mins, respectively.

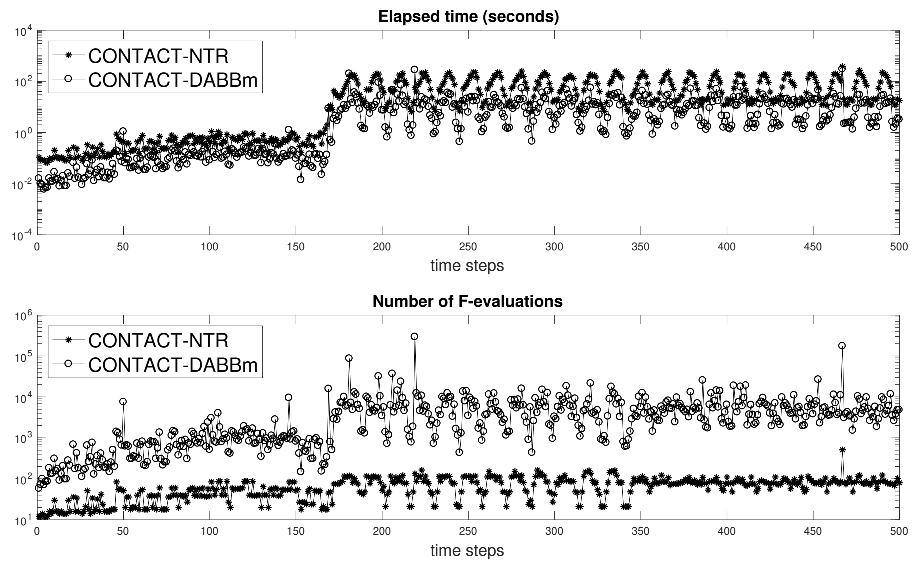


Figure 3.5: SET-CONTACT: comparison between CONTACT-DABBm and CONTACT-NTR, $v = 10\, m/s$: number of F-evaluations and elapsed time in seconds (logarithmic scale).

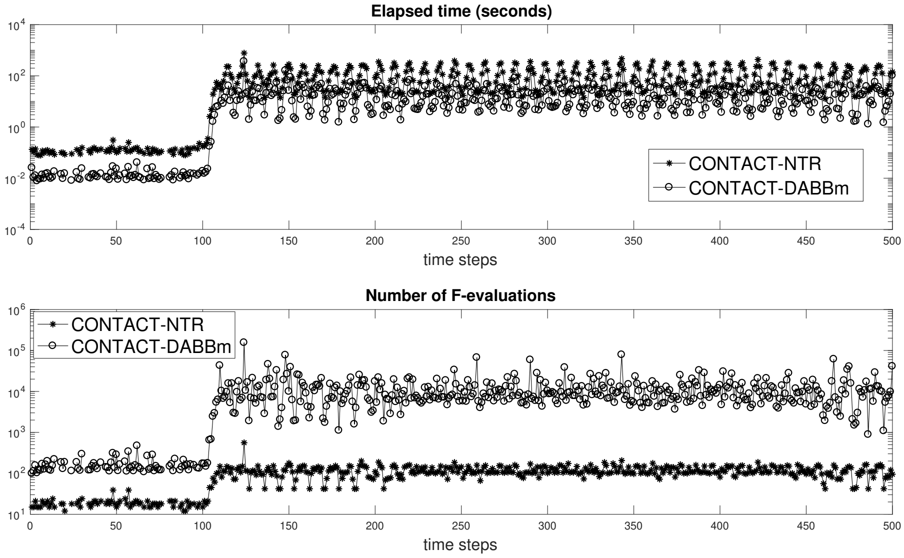


Figure 3.6: SET-CONTACT: comparison between CONTACT-DABBm and CONTACT-NTR, $v = 16\,m/s$: number of F-evaluations and elapsed time in seconds (logarithmic scale).

3.4 Numerical validation of SRAND2

In this section we compare the performance of SRAND1 and SRAND2 algorithms on two problem sets. The first set (named SET-LUKSAN) contains 17 nonlinear systems from the Luksan's test collection described in [39]; these tests are commonly used as benchmark for optimization algorithms. Problems in SET-LUKSAN were solved setting $n = 500$ and starting from the initial guess x_0 suggested in [39]. Problem lu5 requires an odd value for n and therefore we set $n = 501$. The second set is the SET-CONTACT described in Section 3.3.1 and detailed in Table 3.2.

Considering SET-LUKSAN, we experimented SRAND1 and SRAND2 combined with all the rules described in Section 3.2 for the choice of β_k. For 16 out of 17 problems considered, SRAND1 and SRAND2 give the same results with all the choices of β_k: Table 3.3 reports the number of F-evaluations varying the updating rule for β_k. SRAND1 and SRAND2 only differ for the kind of failure in a few runs (note that in Table 3.3 we use the symbol $\mathtt{F_{in}}/\mathtt{F_{bt}}$ to indicate that $\mathtt{F_{in}}$ and $\mathtt{F_{bt}}$ are the failures produced by SRAND1 and SRAND2 respectively and the symbol $\mathtt{F_{bt}}/\mathtt{F_{in}}$ to indicate that $\mathtt{F_{bt}}$ and $\mathtt{F_{in}}$ are the failures produced by SRAND1 and SRAND2 respectively). Problem lu16 reported in Table 3.4 is of interest because, though performing a large number of F-evaluations in some cases, SRAND2 is able to successfully solve it using all the rules except for BB1, whereas SRAND1 returns a failure with most of the attempted β_k rules.

Problem	BB1	BB2	ALT	ABB $\tau = 0.1$	ABB $\tau = 0.8$	ABBm $\tau = 0.1$	ABBm $\tau = 0.8$	DABBm
	Srand1 and Srand2							
lu1	$\mathtt{F_{in}}$	1066	$\mathtt{F_{bt}}$	$\mathtt{F_{in}}/\mathtt{F_{bt}}$	1066	$\mathtt{F_{bt}}$	1053	1288
lu2	496	376	455	852	842	252	501	562
lu3	5	5	5	5	5	5	5	5
lu4	31	32	31	31	29	31	33	35
lu5	15499	1013	2634	1632	1057	2131	1152	1147
lu6	$\mathtt{F_{in}}$	$\mathtt{F_{in}}$	74	$\mathtt{F_{in}}$	$\mathtt{F_{in}}$	$\mathtt{F_{in}}$	$\mathtt{F_{bt}}$	$\mathtt{F_{bt}}$
lu7	$\mathtt{F_{in}}$	$\mathtt{F_{in}}$	417	$\mathtt{F_{in}}$	$\mathtt{F_{in}}$	$\mathtt{F_{in}}$	$\mathtt{F_{in}}$	$\mathtt{F_{in}}$
lu8	419	$\mathtt{F_{in}}$	266	$\mathtt{F_{in}}$	$\mathtt{F_{in}}/\mathtt{F_{bt}}$	$\mathtt{F_{in}}/\mathtt{F_{bt}}$	$\mathtt{F_{in}}$	$\mathtt{F_{in}}$
lu9	$\mathtt{F_{in}}$	$\mathtt{F_{in}}$	182	2852	1150	$\mathtt{F_{in}}$	4363	4365
lu10	457	$\mathtt{F_{in}}$	1168	$\mathtt{F_{in}}$	$\mathtt{F_{bt}}/\ \mathtt{F_{in}}$	$\mathtt{F_{in}}$	$\mathtt{F_{in}}/\mathtt{F_{bt}}$	$\mathtt{F_{in}}/\mathtt{F_{bt}}$
lu11	$\mathtt{F_{in}}$	$\mathtt{F_{in}}$	$\mathtt{F_{in}}$	$\mathtt{F_{in}}$	$\mathtt{F_{in}}$	$\mathtt{F_{in}}$	$\mathtt{F_{in}}$	$\mathtt{F_{in}}$
lu12	$\mathtt{F_{in}}$	$\mathtt{F_{in}}$	$\mathtt{F_{in}}$	$\mathtt{F_{in}}$	$\mathtt{F_{in}}$	$\mathtt{F_{in}}$	$\mathtt{F_{in}}/\mathtt{F_{bt}}$	$\mathtt{F_{in}}/\mathtt{F_{bt}}$
lu13	$\mathtt{F_{in}}$	31	84	123	29	83	33	41
lu14	37	33	36	37	34	37	32	33
lu15	34	33	33	34	33	34	36	34
lu17	137	27	28	155	520	143	$\mathtt{F_{bt}}$	$\mathtt{F_{bt}}$

Table 3.3: set-Luksan: number of F-evaluations performed by Srand1 and Srand2 with different rules for β_k.

	BB1	BB2	ALT	ABB $\tau = 0.1$	ABB $\tau = 0.8$	ABBm $\tau = 0.1$	ABBm $\tau = 0.8$	DABBm
	Problem lu16							
Srand1	$\mathtt{F_{fe}}$	$\mathtt{F_{in}}$	$\mathtt{F_{bt}}$	$\mathtt{F_{in}}$	$\mathtt{F_{in}}$	2688	1674	3774
Srand2	$\mathtt{F_{fe}}$	45624	57432	35413	58456	2688	1674	5439

Table 3.4: set-Luksan: number of F-evaluations performed by Srand1 and Srand2 with different rules for β_k on Problem lu16.

In Figure 3.7 we give an insight into the convergence behavior of both methods with BB2 rule on Problem lu16. We display: $\|F_k\|$ versus the iterations and the number of F-evaluations (top part), the number of backtracks performed by both algorithms (central part), and values of $\|F_k\|$ and γ_k versus the iterations for both algorithms (bottom part). All plots are obtained by disabling the stopping criterion on the number of consecutive increases of $\|F\|$. In this setting SRAND1 fails after performing 3278 iterations and 56883 F-evaluations since the maximum number of backtracks is reached, while SRAND2 converges requiring 8456 iterations and 45624 F-evaluations. We observe that the sequence of $\{\|F_k\|\}$ generated by SRAND1 does not satisfy the stopping criterion (3.1), whereas the increasing number of backtracks along the iterations corresponds to the fact that $\{\gamma_k\}$ is going to zero. On the contrary, the sequence $\{\|F_k\|\}$ generated by SRAND2 converges to zero and γ_k does not decrease with the iterations. Both situations are in accordance with the theory: at least one among the sequences $\{\|F_k\|\}$ and $\{\gamma_k\}$ converges to zero, but SRAND2 generates a sequence $\{\|F_k\|\}$ that goes to zero.

