A limited memory Broyden method to solve high-dimensional systems of nonlinear equations

Bart van de Rotten[†] Sjoerd Verduyn Lunel[†*]

Abstract

We propose a modification to the method of Broyden that allows us to solve high dimensional nonlinear equations using limited memory. To solve a system of equations $g(x) = 0$, with $x \in \mathbb{R}^n$, and g a given nonlinearity, quasi-Newton methods use an $(n \times n)$-matrix B_k. This quasi-Newton matrix approximates the Jacobian of g at x_k, where x_k denotes the approximation of the zero x^* in the kth iteration. Limited memory puts severe restrictions on the size of the system that a quasi-Newton method can solve. In order to implement a limited memory approach we use the fact that in the pth iteration the matrix B_p is the result of updating an initial matrix B_0 by p rank-one matrices. This implies that $B_p = B_0 + CD^T$, where C and D are $(n \times p)$-matrices. By reducing the rank of the matrix CD^T every subsequent iteration, it is possible to use $2np$ locations for storing forthcoming quasi-Newton matrices B_k, $k > p$, with $2np \ll n^2$. Numerical simulations with testfunctions of the CUTE collection, cf. [5, 15], and analytical arguments based on [2] show when our limited memory Broyden method behaves as well as the original method of Broyden. To illustrate the importance of this new method in applications, we show that the limited Broyden method gives rise to an efficient algorithm to compute limiting periodic states of reverse flow reactors.

Keywords

Limited memory, large scale, Broyden, quasi-Newton, rank reduction, q-superlinear convergence.

1 Introduction

Periodic chemical processes form a field of major interest in Chemical Reactor Engineering. Examples of such processes are the pressure swing adsorber (PSA), the thermal swing adsorber (TSA), the Reverse Flow Reactor (RFR), and the more recently developed pressure swing reactor (PSR). The state of a chemical reactor, that contains a given periodically forced process, at a certain time is given by the temperature profiles and concentration profiles of the reactants. Starting with an initial state, the reactor generally goes through a transient phase during many periods before converging to a periodic limiting state. This periodic limiting state is also called the cyclic steady state (CSS), and because the reactor operates in this state most of the time, it is interesting to know the dependence of the cyclic steady state on the operating parameters of the reactor.

In [17] and [18] Van Noorden compared several iterative algorithms that accelerate the determination of the CSS of the reactor. He induced that for these type of problems, especially the Newton-Picard method and the method of Broyden were promising. In this paper we extend the study to reduce the memory needed by the iterative algorithms.

In order to investigate the qualitative behavior of the process, one first has to model the process using partial differential equations. The simplest model leads to a partial differential equation with one space dimension, the axial direction and time-dependent boundary conditions. The action of the reactor during one cycle can be computed by discretizing the model in space, and then

[†] *Universiteit Leiden, Mathematical Institute, Niels Bohrweg 1, 2333 CA Leiden, The Netherlands*

[*] Corresponding author. e-mail: verduyn@math.leidenuniv.nl

integrating the obtained system of ordinary differential equations in time for one period. This action can be denoted by the so-called period map, $f : \mathbb{R}^n \to \mathbb{R}^n$, which, in general, is highly nonlinear. The dynamical process in the reactor can now be formulated by a dynamical system

$$x_{k+1} = f(x_k), \qquad k \in \mathbb{N},$$

where x_k denotes the state of the reactor after k periods. The performance of the reactor over one cyclic can be obtained by applying the period map f once. Periodic states of the reactor are fixed points of the period map f and a stable cyclic steady state can be computed by taking the limit of x_k as $k \to \infty$. Since the transient phase of the process might be very long, more efficient methods to find the fixed points of f are needed. Fix points of the map f correspond to zeros of $g : \mathbb{R}^n \to \mathbb{R}^n$ where g is given by