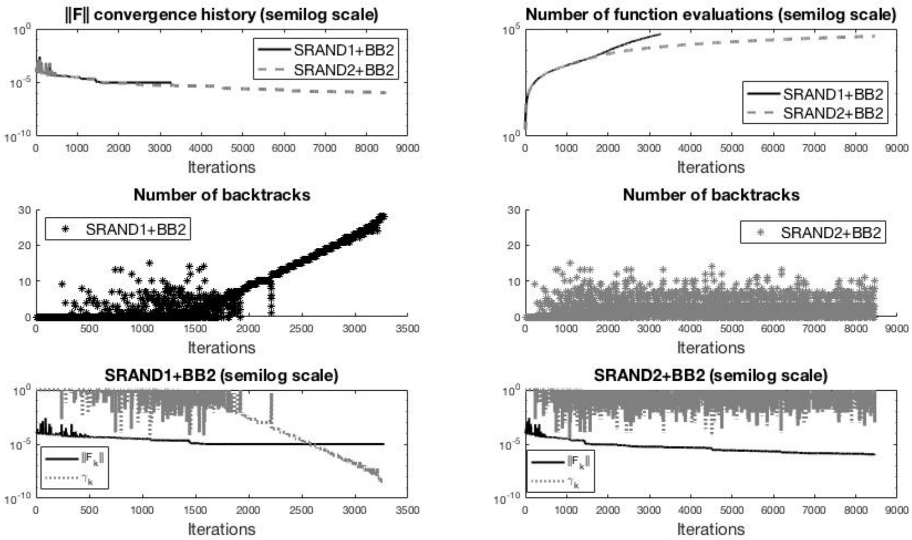


Figure 3.7: SET-LUKSAN: convergence history of SRAND1 and SRAND2 with BB2 rule, Problem lu16.

Finally, we investigate a case of failure of SRAND2 algorithm with the aim of understanding the behavior of the method when the stopping criterion (3.1) is not met. To pursue this issue we considered Problem lu1 not solved by SRAND2 combined with ALT rule. The experiment is carried out changing some parameters in order to emphasize the asymptotic behaviour of the sequence generated by SRAND2. The dimension n is set to 10 and the maximum number of backtracks is raised to 60. Also the stopping criterion on the number of consecutive increases of $\|F\|$ is disabled. The remaining parameters are set as in the previous experiments. In Figure 3.8 we display values of $\|F_k\|$ and of the scalar product $\nabla f_k^T F_k$ versus the iterations. We observe that $\nabla f_k^T F_k$ decreases along the iterations while the norm of F stagnates. This experiment is in line with Theorem 2.3.6 according to which, even if the sequence $\{\|F_k\|\}$ does not converge to zero, the sequences $\{\nabla f_k\}$ and $\{F_k\}$ tend to become orthogonal.

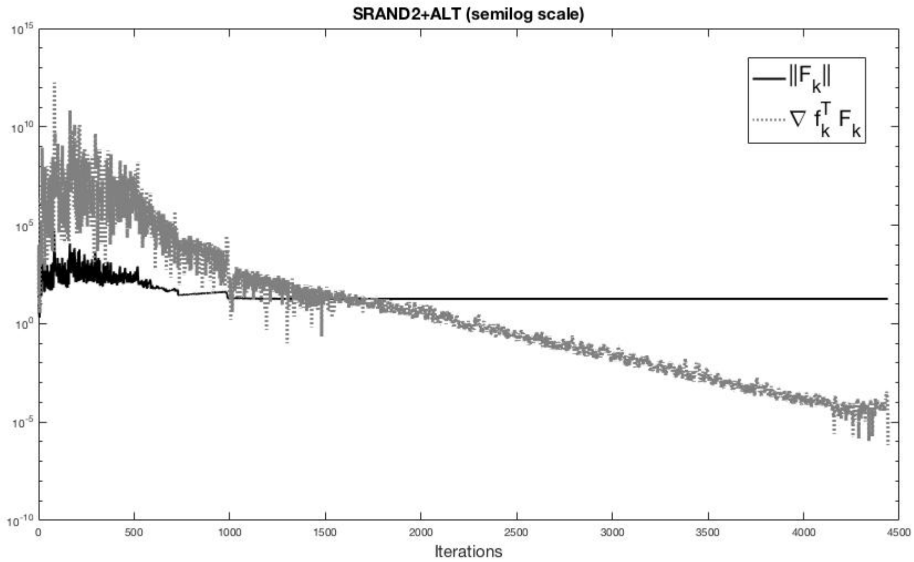


Figure 3.8: SET-LUKSAN: a case of failure of SRAND2 combined with ALT rule, Problem lu1.

The practical advantages of the new linesearch are also confirmed by the experiments performed with the problems in SET-CONTACT using both $v = 10\,m/s$ and $v = 16\,m/s$ for a total of 274 problems. Results obtained for these problems are summarized in the F-evaluations performance profiles [16] of Figure 3.9, where SRAND1 and SRAND2, combined with rules BB2 (top plot), ALT (central plot) and DABBm (bottom plot), are compared. In this case we tested the algorithms using these three classical rules together with the DABBm rule that in Section 3.3 yielded the most robust version of SRAND1 on this set of problems. Results with BB1 are not reported since the behaviour of the two algorithms did not differ in terms of number of solved problems. The complete results are reported in Appendix B. The plots clearly show that the two algorithms perform similarly and SRAND2 is slightly more robust. In detail, SRAND1 and SRAND2 with DABBm solves 271 and 272 problems, respectively. Also, in combination with the BB2 and ALT rules, SRAND2 solves 3 and 6 problems respectively more than SRAND1.

In the ten cases recovered by SRAND2, the behaviour of the two methods is similar to what observed with Problem lu16. To witness, the graphs reported in Figure 3.10 are relative to one of the cases where the BB2 rule was in use. Analogous observations as for Figure 3.7 can be drawn, regarding convergence to zero of the sequences $\{\gamma_k\}$ and $\{\|F_k\|\}$.

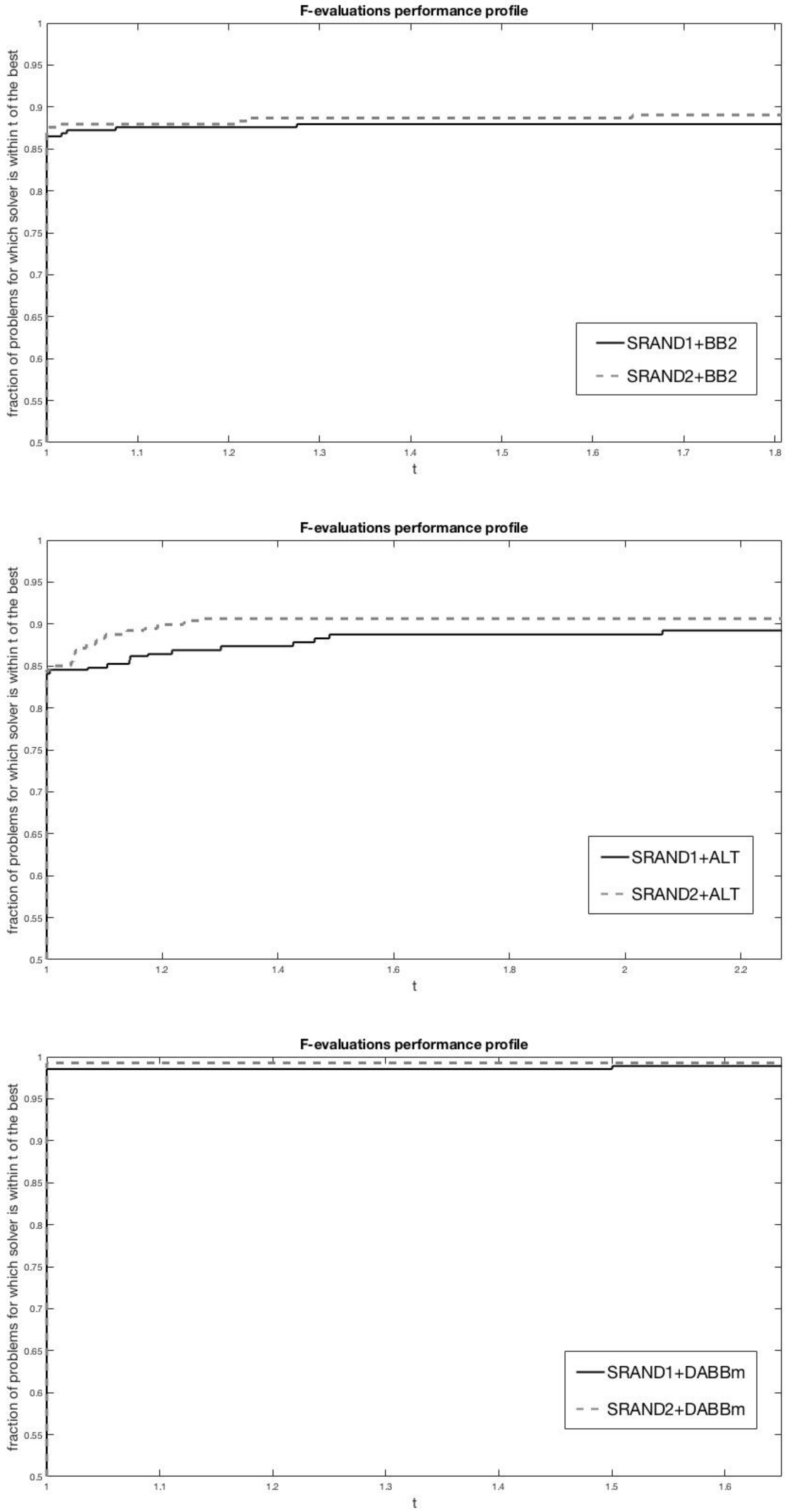


Figure 3.9: SET-CONTACT: F-evaluation performance profile of SRAND1 and SRAND2 with BB2 rule (top), ALT rule (center) and DABBm rule (bottom) ($v = 10\,m/s$ and $v = 16\,m/s$).

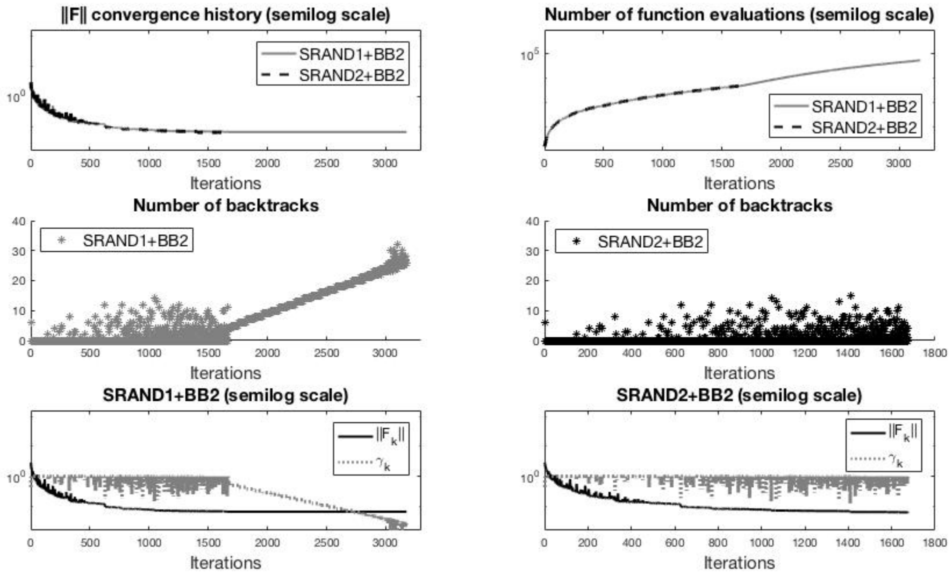


Figure 3.10: SET-CONTACT: convergence history generated by SRAND1 and SRAND2 with BB2 rule, problem 155_3_3 in Table B.4.