$$g(x) = f(x) - x.$$

Thus it is equivalent to solve

$$g(x) = 0 \qquad \text{for } x \in \mathbb{R}^n. \tag{1.1}$$

Because these equations are nonlinear and the dimension n of the problem might be large, iterative algorithms are needed to approximate a zero of the function g. Iterative algorithms are based on function evaluations of g in the approximations, x_k, of the zero x^*. These function evaluations can be a rather expensive task, and it is generally accepted that the most efficient iterative algorithm to solve Equation (1.1) uses the least number of function evaluations. In case of high temperature fluctuations, it actually turns out to be essential to include a second space dimension in the model, the radial direction from the axis to the wall of the reactor. To obtain an accurate approximation of the periodic state of the reactor, it is necessary to use a fine grid. This implies that the number of equations, n, increases rapidly. The combination of the integration of the system of ordinary differential equations for the evaluation of the function g and a fine grid in the reactor makes it practically impossible to solve (1.1) using classical iterative algorithms.

The standard iterative algorithm is the method of Newton. Let $x_0 \in \mathbb{R}^n$ be an initial estimate in the neighborhood of a zero x^* of g, then Newton's method defines a sequence $\{x_k\}$ in $\mathbb{R}^n$ of approximations of x^* given by

$$x_{k+1} = x_k - J_g^{-1}(x_k)g(x_k), \qquad k \in \mathbb{N}, \tag{1.2}$$

where $J_g(x)$, is the Jacobian of g at the point x. An advantage of the method of Newton is that in a neighborhood of a zero the convergence is quadratic, i.e.,

$$\|x_{k+1} - x^*\| < c\|x_k - x^*\|^2$$

for a certain constant $c > 0$. Since it is not always possible to determine the Jacobian of g analytically, we have to approximate J_g using finite differences, but then the number of function evaluations per Newton iteration step becomes $(n + 1)$. In 1965 Broyden [1] proposed a method that uses only one function evaluation per iteration step instead of $(n + 1)$.

The main idea of Broyden's method is to approximate the Jacobian of g by a matrix B_k. Thus the scheme (1.2) is replaced by

$$x_{k+1} = x_k - B_k^{-1}g(x_k), \qquad k \in \mathbb{N}. \tag{1.3}$$

After every iteration step the Broyden matrix B_k is adapted with a rank-one-matrix. According to the fact that

$$g(x_{k+1}) - g(x_k) \approx J_g(x_{k+1})(x_{k+1} - x_k),$$

the updated Broyden-matrix B_{k+1} is chosen such that it satisfies the equation

$$y_k = B_{k+1}s_k, \tag{1.4}$$

with

$$s_k = x_{k+1} - x_k \qquad \text{and} \qquad y_k = g(x_{k+1}) - g(x_k).$$

Equation (1.4) is called the secant or quasi-Newton equation. Algorithms for which this condition is satisfied, are called quasi-Newton methods. Assuming that B_{k+1} and B_k are identical on the orthogonal complement of the linear space spanned by s_k, the condition in (1.4) results in the following update scheme for the Broyden matrix B_k

$$B_{k+1} = B_k + (y_k - B_k s_k)\frac{s_k^T}{s_k^T s_k} = B_k + \frac{g(x_{k+1})s_k^T}{s_k^T s_k} \tag{1.5}$$

In 1973 Broyden, Dennis and Moré [2] published a proof that the method of Broyden is locally q-superlinearly convergent, i.e.,

$$\lim_{k\to\infty} \frac{\|x_{k+1} - x^*\|}{\|x_k - x^*\|} = 0.$$

A good overview of quasi-Newton methods, and in particular the method of Broyden, can be found in [5, 4, 12].

In 1979 Gay [6] published an fundamental proof that for linear problems the method of Broyden is actually exactly convergent in $2n$ iterations, which implies locally $2n$-step, quadratic convergence for nonlinear problems

$$\|x_{k+2n} - x^*\| < c\|x_k - x^*\|^2,$$

with $c > 0$. This proof of exact convergence was simplified and sharped in 1981 by Gerber and Luk [7] and, independently, by O'Leary in 1995 [19]. In practice, these results imply that the method of Broyden needs more iterations to converge than the method of Newton. However, since for every iteration step only one function evaluation is made, the method of Broyden uses generally much less CPU-time to solve the problem.

Both methods, Newton and Broyden, need to store a $(n \times n)$-matrix, see (1.2) and (1.3). Therefore for high-dimensional systems, this might lead to severe memory constraints. From the early seventies one has tried to reduce the number of storage locations for the iterative methods. Different techniques have appeared for solving large nonlinear problems. To motivate our approach, we point out two known types of reduction.

In 1970 Schubert [22] introduced a quasi-Newton method that takes the sparsity structure of the Jacobian of g into account. In 1977 and 1981 Toint [23, 24] derived a similar procedure for symmetric matrix updating, in case of optimization problems. Unfortunately this approach cannot be applied when the Jacobian is dense or the sparsity structure is not known beforehand. In case of optimization problems one approach of memory reduction is well known: The limited memory BFGS method (L-BFGS), first described by Nocedal [16]. The method is almost identical to the popular BFGS method, cf. [5]. The modification is in the rule for the matrix update only. The BFGS corrections to the quasi-Newton matrix are stored separately and when the available storage is completely used, the oldest correction is deleted to make space for the new one. Many limited memory methods have now been developed, all based on the idea of removing the earliest updates or skipping certain updates, cf. [11, 3, 10]. In addition, applications of the limited memory quasi-Newton updating to other fields are introduced [13].

The problem that we consider is just a general nonlinear equation ($g(x) = 0$), so beforehand it is not known whether the Jacobian of the system in the solution point, $J_g(x^*)$, is either sparse, symmetric or full. In the last case the above mentioned limited memory methods cannot be applied. In this paper we propose a new limited memory method that is related to the limited memory quasi-Newton method of Nocedal and is an adaption of the method of Broyden. In addition to a large reduction of memory used, this method gives more insight in the original method of Broyden, since we investigate the question of how much and which information can be dropped without destroying the property of superlinear termination. The method is called the Broyden Rank Reduction method (BRR), although, the idea can be applied to other quasi-Newton methods as well. To introduce the main idea of the BRR method we first consider a simple example and then return to out application of the RFR.

The map $f : \mathbb{R}^n \to \mathbb{R}^n$ to be considered is a small (take $\varepsilon = 1.0 \cdot 10^{-2}$) quadratic perturbation to two times the identity map,

$$f(x) = \begin{pmatrix} 2x_1 - \varepsilon x_2^2 \\ \vdots \\ 2x_{n-1} - \varepsilon x_n^2 \\ 2x_n \end{pmatrix}$$

The fixed points of f can be found by applying Broyden's method to solve Equation (1.1) with $g(x) = f(x) - x$ upto a certain error

$$\|g(x)\| < 1.0 \cdot 10^{-12},$$

with initial estimate vector $x_0 = (1, \ldots, 1)$. We first choose the dimension $n = 4$. Starting with a trivial initial matrix $B_0 = -I$, the first Broyden matrix is given by, with $c_1 = g(x_1)/|s_0|$ and $d_1 = s_0/|s_0|$,

$$\begin{aligned} B_1 &= B_0 + c_1 d_1^T \\ &= \begin{pmatrix} -0.50996 & 0.49004 & 0.49004 & 0.49499 \\ 0.49004 & -0.50996 & 0.49004 & 0.49499 \\ 0.48994 & 0.48994 & -0.51006 & 0.49489 \\ 0.5025 & 0.5025 & 0.5025 & -0.49242 \end{pmatrix} \end{aligned}$$

Because B_0 doesn't have to be stored, but can be included in the computer code, it is more economical to store the vector c_1 and d_1 instead of the matrix B_1 itself. Applying another update we obtain the second Broyden matrix, with $c_2 = g(x_2)/|s_1|$ and $d_2 = s_1/|s_1|$,

$$\begin{aligned} B_2 &= B_1 + c_2 d_2^T = B_0 + c_1 d_1^T + c_2 d_2^T \\ &= \begin{pmatrix} -0.50933 & 0.49067 & 0.49067 & 0.49564 \\ 0.49067 & -0.50933 & 0.49067 & 0.49564 \\ 0.49052 & 0.49052 & -0.50948 & 0.49549 \\ 0.50817 & 0.50817 & 0.50817 & -0.48661 \end{pmatrix} \end{aligned}$$