Chapter 4

Research perspectives

The numerical behaviour of spectral residual methods for nonlinear systems heavily depends on the choice of the spectral steplength. Although most of the works on this subject use the stepsize denoted in literature as $\beta_{k,1}$, known results on the spectral gradient methods for unconstrained optimization suggest that a suitable combination of the stepsizes $\beta_{k,1}$ and $\beta_{k,2}$ could be of benefit for spectral residual methods as well. This thesis aimed to contribute to this study by providing a first systematic analysis of the stepsizes $\beta_{k,1}$ and $\beta_{k,2}$. Moreover, practical guidelines for dynamic choices of the steplength were derived from new theoretical results in order to increase both the robustness and the efficiency of spectral residual methods. Such findings have been extensively tested and validated on sequences of nonlinear systems arising in the solution of a wheel-rail contact model.

Further we showed how to modify the SRAND1 algorithm proposed in [48] in order to establish a more general framework, denoted as SRAND2, such that the sequence $\{\|F_k\|\}$ is guaranteed to converge to zero under more general conditions, and showed experimentally practical benefits in terms of robustness on test problems from both the literature and applications.

The SRAND1 algorithm in [48] was developed for solving constrained nonlinear systems of the form

$$F(x) = 0, \quad x \in \Omega, \tag{4.1}$$

where $\Omega \subset \mathbb{R}^n$ is a convex set whose relative interior is non-empty. SRAND2 may also be adapted to the solution of constrained problems of the form (4.1) by relying on suitable projection operator onto the feasible set Ω as follows. Proceeding as in [48], feasible iterates $\{x_k\}$ can be defined by starting from a feasible x_0, and by setting for $k > 0$

$$x_{k+1} = P(x_k \pm \gamma_k \beta_k F_k),$$

where P denotes a projection operator onto the considered domain and the new global convergence result in Theorem 2.3.6 applies to limit points lying in the interior of Ω. Convergence to solutions on the boundary of Ω deserves investigation.

Appendix A

Kalker's contact model and CONTACT algorithm

We give an overview of the model and algorithm used to generate our set of nonlinear systems. Bold letters represent vectors, subscript T denotes a vector with components in the tangential x-y contact place, subscript N denotes the component of a vector in the normal z contact direction. The contact problem between two elastic bodies [30,31] determines the contact region C inside the potential contact area A_c (usually the interpenetration area between the wheel and rail contact surfaces), its subdivision into adhesion area H and slip area S, and the tangential $\mathbf{p}_T$ and normal p_N pressures such that the following contact conditions are satisfied:

$$
\begin{array}{lll}
\text{normal problem} & \text{in contact } C: & e = 0, \quad p_N \geq 0 \\
 & \text{in exterior } E: & p_N = 0, \quad e > 0 \\
 & C \cup E = A_c, & C \cap E = \emptyset \\
\text{tangential problem} & \text{in adhesion } H: & \|\mathbf{s}_T\| = 0, \quad \|\mathbf{p}_T\| \leq g \\
 & \text{in slip } S: & \|\mathbf{s}_T\| \neq 0, \quad \mathbf{p}_T = -g\,\mathbf{s}_T/\|\mathbf{s}_T\| \\
 & S \cup H = C, & S \cap H = \emptyset.
\end{array}
\tag{A.1}
$$

Above, e is the deformed distance between the two bodies and, by definition, it holds $e = 0$ and $p_N \geq 0$ in C. Referring to Figure A.1, the region E where $e > 0$ is called the exterior area and $p_N = 0$ therein. The potential contact area is such that $A_c = C \cup E$. The contact area C is divided into the area of adhesion H where the tangential component $\mathbf{s}_T$ of the slip vanishes, and the area S of slip where $\mathbf{s}_T$ is nonzero. The slip $\mathbf{s}_T$ is the difference between the velocities of two homologous points belonging to the deformed wheel and rail surfaces inside the contact area and is a function of the pressures $\mathbf{p}_T$ and p_N, g is the traction bound (Coulomb friction model [30,31]). Overall, the first three equations in (A.1) model the normal contact problem (computation of p_N and of the shapes of the regions C and E), whereas the last three equations describe the tangential contact problem (computation of $\mathbf{p}_T$, of local slidings $\mathbf{s}_T$ and of the shapes of the regions H and S).

Let us consider the discretization of (A.1). Assuming that the contact patch is entirely contained in a plane, the region within which the potential contact area A_c can be located is easily discretized through a planar quadrilateral mesh, see Figure A.1. The coordinates of the center of each quadrilateral element are denoted $\mathbf{x}_I = (x_{I1}, x_{I2}, 0)$ where the capital index I identifies the specific element, say $I = 1, \ldots, N_E$. Also, the standard indices $i = 1, 2, 3$, will indicate the vector components. For any element I and any generic vector $\mathbf{w}_I = (w_{I1}, w_{I2}, w_{I3})$ associated to such mesh element, w_{I1}, w_{I2} are the components in the x-y contact plane and w_{I3} is the component in the normal contact direction z. Namely, $\mathbf{w}_{I,T} = (w_{I1}, w_{I2})$ and w_{I3} are the discrete counterparts of $\mathbf{w}_T$ and w_N, respectively.

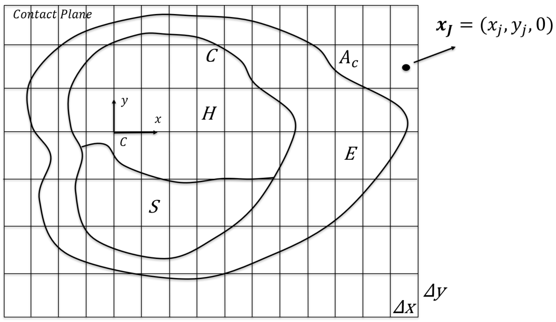


Figure A.1: Local representation of the discretized contact area.

The discrete values of the elastic deformation $\mathbf{u}$ on the mesh nodes (i.e. the deformation of the elastic bodies in the contact area [30, 31]) are defined both at the current time instance t and at the previous time instance t':

$$\mathbf{u}_I = (u_{Ii}) \quad \text{at} \quad (\mathbf{x}_I, t), \quad \mathbf{u}'_I = \left(u'_{Ii}\right) \quad \text{at} \quad \left(\mathbf{x}_I + \mathbf{v}\left(t - t'\right), t'\right), \tag{A.2}$$

where $\mathbf{v}$ is the rolling velocity (i.e. the longitudinal velocity of the wheel) and I is an arbitrary mesh element). Analogously, for the contact pressures $\mathbf{p}$ it holds

$$\mathbf{p}_J = (p_{Jj}) \quad \text{at} \quad (\mathbf{x}_J, t), \quad \mathbf{p}'_J = \left(p'_{Jj}\right) \quad \text{at} \quad \left(\mathbf{x}_J + \mathbf{v}\left(t - t'\right), t'\right), \tag{A.3}$$

where J is an arbitrary mesh element. According to the Boundary Element Method Theory [30, 31], the discretized displacements $\mathbf{u}_I$ can now be written as a function of the discretized contact pressures $\mathbf{p}_J$ through the discretized version of the problem shape functions, that is

$$u_{Ii} = \sum_{J=1}^{N_E} \sum_{j=1}^{3} A_{IiJj} p_{Jj}, \quad \text{with } A_{IiJj} := B_{iJj}\left(\mathbf{x}_I\right),$$

and $B_{iJj}(\mathbf{x}_I)$ are the discrete shape functions of the problem describing the effect of a contact pressure $\mathbf{p}_J$ applied to the element J on displacement $\mathbf{u}_I$ of the node I (see [30, 31]). The shape function B_{iJj} usually depends on the problem geometry and the characteristics of the materials. An analogous expression can be derived for u'_{Ii}. The elastic penetration e can be calculated at each node $\mathbf{x}_I$ as

$$e_I = h_I + \sum_J A_{I3J3} p_{J3},$$

where h_I is the discretization of the (known) undeformed distance between the two bodies, see [30, 31]. Similarly, the slip $\mathbf{s}_T$ can be discretized by setting

$$\mathbf{s}_{I,T} = \mathbf{c}_{I,T} + (\mathbf{u}_{I,T} - \mathbf{u}'_{I,T})/(t - t'), \tag{A.4}$$

where $\mathbf{c}_{I,T}$ is the discretization of the (given) rigid creep, that is the difference between the velocities of two homologous points belonging to the undeformed wheel and rail surfaces inside the contact area and thought of as rigidly connected to the bodies.

We observe that both $\mathbf{u}$ and $\mathbf{s}_T$ depend linearly on the pressures $\mathbf{p}$ and $\mathbf{p}'$. Therefore, the discretization of equation $e = 0$ in the norm problem (A.1) yields a linear system in the discretized normal pressures (p_{I3}) while the discretization of the nonlinear equation

$$\mathbf{p}_T = -g\, \mathbf{s}_T / \|\mathbf{s}_T\|,$$

in the tangential problem yields the nonlinear system

$$\mathbf{s}_{I,T} = -\|\mathbf{s}_{I,T}\| \mathbf{p}_{I,T} / g_I, \tag{A.5}$$

with $\mathbf{p}_{I,T} = (p_{I1}, p_{I2})$ being the unknown*. When using the Coulomb-like friction model [30,31], the friction limit function takes the form $g_I = f_I p_{I3}$, where f_I is a given constant friction value.

The flow of Kalker's CONTACT algorithm is displayed in Figure A.2 [30,31]. At each time step of time integration, the inputs of the CONTACT algorithm are the potential contact area A_c (usually the interpenetration area between wheel and rail surfaces), the rigid penetration h and the rigid local sliding $\mathbf{c}_T$ (inputs calculated, on turn, from the kinematic variables of the body: position and velocities of the gravity centers $\mathbf{G}_1$, $\mathbf{G}_2$, $\mathbf{V}_{G1}$, $\mathbf{V}_{G2}$, rotation matrices $\mathbf{R}_1$, $\mathbf{R}_2$ and angular velocities ω_1, ω_2) [30, 31]. All these kinematic quantities are calculated at each time step by the ODE solver of the Simpack Rail multibody environment [56]. NORM algorithm solves the normal contact problem and returns the contact area C, the non-contact area E, the normal contact pressures p_N. Then, TANG algorithm returns the sliding area S, adhesion area H, the tangential contact pressures $\mathbf{p}_T$ and local sliding $\mathbf{s}_T$. Repetitions of NORM and TANG algorithms are then performed to approximate accurately normal and tangential pressures $\mathbf{p}_T$, p_N. At the end of CONTACT algorithm, forces and torques exchanged by the contact bodies ($\mathbf{F}^1$, $\mathbf{F}^2$ and $\mathbf{M}^1$, $\mathbf{M}^2$) are computed by numerical integration and returned to the time integrator for proceeding in the dynamic simulation of the multibody system.

*In the unlikely event $\mathbf{s}_{I,T} = 0$, the system is nonsmooth. We regularize (A.5) replacing the term $\sqrt{s_{I1}^2 + s_{I2}^2}$ with $\sqrt{s_{I1}^2 + s_{I2}^2 + \epsilon}$, for some small positive ϵ.

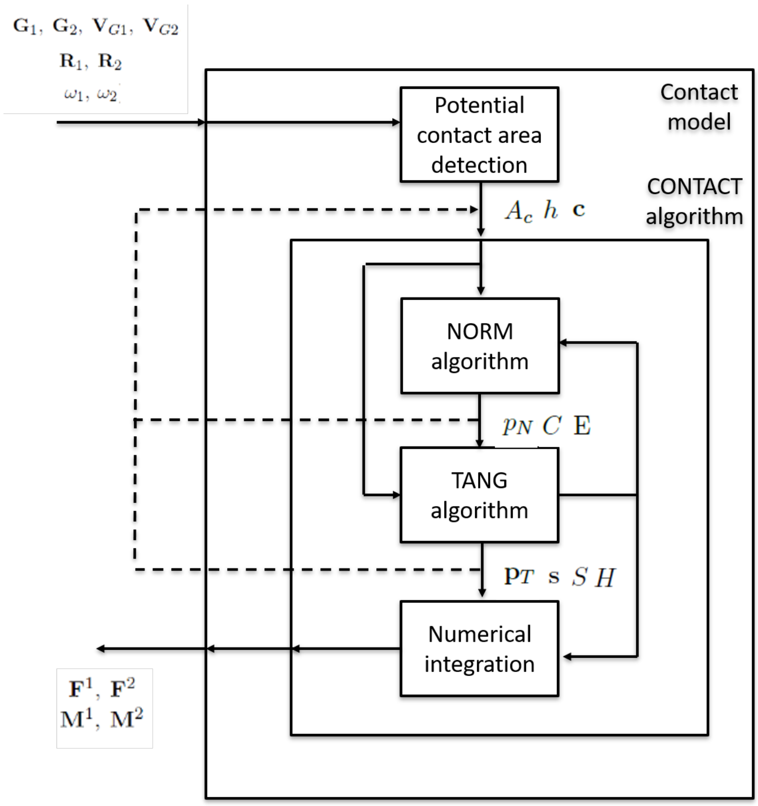


Figure A.2: The architecture of the Kalker's CONTACT algorithm.

Appendix B

Complete results

In this section we collect the complete results for the runs which gave rise to the performance profiles in Figures 3.2 and 3.9. Results in Tables B.1-B.6 refer to SRAND1 method whereas results in Tables B.7-B.12 refer to SRAND2. For each method, results concern two velocities ($v = 10\,m/s$ in Tables B.1, B.3, B.5, B.7, B.9, B.11 and $v = 16\,m/s$ in Tables B.2, B.4, B.6, B.8, B.10, B.12) and three different track sections (straight line in Tables B.1, B.2, B.7 and B.8, cycloid in Tables B.3, B.4, B.9 and B.10 and curve in Tables B.5, B.6, B.11 and B.12). Given a sequence of nonlinear systems, we label a single system from the sequence as Time_Citer_Titer specifying the instant time (Time), the CONTACT iteration (Citer) and the TANG iteration (Titer). For each run we report the number of F-evaluations performed in case of convergence, or, in case of failure, the corresponding flag. We recall from Section 3.1 that a run is successful when $\|F_k\| \leq 10^{-6}$. A failure is declared either because the assigned maximum number of iterations or F-evaluations or backtracks was reached, or because $\|F\|$ was not reduced for 500 consecutive iterations. Such occurrences are denoted as $\mathtt{F_{it}}$ $\mathtt{F_{fe}}$, $\mathtt{F_{bt}}$, $\mathtt{F_{in}}$, respectively.

Srand1 - $v = 10\ m/s$ - straight line								
System	BB1	BB2	ALT	ABB		ABBm		DABBm
				$\tau = 0.1$	$\tau = 0.8$	$\tau = 0.1$	$\tau = 0.8$	
101_1_2	69	59	74	75	59	71	57	69
101_2_2	382	148	248	295	205	174	198	220
103_1_2	37	31	35	37	30	37	31	34
103_2_2	37	31	35	37	30	37	31	34
104_1_2	36	36	37	36	38	36	39	38
104_2_2	36	36	37	36	38	36	39	38
105_1_2	39	38	39	39	38	39	39	39
105_1_3	77	69	82	79	70	82	67	74
105_2_2	40	37	39	40	38	40	39	39
105_2_3	74	73	86	75	70	75	67	76

Table B.1: Number of function evaluations performed by Srand1 variants in the solution of nonlinear systems arising from time 100 to time 105 and corresponding to a straight line with velocity 10 m/s. In the first column we indicate the time step, the CONTACT and the TANG iteration.

Srand1 - velocity 16 m/s - straight line								
System	BB1	BB2	ALT	ABB		ABBm		DABBm
				$\tau = 0.1$	$\tau = 0.8$	$\tau = 0.1$	$\tau = 0.8$	
50_1_2	60	45	53	52	47	52	46	49
50_2_2	53	44	51	54	48	54	48	53
50_3_2	53	44	51	48	48	48	48	53
52_2_2	75	78	53	76	75	101	61	91
52_3_2	89	78	53	76	88	112	61	91
55_1_2	65	66	66	83	66	80	62	72
55_2_2	69	79	60	76	61	73	67	71
55_3_2	69	79	60	80	61	73	67	71

Table B.2: Number of function evaluations performed by Srand1 variants in the solution of nonlinear systems arising from time 50 to time 55 and corresponding to a straight line with velocity 16 m/s. In the first column we indicate the time step, the CONTACT and the TANG iteration.

System	BB1	BB2	ALT	ABB		ABBm		DABBm	System	BB1	BB2	ALT	ABB		ABBm		DABBm
				$\tau = 0.1$	$\tau = 0.8$	$\tau = 0.1$	$\tau = 0.8$						$\tau = 0.1$	$\tau = 0.8$	$\tau = 0.1$	$\tau = 0.8$	
300_1_2	178	128	137	145	149	174	133	163	303_2_2	F_{fe}	F_{in}	2196	F_{in}	F_{in}	1111	763	887
300_1_3	513	304	257	296	252	271	230	298	303_2_3	F_{fe}	1062	7400	1486	1413	911	722	798
300_1_4	569	402	290	464	350	460	278	299	303_2_4	F_{fe}	1713	10229	1780	1400	F_{in}	889	1054
300_2_2	343	203	266	229	194	209	168	204	303_2_5	F_{fe}	1424	23393	2053	1776	1201	1046	1358
300_2_3	16421	388	398	406	686	410	330	408	303_3_2	F_{fe}	926	6424	1352	806	896	814	821
300_3_2	357	223	248	257	205	225	187	232	303_3_3	F_{fe}	1318	6285	1508	886	1074	981	896
300_3_3	1650	385	368	432	530	462	339	499	303_3_4	F_{fe}	1279	14647	2295	1501	1244	959	1012
301_1_2	415	281	247	326	325	264	243	248	303_3_5	F_{fe}	F_{in}	17619	2353	F_{in}	1484	1311	1193
301_1_3	503	319	351	342	480	280	286	329	304_1_2	39075	962	815	643	504	714	447	491
301_1_4	582	442	281	380	376	344	291	305	304_1_3	F_{fe}	711	2891	860	1242	710	607	562
301_2_2	1127	286	298	271	430	310	284	297	304_1_4	F_{fe}	1524	3611	966	1423	785	515	752
301_2_3	630	414	367	388	430	322	313	337	304_2_2	725	366	381	393	416	300	311	317
301_2_4	758	345	372	408	355	363	319	386	304_2_3	65775	558	648	753	734	577	453	548
301_3_2	918	357	299	315	350	294	288	326	304_2_4	56953	709	1870	638	920	562	475	523
301_3_3	750	400	320	473	423	350	305	313	304_3_2	415	421	370	470	431	357	339	325
301_3_4	440	363	302	352	434	310	301	393	304_3_3	47176	533	2376	616	627	518	411	612
302_1_2	F_{fe}	743	3727	993	1022	558	457	495	304_3_4	86605	696	1180	709	603	557	468	488
302_1_3	F_{fe}	844	4067	1183	972	1068	670	678	305_1_2	796	270	311	302	323	329	242	364
302_1_4	F_{fe}	3546	25810	6171	2529	1735	1267	1342	305_1_3	339	293	270	271	294	288	243	310
302_2_2	634	444	417	552	539	431	332	376	305_1_4	430	342	301	354	335	307	230	309
302_2_3	27285	610	508	890	544	502	398	548	305_2_2	F_{fe}	F_{in}	2434	1401	800	F_{in}	1282	1208
302_2_4	F_{fe}	F_{in}	7325	1359	1951	927	853	693	305_2_3	F_{fe}	1110	2222	1713	1030	950	717	684
302_3_2	743	426	373	455	438	402	332	361	305_2_4	F_{fe}	F_{in}	842	1527	846	748	768	648
302_3_3	39825	739	502	869	616	459	401	463	305_2_5	F_{fe}	F_{in}	3329	1516	850	1332	573	597
302_3_4	F_{fe}	2245	7598	1141	938	1005	660	702	305_3_2	F_{fe}	980	6755	1524	F_{in}	920	1036	1518
303_1_2	22687	554	679	502	F_{in}	609	405	460	305_3_3	F_{fe}	F_{in}	5805	1829	756	694	634	579
303_1_3	33798	468	684	571	578	461	411	562	305_3_4	F_{fe}	871	2502	1363	997	857	716	648
303_1_4	F_{fe}	965	1163	734	669	653	524	613	305_3_5	F_{fe}	F_{in}	1786	1286	843	929	702	663

Table B.3: Results for each system of the sequences generated in the cycloid section of the train track with velocity $v = 10\,m/s$.