Now $4 \cdot 4 = 16$ location are used to store the vector pairs $\{c_1, d_1\}$ and $\{c_2, d_2\}$. In the next iteration step we would need $6 \cdot 4 = 24$ storage locations, which is more than the 16 storage locations for the Broyden matrix itself. Therefore we have to clean up the Broyden matrix first, before we make the next update. Consider for purpose of presentation the first two updates, and combine them to one update matrix Q,

$$Q = c_1 d_1^T + c_2 d_2^T = \begin{pmatrix} 0.49067 & 0.49067 & 0.49067 & 0.49564 \\ 0.49067 & 0.49067 & 0.49067 & 0.49564 \\ 0.49052 & 0.49052 & 0.49052 & 0.49549 \\ 0.50817 & 0.50817 & 0.50817 & 0.51339 \end{pmatrix}$$

Because Q is the sum of two rank one matrices, it has rank less or equal to two, and if we compute the singular value decomposition of Q, see Section 2 for details, we see that Q can be written as

$$Q = \sigma_1 u_1 v_1^T + \sigma_2 u_2 v_2^T$$

with $\{u_1, u_2\}$ and $\{v_1, v_2\}$ orthonormal sets of vectors and

$$\sigma_1 = 1.9853, \qquad \sigma_2 = 5.3551 \cdot 10^{-5}.$$

This suggest that we can remove the singular value σ_2 from the singular value decomposition of Q without destroying the properties of Q severely. We replace the matrix Q by $\widetilde{Q}$ with

$$\widetilde{Q} = Q - \sigma_2 u_2 v_2^T = \sigma_1 u_1 v_1^T = c_1 d_1^T.$$

Here $c_1 = \sigma_1 u_1$ and $d_1 = v_1$. The difference between the original Broyden matrix B_2 and the new reduced one $\widetilde{B}_2$ can be estimated by

$$\|B_2 - \widetilde{B}_2\| = \|B_0 + Q - B_0 - \widetilde{Q}\| = \|\sigma_2 u_2 v_2^T\| = \sigma_2 \|u_2\|\|v_2\| = \sigma_2.$$

After this reduction we can store a new pair of update vectors $c_2 = g(x_3)/\|s_0\|$ and $d_2 = s_2/\|s_2\|$. From now on every iteration step we first remove the singular value σ_2 of Q before computing the new update.

Because no memory difficulties exist for this four dimensional problem. We do the same computations for $n = 100000$ and different values of p, where p denotes the number of update pairs that are stored (in the last paragraph $p = 2$). The results of these computations are listed in Table 1. In the fourth and fifth column of this table the initial error $N_1 = \|g(x_0)\|$ and the final error $N_m = \|g(x_{m-1})\|$ are given, where m denotes the number of function evaluations. The convergence rate is computed by

$$R = \log\left(\frac{N_1}{N_m}\right)\frac{1}{m}.$$

The last column states the maximum of all σ_p's that are removed from the update matrix Q during the process. The results for $p = 1$ are omitted because the BRR process was not converging. If p is larger than 5, the rate of convergence is not increasing. This implies that it is justified to take $p = 5$ in this situation. Note that for $p = 5$ the number of needed storage locations is reduced from $n^2 = 10^{10}$ to $2pn = 10^6$.

Method	n	p	N_1	N_m	m	R	$\sigma_{\max}$
BRRM	100000	10	$0.313 \cdot 10^3$	$0.136 \cdot 10^{-13}$	15	2.51	0.0
BRRM	100000	5	$0.313 \cdot 10^3$	$0.136 \cdot 10^{-13}$	15	2.51	$0.258 \cdot 10^{-5}$
BRRM	100000	4	$0.313 \cdot 10^3$	$0.569 \cdot 10^{-12}$	22	1.54	1.60

Table 1: The performance of the BRR method, for different values of p.