SRAND1 - velocity 16 m/s - cycloid																	
System	BB1	BB2	ALT	ABB		ABBm		DABBm	System	BB1	BB2	ALT	ABB		ABBm		DABBm
				$\tau = 0.1$	$\tau = 0.8$	$\tau = 0.1$	$\tau = 0.8$						$\tau = 0.1$	$\tau = 0.8$	$\tau = 0.1$	$\tau = 0.8$	
150_1_2	985	297	330	366	357	351	278	343	153_1_3	F_{fe}	1173	1181	1162	1179	735	568	596
150_1_3	26886	569	512	612	555	487	419	437	153_1_4	F_{fe}	991	3881	1003	1590	1044	635	771
150_1_4	F_{fe}	967	3163	653	F_{in}	550	604	617	153_2_2	21846	475	603	688	532	578	396	446
150_1_5	F_{fe}	F_{in}	810	647	1549	614	510	710	153_2_3	F_{fe}	1149	3920	1316	1506	843	621	704
150_2_2	476	228	307	295	302	277	216	301	153_2_4	F_{fe}	1445	5035	1262	1272	1215	602	784
150_2_3	627	584	404	437	485	377	344	443	153_2_5	F_{fe}	772	4023	926	1576	1188	764	725
150_2_4	52373	585	479	494	730	438	391	435	153_3_2	1873	628	754	674	585	489	429	471
150_3_2	F_{fe}	1304	F_{in}	F_{in}	1777	2707	1237	911	153_3_3	F_{fe}	770	4768	1187	1882	941	699	860
150_3_3	F_{fe}	2498	F_{in}	F_{in}	F_{in}	2300	1973	1737	153_3_4	F_{fe}	1568	4872	923	1161	1173	678	709
150_3_4	F_{fe}	6214	F_{in}	F_{in}	F_{in}	3097	2576	F_{in}	153_3_5	F_{fe}	1226	5474	1145	1118	730	688	730
151_1_2	F_{fe}	F_{in}	5095	841	905	664	605	689	154_1_2	66851	776	3124	727	1033	585	534	527
151_1_3	F_{fe}	1114	5312	1421	1144	810	616	829	154_1_3	1031	386	513	467	681	433	310	346
151_1_4	F_{fe}	1454	8154	1630	3755	1125	1139	1046	154_1_4	18703	533	421	539	518	434	404	447
151_1_5	F_{fe}	3590	13111	2610	1435	1231	864	1043	154_2_2	947	319	312	420	357	341	294	356
151_2_2	F_{fe}	1337	12656	1333	3092	973	864	856	154_2_3	255	193	220	216	241	238	201	246
151_2_3	F_{fe}	3776	9599	1983	2198	1077	949	961	154_2_4	348	266	255	255	258	250	228	276
151_2_4	F_{fe}	3013	9073	1867	3551	1409	870	974	154_3_2	569	403	288	336	394	302	277	354
151_2_5	F_{fe}	5005	18543	1831	3662	1635	1270	1345	154_3_3	248	218	249	253	276	217	206	233
151_3_2	F_{fe}	F_{in}	7743	F_{in}	3893	F_{in}	939	803	154_3_4	346	318	278	281	271	267	239	250
151_3_3	F_{fe}	2293	9494	1383	1689	1080	809	982	155_1_2	F_{fe}	1161	5470	1151	987	824	718	859
151_3_4	F_{fe}	1235	7622	1416	1884	1075	856	941	155_1_3	F_{fe}	F_{in}	31313	4192	4270	1758	1401	1193
151_3_5	F_{fe}	4085	24983	1853	F_{in}	1509	1147	1330	155_1_4	F_{fe}	5839	19894	F_{in}	4182	1621	1729	1380
152_1_2	68856	822	1395	742	661	680	473	575	155_1_5	F_{fe}	F_{in}	F_{it}	F_{in}	F_{in}	1624	1351	1339
152_1_3	F_{fe}	682	4009	1153	1085	859	648	669	155_2_2	F_{fe}	1211	3754	1267	1275	764	651	635
152_1_4	F_{fe}	725	2905	986	1423	799	646	720	155_2_3	F_{fe}	F_{in}	F_{in}	2536	F_{in}	1658	1328	1273
152_2_2	21104	604	641	407	681	543	347	399	155_2_4	F_{fe}	1623	24770	3690	F_{in}	1626	1461	1427
152_2_3	80349	701	1082	636	845	632	476	610	155_2_5	F_{fe}	F_{in}	F_{bt}	F_{in}	14474	1683	1715	1559
152_2_4	F_{fe}	1748	3725	1395	1034	873	590	849	155_3_2	F_{fe}	877	6004	990	882	795	567	818
152_3_2	20711	567	601	382	664	453	358	420	155_3_3	F_{fe}	F_{in}	23302	1784	F_{in}	F_{in}	1539	1238
152_3_3	75894	966	1098	522	898	639	535	627	155_3_4	F_{fe}	2895	32130	1953	F_{in}	1539	1739	1315
152_3_4	F_{fe}	1146	4114	848	1152	744	558	734	155_3_5	F_{fe}	F_{in}	F_{in}	6554	F_{in}	F_{in}	F_{in}	F_{in}
153_1_2	1281	408	589	512	495	472	400	397									

Table B.4: Results for each system of the sequences generated in the cycloid section of the train track with velocity $v = 16\ m/s$.

System	BB1	BB2	ALT	ABB $\tau = 0.1$	ABB $\tau = 0.8$	ABBm $\tau = 0.1$	ABBm $\tau = 0.8$	DABBm	System	BB1	BB2	ALT	ABB $\tau = 0.1$	ABB $\tau = 0.8$	ABBm $\tau = 0.1$	ABBm $\tau = 0.8$	DABBm
450_1_2	386	210	246	251	293	293	211	284	453_1_3	402	319	457	427	405	409	255	316
450_1_3	623	204	303	285	281	268	1580	1627	453_1_4	$\mathtt{F_{fe}}$	$\mathtt{F_{in}}$	2705	656	1285	996	611	544
450_2_2	29520	492	457	475	416	458	320	471	453_2_2	536	356	379	593	409	362	329	355
450_2_3	12031	428	433	412	458	415	309	387	453_2_3	$\mathtt{F_{fe}}$	739	872	1030	557	726	5527	560
450_3_2	13652	560	403	562	416	463	379	382	453_2_4	$\mathtt{F_{fe}}$	1772	$\mathtt{F_{it}}$	$\mathtt{F_{in}}$	2018	1579	1535	$\mathtt{F_{in}}$
450_3_3	11509	464	448	518	493	475	393	391	453_3_2	566	351	355	548	392	367	337	398
451_1_2	681	437	382	520	570	519	340	397	453_3_3	$\mathtt{F_{fe}}$	558	598	796	617	612	536	568
451_1_3	$\mathtt{F_{fe}}$	1218	4314	999	1564	868	613	1501	453_3_4	$\mathtt{F_{fe}}$	$\mathtt{F_{in}}$	$\mathtt{F_{bt}}$	2308	$\mathtt{F_{in}}$	1487	1187	1667
451_1_4	$\mathtt{F_{fe}}$	3805	18920	1790	$\mathtt{F_{in}}$	1305	1083	1334	454_1_2	147	153	165	139	153	137	138	150
451_2_2	324	274	329	264	264	263	210	250	454_1_3	207	175	206	229	192	194	154	175
451_2_3	$\mathtt{F_{fe}}$	1652	1046	859	1304	691	520	595	454_1_4	2367	276	293	286	332	283	252	314
451_2_4	$\mathtt{F_{fe}}$	1573	$\mathtt{F_{in}}$	1260	$\mathtt{F_{in}}$	1232	$\mathtt{F_{in}}$	941	454_1_5	861	351	250	269	332	291	231	301
451_3_2	381	253	240	301	243	285	209	270	454_2_2	237	172	209	194	191	202	153	207
451_3_3	$\mathtt{F_{fe}}$	3141	4232	660	801	640	606	635	454_2_3	413	279	211	288	315	240	254	280
451_3_4	$\mathtt{F_{fe}}$	$\mathtt{F_{in}}$	$\mathtt{F_{in}}$	$\mathtt{F_{in}}$	$\mathtt{F_{in}}$	1042	936	888	454_2_4	901	363	209	256	307	262	227	261
451_4_2	358	296	321	279	295	268	213	263	454_3_2	259	204	204	183	198	183	157	183
451_4_3	$\mathtt{F_{fe}}$	2108	901	688	729	676	597	639	454_3_3	469	317	329	273	290	244	251	265
451_4_4	$\mathtt{F_{fe}}$	$\mathtt{F_{in}}$	12872	1797	$\mathtt{F_{in}}$	1093	905	821	454_3_4	450	302	231	277	297	254	229	270
452_1_2	66785	638	638	548	743	585	545	522	455_1_2	147	137	145	144	126	145	127	136
452_1_3	71198	701	725	535	789	489	552	508	455_1_3	212	184	203	219	166	226	166	196
452_1_4	45680	803	521	617	594	584	470	520	455_1_4	482	272	256	291	278	251	237	246
452_2_2	498	557	887	514	539	417	301	467	455_2_2	497	372	250	496	288	256	270	284
452_2_3	37679	608	714	474	672	456	425	454	455_2_3	563	393	473	641	340	436	357	348
452_2_4	40269	718	797	565	790	484	379	501	455_2_4	$\mathtt{F_{fe}}$	840	5928	1544	929	1131	618	632
452_3_2	31230	433	451	438	517	345	405	354	455_3_2	341	270	268	391	392	302	238	282
452_3_3	41623	581	634	575	726	509	400	451	455_3_3	603	432	405	592	415	363	346	353
452_3_4	5592	477	658	572	570	457	407	470	455_3_4	$\mathtt{F_{fe}}$	792	7505	1586	855	914	663	744
453_1_2	288	200	257	227	210	279	190	210									

SRAND1 - velocity 10 m/s - curve

Table B.5: Results for each system of the sequences generated in the curve segment of the train path with velocity $v = 10\,m/s$.