This example is only used as an introduction to the Broyden Rank Reduction method. As could be seen easily, the Jacobian of this function is a bidiagonal matrix, that can be stored using $(2n-1)$ storage location. In the next section we will consider the example of the Reverse Flow Reactor. Here the Jacobian cannot be determined directly and the sparsity structure of the Jacobian is not known beforehand.

We return to our motivation from Chemical Reactor Engineering formulated in the beginning of the introduction. We are looking for periodic states of the reactor. Therefore we consider the model of the reactor developed in [9] with one direction in space. Define $f : \mathbb{R}^n \to \mathbb{R}^n$ to be the associated period map. The state vector x consist of the temperature and the concentration in every gridpoint in axial direction. We use one hundred gridpoints, which implies that $n = 200$. We apply Broyden's method and the BRR method to find a zero of the function $g(x) = f(x) - x$ upto an error of 10^{-10}, i.e.,

$$\|g(x)\| < 10^{-10}.$$

As initial estimate we choose the state in which the reactor is at constant high temperature and filled with reactants only. In dimensionless terms this leads to $x_0 = (1, \ldots, 1)$.

In Table 2 it is clear that the performance of the BRR method is not worse than the method of Broyden. For $p = 15$ the BRR method even has a higher convergence rate. Other values of p are omitted because the results are rather similar. Note that the largest removed singular value becomes bigger if p decreases, as we would expect. The dimension of the problem ($n = 200$) is small enough to apply the method of Broyden directly. In Section 5 we consider a model of the RFR with two space dimensions. Using 100 gridpoints in axial direction and 25 gridpoints in radial direction (perpendicular to the axis of the reactor), the dimension of the problem becomes $n = 5000$. This problem is now too large for Broyden to handle. But as we shall see, the BRR method can still solve the problem.

Method	n	p	N_1	N_m	m	R	$\sigma_{\max}$
Broyden	200	-	6.29	$0.957 \cdot 10^{-10}$	50	0.498	-
BRRM	200	15	6.29	$0.836 \cdot 10^{-10}$	50	0.501	$0.736 \cdot 10^{-2}$
BRRM	200	10	6.29	$0.635 \cdot 10^{-10}$	60	0.422	0.108
BRRM	200	5	6.29	$0.550 \cdot 10^{-10}$	59	0.432	0.422
BRRM	200	4	6.29	$0.894 \cdot 10^{-10}$	66	0.378	0.453
BRRM	200	3	6.29	$0.425 \cdot 10^{-10}$	96	0.268	0.973
BRRM	200	2	6.29	$0.961 \cdot 10^{-10}$	111	0.224	0.868
BRRM	200	1	6.29	$0.106 \cdot 10^{-5}$	201	$0.776 \cdot 10^{-1}$	2.00

Table 2: The performance of the BRR method, for different values of p, applied to the period map of the RFR.

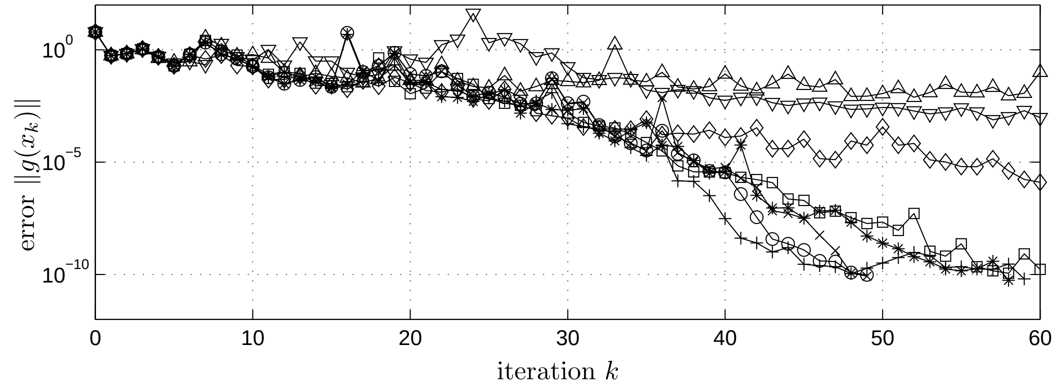


Figure 1: The convergence rate of Broyden and the BRR method applied to the period map of the RFR, for different values of p. ['○' (Broyden), '×' ($p = 15$), '+' ($p = 10$), '∗' ($p = 5$), '□' ($p = 4$), '◇' ($p = 3$), '▽' ($p = 2$), '△' ($p = 1$)]