								Srand1 - velocity 16 m/s - curve									
System	BB1	BB2	ALT	ABB		ABBm		DABBm	System	BB1	BB2	ALT	ABB		ABBm		DABBm
				$\tau = 0.1$	$\tau = 0.8$	$\tau = 0.1$	$\tau = 0.8$						$\tau = 0.1$	$\tau = 0.8$	$\tau = 0.1$	$\tau = 0.8$	
350_1_2	424	320	308	359	366	297	284	286	352_4_5	F_{fe}	1132	7322	1252	F_{in}	921	F_{in}	724
350_1_3	F_{fe}	825	5650	826	905	771	540	687	353_1_2	468	357	398	482	342	352	307	357
350_2_2	308	208	220	244	261	243	197	247	353_1_3	887	640	588	557	441	508	446	456
350_2_3	F_{fe}	1322	3384	572	F_{in}	501	433	497	353_1_4	F_{fe}	695	4525	905	1369	781	625	656
350_2_4	F_{fe}	F_{in}	6845	1204	1523	746	790	718	353_1_5	F_{fe}	877	4670	793	1551	782	682	764
350_3_2	311	221	277	264	234	214	188	213	353_2_2	589	357	365	461	398	426	370	386
350_3_3	76754	F_{in}	885	639	666	491	416	481	353_2_3	47619	755	572	913	812	529	459	528
350_3_4	F_{fe}	F_{in}	6032	675	F_{in}	1141	761	647	353_2_4	F_{fe}	1143	3476	F_{in}	857	798	642	687
350_4_2	271	207	233	229	226	220	201	218	353_2_5	F_{fe}	1984	8598	1370	1700	F_{in}	867	1111
350_4_3	91233	764	3110	633	829	536	432	526	353_3_2	711	381	394	481	380	408	368	361
350_4_4	F_{fe}	1593	6301	722	F_{in}	637	F_{in}	751	353_3_3	65122	672	600	710	996	604	511	457
351_1_2	F_{fe}	1241	1625	920	913	772	597	538	353_3_4	F_{fe}	837	1623	815	1111	759	588	633
351_1_3	F_{fe}	1596	11134	1807	F_{in}	1374	1199	1090	353_3_5	F_{fe}	1250	6524	1233	1350	1110	915	855
351_1_4	F_{fe}	2272	20207	1862	F_{in}	1555	1217	1240	353_4_2	575	448	505	425	360	350	341	372
351_2_2	F_{fe}	1088	42218	F_{in}	1207	1385	959	1050	353_4_3	57903	732	725	644	469	517	492	533
351_2_3	F_{fe}	2428	F_{it}	F_{in}	F_{in}	2185	1567	1825	353_4_4	F_{fe}	1030	932	873	1055	679	630	669
351_2_4	F_{fe}	5683	F_{it}	F_{in}	F_{in}	2421	2064	1636	353_4_5	F_{fe}	F_{in}	8112	1276	1502	980	904	967
351_2_5	F_{fe}	F_{in}	F_{it}	F_{in}	F_{in}	3192	2052	2770	354_1_2	313	229	219	320	261	265	187	253
351_3_2	F_{fe}	1261	12388	3742	1566	992	1166	876	354_1_3	502	323	369	398	337	318	267	342
351_3_3	F_{fe}	2029	F_{it}	F_{in}	F_{in}	F_{in}	F_{in}	1704	354_1_4	87446	710	4042	610	716	579	536	673
351_3_4	F_{fe}	2397	F_{it}	F_{in}	4270	2105	2074	1630	354_2_2	445	321	348	373	292	289	230	296
351_3_5	F_{fe}	F_{in}	F_{it}	F_{in}	F_{in}	2833	F_{in}	2635	354_2_3	1771	462	359	434	473	355	345	372
351_4_2	F_{fe}	1285	F_{it}	4846	1378	1262	1313	1028	354_2_4	F_{fe}	1054	4522	1052	1159	757	649	701
351_4_3	F_{fe}	1778	F_{it}	F_{in}	2581	2073	2144	1764	354_3_2	451	315	295	324	275	259	265	316
351_4_4	F_{fe}	F_{in}	F_{it}	F_{in}	F_{in}	2848	1794	1763	354_3_3	789	382	392	508	521	409	408	409
351_4_5	F_{fe}	F_{in}	F_{in}	F_{in}	F_{in}	F_{in}	3340	4432	354_3_4	F_{fe}	913	3478	786	921	845	607	665
352_1_2	F_{fe}	1794	F_{bt}	5760	1636	1619	1933	1728	354_4_2	405	323	289	350	308	317	256	295
352_1_3	F_{fe}	3141	F_{bt}	3787	2872	1686	1495	1524	354_4_3	1776	497	363	452	338	399	333	370
352_1_4	F_{fe}	F_{in}	F_{it}	F_{in}	F_{in}	2334	1657	1721	354_4_4	F_{fe}	991	4561	830	1141	704	553	634
352_1_5	F_{fe}	F_{in}	F_{it}	F_{in}	F_{in}	2318	2846	1623	355_1_2	638	226	262	264	292	268	258	266
352_2_2	72375	676	1359	708	586	643	459	501	355_1_3	527	339	509	348	348	348	286	331
352_2_3	74955	801	878	794	718	857	481	519	355_1_4	35134	489	1201	464	525	477	382	408
352_2_4	F_{fe}	866	5116	1209	1071	837	648	746	355_2_2	346	222	252	246	243	221	194	242
352_2_5	F_{fe}	F_{in}	12683	1209	F_{in}	921	803	909	355_2_3	2303	480	396	402	357	313	261	358
352_3_2	59157	701	1249	712	652	687	420	589	355_2_4	41075	671	542	511	401	376	355	433
352_3_3	87628	1116	682	804	611	639	517	517	355_3_2	336	289	249	264	282	194	232	241
352_3_4	F_{fe}	808	6379	845	830	726	782	685	355_3_4	639	268	480	340	370	304	291	369
352_3_5	F_{fe}	1213	8333	1658	1133	863	697	781	355_3_5	24592	624	753	457	744	448	388	428
352_4_2	48585	603	818	679	775	668	460	528	355_4_2	363	214	268	226	261	261	203	221
352_4_3	79649	867	628	720	876	590	470	511	355_4_3	714	463	360	369	343	383	260	314
352_4_4	F_{fe}	F_{in}	4570	1046	1200	858	708	804	355_4_4	32137	404	700	411	532	562	367	451

Table B.6: Results for each system of the sequences generated in the curve section of the train track with velocity $v = 16\ m/s$.

SRAND2 - $v = 10\ m/s$ - straight line				
System	BB1	BB2	ALT	DABBm
101_1_2	69	59	74	69
101_2_2	382	148	248	220
103_1_2	37	31	35	34
103_2_2	37	31	35	34
104_1_2	36	36	37	38
104_2_2	36	36	37	38
105_1_2	39	38	39	39
105_1_3	77	69	82	74
105_2_2	40	37	39	39
105_2_3	74	73	86	76

Table B.7: Number of function evaluations performed by SRAND2 variants in the solution of nonlinear systems arising from time 100 to time 105 and corresponding to a straight line with velocity 10 m/s. In the first column we indicate the time step, the CONTACT and the TANG iteration.

SRAND2 - velocity 16 m/s - straight line				
System	BB1	BB2	ALT	DABBm
50_1_2	60	45	53	49
50_2_2	53	44	51	53
50_3_2	53	44	51	53
52_2_2	75	78	53	91
52_3_2	89	78	53	91
55_1_2	65	66	66	72
55_2_2	69	79	60	71
55_3_2	69	79	60	71

Table B.8: Number of function evaluations performed by SRAND2 variants in the solution of nonlinear systems arising from time 50 to time 55 and corresponding to a straight line with velocity 16 m/s. In the first column we indicate the time step, the CONTACT and the TANG iteration.

SRAND2 - velocity 10 m/s - cycloid									
System	BB1	BB2	ALT	DABBm	System	BB1	BB2	ALT	DABBm
300_1_2	178	128	137	163	303_2_2	F_{fe}	F_{in}	2196	887
300_1_3	513	304	257	298	303_2_3	F_{fe}	1062	7399	798
300_1_4	569	402	290	299	303_2_4	F_{fe}	1713	12752	1054
300_2_2	343	203	266	204	303_2_5	F_{fe}	1424	21841	1358
300_2_3	16421	388	398	408	303_3_2	F_{fe}	926	5467	821
300_3_2	357	223	248	232	303_3_3	F_{fe}	1318	6284	896
300_3_3	1650	385	368	499	303_3_4	F_{fe}	1279	15483	1012
301_1_2	415	281	247	248	303_3_5	F_{fe}	F_{in}	21781	1193
301_1_3	503	319	351	329	304_1_2	39074	962	815	491
301_1_4	582	442	281	305	304_1_3	F_{fe}	711	2891	562
301_2_2	1127	286	298	297	304_1_4	F_{fe}	1524	3610	752
301_2_3	630	414	367	337	304_2_2	725	366	381	317
301_2_4	758	345	372	386	304_2_3	67575	558	648	548
301_3_2	918	357	299	326	304_2_4	56102	709	1870	523
301_3_3	750	400	320	313	304_3_2	415	421	370	325
301_3_4	440	363	302	393	304_3_3	47678	533	2376	612
302_1_2	F_{fe}	743	3727	495	304_3_4	87138	696	1180	488
302_1_3	F_{fe}	844	4067	678	305_1_2	796	270	311	364
302_1_4	F_{fe}	3545	32612	1342	305_1_3	339	293	270	310
302_2_2	634	444	417	376	305_1_4	430	342	301	309
302_2_3	27293	610	508	548	305_2_2	F_{fe}	F_{in}	2434	1208
302_2_4	F_{fe}	F_{in}	7325	693	305_2_3	F_{fe}	1110	2222	684
302_3_2	743	426	373	361	305_2_4	F_{fe}	F_{in}	842	648
302_3_3	39825	739	502	463	305_2_5	F_{fe}	F_{in}	3329	597
302_3_4	F_{fe}	2245	7598	702	305_3_2	F_{fe}	980	6754	1518
303_1_2	22921	554	679	460	305_3_3	F_{fe}	F_{in}	5805	579
303_1_3	33798	468	684	562	305_3_4	F_{fe}	871	2502	648
303_1_4	F_{fe}	965	1163	613	305_3_5	F_{fe}	F_{in}	1786	663

Table B.9: Results for each system of the sequences generated in the cycloid section of the train track with velocity $v = 10\ m/s$.