In Section 2 we describe the BRR method and discuss some properties of this new iterative algorithm. Then, in Section 3 we give a proof of convergence for a perturbed version of Broyden's method, and discuss the consequence of this proof for the BRR method. In Section 4 we check the performance of the BRR method using known and often used testfunctions of the CUTE collection, [15, 5]. We show when BRR method fails to approximate the method of Broyden. In Section 5 we repeat this for the example of the RFR introduced in this section. Additionally, we follow the singular values of the update matrix after applying the reduction step. Finally, in Section 6, we summarize the conclusions.

2 Definition of the Broyden Rank Reduction Method

For the reader convenience we use the notation and terminology from [2].

We consider a function g whose domain $\mathcal{D}$ and range $g(\mathcal{D})$ lie in $\mathbb{R}^n$. For the general system of simultaneous equations $g(x) = 0$, the algorithm that we will study takes the form

$$x_{k+1} = x_k - B_k^{-1} g(x_k) \tag{2.1}$$

where $\{B_k\}$ is generated by the method in such a way that the quasi-Newton equation

$$g(x_{k+1}) - g(x_k) = B_{k+1}(x_{k+1} - x_k)$$

is satisfied at each step. Algorithms of this type are known as quasi-Newton methods. The most well known quasi-Newton method is the 'good' method of Broyden, where the updates are generated by

$$B_{k+1} = B_k + (y_k - B_k s_k)\frac{s_k^T}{s_k^T s_k} = B_k + \frac{g(x_{k+1})s_k^T}{s_k^T s_k}, \tag{2.2}$$

with

$$s_k = x_{k+1} - x_k \qquad \text{and} \qquad y_k = g(x_{k+1}) - g(x_k).$$

Equation (2.2) implies that if an initial matrix B_0 is updated p times, the resulting matrix B_p can be written as follows

$$B_p = B_0 + \sum_{k=0}^{p-1} \frac{g(x_{k+1})s_k^T}{s_k^T s_k}. \tag{2.3}$$

In this paper we choose B_0 to be minus identity ($B_0 = -I$). Three arguments can be given to justify this choice. The matrix B_0 doesn't have to be stored and matrix-vector products with B_0 are trivial. The main reason, however, comes from applications. Consider the example of the RFR where the period map is denoted by $f : \mathbb{R}^n \to \mathbb{R}^n$ and $g(x) = f(x) - x$. It turns out that starting from an initial state, the major step towards the stable periodic limiting state, is made in the first iteration of the period map f. By setting $B_0 = -I$, the first Broyden iteration becomes

$$x_1 = x_0 - B_0^{-1} g(x_0) = x_0 + f(x_0) - x_0 = f(x_0),$$

which is just a dynamical simulation step.

Define $C = [c_1, \ldots, c_p]$ and $D = [d_1, \ldots, d_p]$ by

$$c_k = g(x_{k+1})/\|s_k\| \qquad \text{and} \qquad d_k = s_k/\|s_k\|, \qquad k = 0, \ldots, p-1.$$

If we set

$$Q = CD^T = \sum_{k=1}^{p} c_k d_k^T, \tag{2.4}$$

then $B_p = B_0 + Q$. Therefore instead of B_p, we can store the matrices C and D separately, i.e., we store the first p rank-one-updates. In addition, we take advantage of Equation (2.4) to compute the product Qz for any vector $z \in \mathbb{R}^n$. The following lemma is clear.

Lemma 2.1 *Let $Q = CD^T$, where C and D are arbitrary $(n \times p)$-matrices. Storing the matrices C and D requires $2pn$ storage locations. Furthermore the computation of the matrix vector product Qz, with $z \in \mathbb{R}^n$, costs $2pn$ floating point operations.*

Although we have stored Q using $2np$ storage locations, we would like to reduce this further to $2n(p-1)$. This would allow us to store both vectors of the new update to the Broyden matrix. The question is whether and how this can be done without influencing the Broyden process.