System	BB1	BB2	ALT	DABBm	System	BB1	BB2	ALT	DABBm
SRAND2 - velocity 16 m/s - cycloid									
150_1_2	985	297	330	343	153_1_3	$\texttt{F}_{\texttt{fe}}$	1173	1181	596
150_1_3	26886	569	512	437	153_1_4	$\texttt{F}_{\texttt{fe}}$	991	3881	771
150_1_4	$\texttt{F}_{\texttt{fe}}$	967	3163	617	153_2_2	21846	475	603	446
150_1_5	$\texttt{F}_{\texttt{fe}}$	$\texttt{F}_{\texttt{in}}$	810	710	153_2_3	$\texttt{F}_{\texttt{fe}}$	1149	3920	704
150_2_2	476	228	307	301	153_2_4	$\texttt{F}_{\texttt{fe}}$	1445	5035	784
150_2_3	627	584	404	443	153_2_5	$\texttt{F}_{\texttt{fe}}$	772	4023	725
150_2_4	52371	585	479	435	153_3_2	1873	628	754	471
150_3_2	$\texttt{F}_{\texttt{fe}}$	1304	93989	911	153_3_3	$\texttt{F}_{\texttt{fe}}$	770	4995	860
150_3_3	$\texttt{F}_{\texttt{fe}}$	2498	$\texttt{F}_{\texttt{fe}}$	1737	153_3_3	$\texttt{F}_{\texttt{fe}}$	770	4995	860
150_3_4	$\texttt{F}_{\texttt{fe}}$	6079	$\texttt{F}_{\texttt{in}}$	2237	153_3_4	$\texttt{F}_{\texttt{fe}}$	1568	4872	709
151_1_2	$\texttt{F}_{\texttt{fe}}$	$\texttt{F}_{\texttt{in}}$	5094	689	153_3_5	$\texttt{F}_{\texttt{fe}}$	1226	5474	730
151_1_3	$\texttt{F}_{\texttt{fe}}$	1114	5311	829	154_1_2	65690	776	3124	527
151_1_4	$\texttt{F}_{\texttt{fe}}$	1454	8154	1046	154_1_3	1031	386	513	346
151_1_5	$\texttt{F}_{\texttt{fe}}$	3589	13663	1043	154_1_4	18703	533	421	447
151_2_2	$\texttt{F}_{\texttt{fe}}$	1337	9728	856	154_2_2	947	319	312	356
151_2_3	$\texttt{F}_{\texttt{fe}}$	2962	9597	961	154_2_3	255	193	220	246
151_2_4	$\texttt{F}_{\texttt{fe}}$	3013	6363	974	154_2_4	348	266	255	276
151_2_5	$\texttt{F}_{\texttt{fe}}$	6045	20420	1345	154_3_2	569	403	288	354
151_3_2	$\texttt{F}_{\texttt{fe}}$	$\texttt{F}_{\texttt{in}}$	7742	803	154_3_3	248	218	249	233
151_3_3	$\texttt{F}_{\texttt{fe}}$	2293	8594	982	154_3_4	346	318	278	250
151_3_4	$\texttt{F}_{\texttt{fe}}$	1235	7998	941	155_1_2	$\texttt{F}_{\texttt{fe}}$	1161	6519	859
151_3_5	$\texttt{F}_{\texttt{fe}}$	6713	21858	1330	155_1_3	$\texttt{F}_{\texttt{fe}}$	$\texttt{F}_{\texttt{in}}$	$\texttt{F}_{\texttt{in}}$	1193
152_1_2	68854	822	1395	575	155_1_4	$\texttt{F}_{\texttt{fe}}$	5427	$\texttt{F}_{\texttt{in}}$	1380
152_1_3	$\texttt{F}_{\texttt{fe}}$	682	4009	669	155_1_5	$\texttt{F}_{\texttt{fe}}$	$\texttt{F}_{\texttt{in}}$	$\texttt{F}_{\texttt{in}}$	1339
152_1_4	$\texttt{F}_{\texttt{fe}}$	725	2905	720	155_2_2	$\texttt{F}_{\texttt{fe}}$	1211	3754	635
152_2_2	21102	604	641	399	155_2_3	$\texttt{F}_{\texttt{fe}}$	$\texttt{F}_{\texttt{in}}$	25875	1273
152_2_3	80349	701	1082	610	155_2_4	$\texttt{F}_{\texttt{fe}}$	1623	$\texttt{F}_{\texttt{in}}$	1427
152_2_4	$\texttt{F}_{\texttt{fe}}$	1748	3725	849	155_2_5	$\texttt{F}_{\texttt{fe}}$	$\texttt{F}_{\texttt{in}}$	$\texttt{F}_{\texttt{in}}$	1559
152_3_2	20619	567	601	420	155_3_2	$\texttt{F}_{\texttt{fe}}$	877	6004	818
152_3_3	76611	966	1098	627	155_3_3	$\texttt{F}_{\texttt{fe}}$	4924	25285	1238
152_3_4	$\texttt{F}_{\texttt{fe}}$	1146	4114	734	155_3_4	$\texttt{F}_{\texttt{fe}}$	2893	21582	1315
153_1_2	1281	408	589	397	155_3_5	$\texttt{F}_{\texttt{fe}}$	$\texttt{F}_{\texttt{in}}$	33026	$\texttt{F}_{\texttt{in}}$

Table B.10: Results for each system of the sequences generated in the cycloid section of the train track with velocity $v = 16\ m/s$.

				SRAND2 - velocity 10 m/s - curve					
System	BB1	BB2	ALT	DABBm	System	BB1	BB2	ALT	DABBm
450_1_2	386	210	246	284	453_1_3	402	319	457	316
450_1_3	623	204	303	1627	453_1_4	F_{fe}	F_{in}	2705	544
450_2_2	29519	492	457	471	453_2_2	536	356	379	355
450_2_3	12031	428	433	387	453_2_3	F_{fe}	739	872	560
450_3_2	13879	560	403	382	453_2_4	F_{fe}	1772	38854	F_{in}
450_3_3	11509	464	448	391	453_3_2	566	351	355	398
451_1_2	681	437	382	397	453_3_3	F_{fe}	558	598	568
451_1_3	F_{fe}	1218	4314	1501	453_3_4	F_{fe}	F_{in}	F_{in}	1667
451_1_4	F_{fe}	4642	20768	1334	454_1_2	147	153	165	150
451_2_2	324	274	329	250	454_1_3	207	175	206	175
451_2_3	F_{fe}	1652	1046	595	454_1_4	2367	276	293	314
451_2_4	F_{fe}	1573	F_{in}	941	454_1_5	861	351	250	301
451_3_2	381	253	240	270	454_2_2	237	172	209	207
451_3_3	F_{fe}	3140	4232	635	454_2_3	413	279	211	280
451_3_4	F_{fe}	F_{in}	F_{in}	888	454_2_4	901	363	209	261
451_4_2	358	296	321	263	454_3_2	259	204	204	183
451_4_3	F_{fe}	2108	901	639	454_3_3	469	317	329	265
451_4_4	F_{fe}	F_{in}	F_{in}	821	454_3_4	450	302	231	270
452_1_2	66666	638	638	522	455_1_2	147	137	145	136
452_1_3	72915	701	725	508	455_1_3	212	184	203	196
452_1_4	45679	803	521	520	455_1_4	482	272	256	246
452_2_2	498	557	887	467	455_2_2	497	372	250	284
452_2_3	37679	608	714	454	455_2_3	563	393	473	348
452_2_4	40268	718	797	501	455_2_4	F_{fe}	840	6926	632
452_3_2	31282	433	451	354	455_3_2	341	270	268	282
452_3_3	41622	581	634	451	455_3_3	603	432	405	353
452_3_4	5592	477	658	470	455_3_4	F_{fe}	792	7505	744
453_1_2	288	200	257	210					

Table B.11: Results for each system of the sequences generated in the curve segment of the train path with velocity $v = 10\ m/s$.

SRAND2 - velocity 16 m/s - curve									
System	BB1	BB2	ALT	DABBm	System	BB1	BB2	ALT	DABBm
350_1_2	308	424	320	286	352_4_5	F_{in}	F_{fe}	1132	724
350_1_3	5650	F_{fe}	825	687	353_1_2	398	468	357	357
350_2_2	220	308	208	247	353_1_3	588	887	640	456
350_2_3	3384	F_{fe}	1322	497	353_1_4	4525	F_{fe}	695	656
350_2_4	6843	F_{fe}	F_{in}	718	353_1_5	4670	F_{fe}	877	764
350_3_2	277	311	221	213	353_2_2	365	589	357	386
350_3_3	885	76752	F_{in}	481	353_2_3	572	47617	755	528
350_3_4	6032	F_{fe}	F_{in}	647	353_2_4	3476	F_{fe}	1143	687
350_4_2	233	271	207	218	353_2_5	8657	F_{fe}	1984	1111
350_4_3	3110	90329	764	526	353_3_2	394	711	381	361
350_4_4	6301	F_{fe}	1593	751	353_3_3	600	65120	672	457
351_1_2	1625	F_{fe}	1241	538	353_3_4	1623	F_{fe}	837	633
351_1_3	12677	F_{fe}	1596	1090	353_3_5	6523	F_{fe}	1250	855
351_1_4	13812	F_{fe}	2272	1240	353_4_2	505	575	448	372
351_2_2	20454	F_{fe}	1088	1050	353_4_3	725	57899	732	533
351_2_3	F_{fe}	F_{fe}	2428	1825	353_4_4	932	F_{fe}	1030	669
351_2_4	F_{bt}	F_{fe}	5744	1636	353_4_5	8111	F_{fe}	F_{in}	967
351_2_5	F_{fe}	F_{fe}	F_{in}	2770	354_1_2	219	313	229	253
351_3_2	13238	F_{fe}	1261	876	354_1_3	369	502	323	342
351_3_3	F_{bt}	F_{fe}	2029	1704	354_1_4	4042	88877	710	673
351_3_4	73563	F_{fe}	2397	1630	354_2_2	348	445	321	296
351_3_5	F_{fe}	F_{fe}	F_{in}	2635	354_2_3	359	1771	462	372
351_4_2	25703	F_{fe}	1285	1028	354_2_4	4521	F_{fe}	1054	701
351_4_3	F_{fe}	F_{fe}	1778	1764	354_3_2	295	451	315	316
351_4_4	F_{fe}	F_{fe}	F_{in}	1763	354_3_3	392	789	382	409
351_4_5	F_{fe}	F_{fe}	10011	2954	354_3_4	3478	F_{fe}	913	665
352_1_2	45932	F_{fe}	1794	1728	354_4_2	289	405	323	295
352_1_3	29665	F_{fe}	3091	1524	354_4_3	363	1776	497	370
352_1_4	F_{bt}	F_{fe}	12749	1721	354_4_4	4560	F_{fe}	991	634
352_1_5	F_{fe}	F_{fe}	F_{in}	1623	355_1_2	262	638	226	266
352_2_2	1359	72373	676	501	355_1_3	509	527	339	331
352_2_3	878	74649	801	519	355_1_4	1201	35134	489	408
352_2_4	5116	F_{fe}	866	746	355_2_2	252	346	222	242
352_2_5	10426	F_{fe}	F_{in}	909	355_2_3	396	2303	480	358
352_3_2	1249	59153	701	589	355_2_4	542	40681	671	433
352_3_3	682	87783	1116	517	355_3_2	249	336	289	241
352_3_4	5575	F_{fe}	808	685	355_3_4	480	639	268	369
352_3_5	8716	F_{fe}	1213	781	355_3_5	753	24591	624	428
352_4_2	818	48584	603	528	355_4_2	268	363	214	221
352_4_3	628	79081	867	511	355_4_3	360	714	463	314
352_4_4	4545	F_{fe}	F_{in}	804	355_4_4	700	32137	404	451

Table B.12: Results for each system of the sequences generated in the curve section of the train track with velocity $v = 16\ m/s$.

Bibliography

[1] Awwal, A. M., Kumam, P., Abubakar, A. B., Wakili, A., Pakkaranang, N.: *A new hybrid spectral gradient projection method for monotone system of nonlinear equations with convex constraints.* Thai J. Math. 66-88 (2018).

[2] Barzilai, J., Borwein, J.: *Two point step gradient methods.* IMA J. Numer. Anal. **8**, 141-148 (1988).