Before proceeding, we recall some basic properties of singular values. Every real matrix $A \in \mathbb{R}^{n\times n}$ can be written as

$$A = U\Sigma V^T = \sigma_1 u_1 v_1^T + \cdots + \sigma_n u_n v_n^T \tag{2.5}$$

where $U = [u_1, \ldots, u_n]$ and $V = [v_1, \ldots, v_n]$ are orthogonal matrices and $\Sigma = \operatorname{diag}(\sigma_1, \ldots, \sigma_n)$. The real nonnegative numbers $\sigma_1 \geq \ldots \geq \sigma_n \geq 0$ are called the singular values of A. Because,

$$\left\{ \begin{array}{l} A^T A v_i = V\Sigma U^T U \Sigma V^T v_i = \sigma_i^2 v_i, \\ AA^T u_i = U\Sigma V^T V \Sigma U^T u + i = \sigma_i^2 u_i, \end{array} \right. \qquad i = 1, \ldots, n,$$

the column vectors of U are the eigenvectors of AA^T and are called the left singular vectors of A, and the column vectors of V are the eigenvectors of $A^T A$ and are called the right singular vectors of A. The rank of a matrix A equals p if and only if σ_p is the smallest positive singular value, i.e., $\sigma_p \neq 0$ and $\sigma_{p+1} = 0$. The following basic theorem yields that the best rank-p approximation of a matrix A, is given by the first p terms of the singular value decomposition (2.5), cf. [8].

Theorem 2.2 *Let the singular value decomposition of $A \in \mathbb{R}^{n \times n}$ be given by* (2.5)*. If $p < r =$ rank A and*

$$A_p = \sum_{k=1}^{p} \sigma_k u_k v_k^T$$

then

$$\min_{\text{rank } B = p} \|A - B\| = \|A - A_p\| = \sigma_{p+1},$$

where $\|.\|$ denotes the l_2-matrix norm.

This theorem can be interpreted as follows: The largest singular values of a matrix A contain the most important information of the matrix A. This leads us to consider the following reduction procedure. Compute the singular value decomposition of the matrix $Q(= B_p - B_0) = U\Sigma V^T$, compare (2.5). Because the rank of Q is less or equal to p, both U and V are orthogonal $(n \times p)$-matrices, and Σ is an $(p \times p)$ diagonal matrix. Next remove the pth singular value and its corresponding left and right singular vectors from the singular values decomposition of Q. The matrix B_p from (2.3) will be replaced by

$$\widetilde{B} = B_p - \sigma_p u_p v_p^T = B_0 + \sum_{l=1}^{p-1} \sigma_l u_l v_l^T.$$

In this way, we free memory to store the newest update vectors. This process is repeated during all subsequent iterations. We are now ready to describe our new algorithm precisely.

Definition 2.3 (The Broyden Rank Reduction Method) *Let $g : \mathbb{R}^n \to \mathbb{R}^n$ be given. Choose x_0 in the neighborhood of x^*, $(g(x^*) = 0)$, and $B_0 \in \mathcal{L}(\mathbb{R}^n)$. Define the function $\Phi : \mathbb{R}^n \times \mathcal{L}(\mathbb{R}^n) \to \mathcal{P}\{\mathcal{L}(\mathbb{R}^n)\}$ by $\Phi(x, B) = \{\bar{B} : s \neq 0\}$, where*

$$\bar{B} = B + (y - Bs)\frac{s^T}{s^T s} - \sigma_p u_p v_p^T (I - \frac{ss^T}{s^T s}), \tag{2.6}$$

with $s = \bar{x} - x$ and $y = g(\bar{x}) - g(x)$ for $\bar{x} = x - B^{-1}g(x)$, and

$$B - B_0 = \sum_{l=1}^{n} \sigma_l u_l v_l^T$$

the singular value decomposition of the update matrix $(B - B_0)$. Then an iteration of the Broyden Rank Reduction Method is defined by

$$\left\{ \begin{array}{l} x_{k+1} = x_k - B_k^{-1} g(x_k), \\ B_{k+1} \in \Phi(x_k, B_k). \end{array} \right. \qquad k = 0, 1, \ldots \tag{2.7}$$