[3] Birgin, E. G., Krejic, N., Martinez, J. M.: *Globally convergent inexact quasi-Newton methods for solving nonlinear systems.* Numer. Algorithms. **32**, no. 2-4, 249-260 (2003).

[4] Birgin, E. G., Martinez, J. M., Raydan, M.: *Spectral Projected Gradient Methods: review and Perspectives.* J. Stat. Softw. **60(3)** (2014).

[5] Bonettini, S., Zanella, R., Zanni, L.: *A scaled gradient projection method for constrained image deblurring.* Inverse Probl. **25(1)**, 015002-015002 (2009).

[6] Carcasci, C., Marini, L., Morini, B., Porcelli, M.: *A new modular procedure for industrial plant simulations and its reliable implementation.* Energy **94**, 380-390 (2016).

[7] Crisci, S., Ruggiero, V., Zanni, L.: *Steplength selection in gradient projection methods for box-constrained quadratic programs.* Appl. Math. Comput. **356(1)**, 312-327 (2019).

[8] Curtis, A.R., Powell, M.J.D., Reid, J.K.: *On the estimation of sparse Jacobian matrices*, IMA J. Appl. Math., **13**, 117-119 (1974).

[9] Dai, Y. H., Fletcher R.: *Projected Barzilai-Borwein methods for large-scale box-constrained quadratic programming*, Numer. Math. **100**, 21-47 (2005)

[10] Dai, Y. H., Hager, W., W., Schittkowski, K., Zhang, H.: *The cyclic Barzilai-Borwein method for unconstrained optimization.* IMA J. Numer. Anal. **26(3)**, 604-627 (2006).

[11] De Asmundis, R., di Serafino, D., Riccio, F., Toraldo, G.: *On spectral properties of steepest descent methods.* IMA J. Numer. Anal. **33(4)**, 1416-1435 (2013).

[12] Dembo, R.S., Eisenstat, S.C., Steihaug, T.: *Inexact newton methods.* SIAM J. Numer. Anal. **19(2)**, 400-408 (1982).

[13] Dennis Jr., J. E., Moré, J. J.: *A characterization of superlinear convergence and its application to quasi-Newton methods.* Math. Comput. **28**, 549-560 (1974).

[14] Dennis Jr., J. E., Schnabel., R. B.: *Numerical methods for unconstrained optimization and nonlinear equations.* Prentice Hall Series in Computational Mathematics, Prentice Hall, Inc., Englewood Cliffs, NJ (1983).

[15] di Serafino, D., Ruggiero, V., Toraldo, G., Zanni, L.: *On the steplength selection in gradient methods for unconstrained optimization.* Appl. Math. Comput. **318**, 176-195 (2018).

[16] Dolan, E.D., Moré, J.J.: *Benchmarking optimization software with performance profiles.* Mathematical Programming **91**, 201-213 (2002).

[17] Eisenstat, S. C., Walker, H. F.: *Globally convergent inexact Newton methods.* SIAM J. Opt. **4**, 393-422 (1994).

[18] Facchinei, F., Pang, J.S.: *Finite-Dimensional Variational Inequalities and Complementarity Problems, Volume I.* Springer Series in Operations Research, Springer, New York (2003).

[19] Fletcher, R.: *On the Barzilai-Borwein method.* Optimization and control with applications, Appl. Optimizat. **96**, 235-256, Springer, New York (2005).

[20] Frassoldati, G., Zanni, L., Zanghirati, G.: *New adaptive stepsize selections in gradient methods.* J. Ind. Manag. Optim. **4(2)**, 299-312 (2008).

[21] Gasparo, M.: *A nonmonotone hybrid method for nonlinear systems.* Optim. Method Softw. **13**, 79-94 (2000).

[22] Gill, P. E., Murray, W., Wright, M. H.: *Practical Optimization.* Academic Press (1981).

[23] Glunt, W., Hayden, T., L., Raydan, M.: *Molecular conformations from distance matrices.* J. Comput. Chem. **14(1)**, 114-120 (1993).

[24] Golub, G. H., Van Loan, C. F.: *Matrix computations.* Johns Hopkins Series in the Mathematical Sciences **3**, Johns Hopkins University Press, Baltimore, MD (1983).

[25] Gonçalves, M.L.N., Oliveira, F.R.: *On the global convergence of an inexact quasi-Newton conditional gradient method for constrained nonlinear systems.* Numerical Algorithms **84**, 609-631 (2020).

[26] Griewank, A.: *The "global" convergence of Broyden-like methods with a suitable line search.* J. Austral. Math. Soc. Ser. B 28. **1**, 75-92 (1986).

[27] Grippo, L., Lampariello, S., Lucidi, S.: *A nonmonotone linesearch technique for Newton's methods.* SIAM J. Numer. Anal. **23**, 707-716 (1986).

[28] Grippo, L., Sciandrone, M.: *Nonmonotone derivative-free methods for nonlinear equations.* Comput. Optim. Appl. **37**, 297-328 (2007).

[29] Gu, G. Z., Li, D. H., Qi, L., Zhou, S. Z.: *Descend directions of quasi-Newton methods for symmetric nonlinear equations.* SIAM J. Numer. Anal. **40**, 1763-1774 (2002).

[30] Kalker, J.: *Three-Dimensional elastic bodies in rolling contact.* Kluwer Academic Print, Delft (1990).

[31] Kalker, J., Jacobson, B,: *Rolling contact phenomena.* Springer Verlag, Wien (2000).

[32] Kelley, C. T.: *Iterative Methods for optimization.* Frontiers in Applied Mathematics, SIAM (1999).

[33] La Cruz, W., Raydan, M.: *Nonmonotone spectral methods for large-scale nonlinear systems.* Optim. Method Softw. **18**, 583-599 (2003).

[34] La Cruz, W., Martinez, J. M., Raydan, M.: *Spectral residual method without gradient information for solving large-scale nonlinear systems of equations.* Math. Comput. **75**, 1429-1448 (2006).

[35] La Cruz, W.: *A projected derivative-free algorithm for nonlinear equations with convex constraints.* Optim. Method Softw. **29**, 24-41 (2014).

[36] Li, D.H., Fukushima, M.: *A derivative-free line search and global convergence of Broyden-like method for nonlinear equations.* Optim. Method Softw. **13(3)**, 181-201 (2000).

[37] Li, Q., Li, D. H.: *A class of derivative-free methods for large-scale nonlinear monotone equations.* IMA J. Numer. Anal. **31**, 1625-1635 (2011).

[38] Liu, J., Li, S.: *Multivariate spectral dy-type projection method for convex constrained nonlinear monotone equations.* J. Ind. Manag. Optim. **13**, 283-295 (2017).

[39] Lukšan, L. : *Inexact trust region method for large sparse systems of nonlinear equations.* J. Optimiz. Theory App. **81(3)**, 569-590 (1994).

[40] Lukšan, L., Vlček, J.: *Computational experience with globally convergent descent methods for large sparse systems of nonlinear equations.* Optim. Methods Softw. **8**, 201-223 (1998).

[41] Marini, L. , Morini, B., Porcelli, M.: *Quasi-Newton methods for constrained nonlinear systems: complexity analysis and applications.* Comput. Optim. Appl. **71**, 147-170 (2018).

[42] Martinez, J. M.: *Local convergence theory of inexact Newton methods based on structured least change secant updates*. Math. Comp. **55**, 143-167 (1990).

[43] Martinez, J. M, Zambaldi, M..: *An inverse column-updating method for solving large-scale nonlinear systems of equations*. Optim. Methods Softw. **1 (2)**, 129-140 (1992).

[44] Martinez, J. M.: *Practical quasi-Newton methods for solving nonlinear systems*. J. Comput. Appl. Math. **124**, 97-121 (2000).

[45] Meli E., Morini, B., Porcelli, M., Sgattoni, C.: *Solving nonlinear systems of equations via spectral residual methods: stepsize selection and applications*, pp. 1-28, arXiv:2005.05851 (2020).

[46] Mohammad, H., Abubakar A.,B.: *A positive spectral gradient-like method for large-scale nonlinear monotone equations*. Bull Comput. Appl. Math. **5**, 99-115 (2017).

[47] Morini, B., Porcelli, M.: *TRESNEI, a Matlab trust-region solver for systems of nonlinear equalities and inequalities*. Comput. Optim. Appl. **51**, 27-49 (2012).

[48] Morini, B., Porcelli, M., Toint, P.: *Approximate norm descent methods for constrained nonlinear systems*. Math. Comput. **87**, 1327-1351 (2018).

[49] Nocedal, J., Wright, S. J.: *Numerical Optimization*. Math. Comput. **87**. Springer Series in Operations Research (1999).

[50] Ortega, J., Rheinboldt, W.: *Iterative Solution of Nonlinear Equations in Several Variables*. Academic Press, New York (1970).

[51] Papini, A., Porcelli, M., Sgattoni, C.: *On the global convergence of a new spectral residual algorithm for nonlinear systems of equations*. B. Unione Mat. Ital. (2020). 10.1007/s40574-020-00270-5.

[52] Raydan, M.: *Convergence properties of the Barzilai and Borwein Gradient Method*. PhD Thesis, Rice University (1991).

[53] Raydan, M.: *On the Barzilai and Borwein choice of step length for the gradient method*. IMA J. Numer. Anal. **13**, 321-326 (1993).

[54] Raydan, M.: *The Barzilai and Borwein gradient method for the large scale unconstrained minimization problem*. SIAM J. Optimiz. **7**, 26-33 (1997).

[55] Sherman, A. H.: *On Newton-Iterative methods for the solution of systems of nonlinear equations*. SIAM J. Numer. Anal. **15**, 755-771 (1978).

[56] *Simpack Multibody Simulation Software*. Dassault Systemes GmbH.

[57] Yu, Z., Lin, J., Sun, J., Xiao, Y., Liu, L., Li, Z.: *Spectral gradient projection method for monotone nonlinear equations with convex constraints*. Appl. Numer. Math. **59**, 2416-2423 (2009).

[58] Varadhan, R., Gilbert, P. D.: *BB: an R package for solving a large system of nonlinear equations and for optimizing a high-dimensional nonlinear objective function.* J. Stat. Softw. **32 (4)** (2010).

[59] Vollebregt, E. A. H.: *Refinement of Kalker's rolling contact model.* Bracciali, Proceedings of the 8th International Conference on Contact Mechanics and Wear of Rail-Wheel Systems (CM2009), Firenze, 2009.

[60] Vollebregt, E. A. H.: *User guide for CONTACT, Rolling and sliding contact with friction.* Technical Report TR09-03, version v15.1.1 (2015).

[61] Zhang, L., Zhou, W.: *Spectral gradient projection method for solving nonlinear monotone equations.* J. Comput. Appl. Math. **196**, 478-484 (2006).

[62] Zhou, B., Gao, L., Dai, Y. H.: *Gradient methods with adaptive step-sizes.* Comput. Optim. Appl. **35(1)**, 69-86 (2006).