For practical implementation, we use the following notation for the algorithm (2.7)

$$\left\{ \begin{array}{lcl} x_{k+1} & = & x_k - B_k^{-1} g(x_k), \\ \widetilde{B} & = & B_k - \sigma_p u_p v_p^T, \\ B_{k+1} & = & \widetilde{B} + \left(y_k - \widetilde{B} s_k\right) \frac{s_k^T}{s_k^T s_k} \\ & = & \widetilde{B} + \left(g(x_{k+1}) + \sigma_p u_p v_p^T s_k\right) \frac{s_k^T}{s_k^T s_k} \end{array} \right. \qquad k = 0, 1, \ldots \tag{2.8}$$

In fact, the modified update scheme of the Broyden matrix B_k can be explained as follows. Compute the singular value decomposition of the update matrix $Q = U\Sigma V^T$, and define $C = U\Sigma$ and $D = V$. Next, replace the pth columns of C and D by

$$\begin{array}{lcl} c_p & := & \frac{1}{\|s_k\|}(g(x_{k+1}) + c_p d_p^T s_k) \\ d_p & := & \frac{s_k}{\|s_k\|} \end{array}$$

We conclude this section by itemizing several interesting features of the Broyden Rank Reduction method. A strong point of the BRR algorithm is that all $(n \times n)$-matrix manipulations actually can be reduced to computation on $(p \times p)$-matrices. First of all, instead of solving the equation $B_k s_k = -g(x_k)$, we use the equality

$$(B_0 + CD^T)^{-1} = B_0^{-1} - B_0^{-1}C(I + D^T B_0^{-1}C)^{-1}D^T B_0^{-1}. \tag{2.9}$$

We see that we only have to solve a linear equation in the p-dimensional vector space $\mathbb{R}^p$, since $(I + D^T B_0^{-1}C)$ is a $(p \times p)$-matrix.

To compute the singular value decomposition of Q is it not necessary to actually form this $(n \times n)$-matrix. In order to justify this statement we first give the following lemma.

Lemma 2.4 *Let U be an orthogonal $n \times p$-matrix and A an arbitrary $(n \times n)$-matrix. If $Im\,U$ is an invariant subspace of A, then the matrix $A\big|_{Im\,U}$ and $Z = U^T AU$ have the same eigenvalues, counting multiplicities. In particular if $v \in Im\,U$ is an eigenvector of A corresponding to the eigenvalue λ then $w = U^T v$ is an eigenvector of Z corresponding to λ.*

The proof is related to proof that two similar matrices have the same eigenvalues, and therefore omitted here. Using the QR-decomposition of $D = \widetilde{D}R$ we observe that Q can be written as

$$CD^T = C(\widetilde{D}R)^T = CR^T\widetilde{D}^T = \widetilde{C}\widetilde{D}^T,$$

where $\widetilde{D}$ is orthogonal. As a corollary, we now apply Lemma 2.4 with $A = Q^T Q$ and $U = \widetilde{D}$. We have that the p largest singular values of $Q = \widetilde{C}\widetilde{D}^T$ are the square roots of the eigenvalues of

$$\widetilde{D}^T Q^T Q \widetilde{D} = \widetilde{D}^T\widetilde{D}\ \widetilde{C}^T\widetilde{C}\ \widetilde{D}^T\widetilde{D} = \widetilde{C}^T\widetilde{C}.$$

The eigenvectors corresponding to the eigenvalues of this $(p \times p)$-matrix $\widetilde{C}^T\widetilde{C}$ stored in a matrix W yield the singular value decomposition of the update matrix. Note that W is an orthogonal square matrix and therefore

$$(\widetilde{C}W)(\widetilde{D}W)^T = \widetilde{C}WW^T\widetilde{D}^T = \widetilde{C}\widetilde{D}^T = Q.$$

These observations lead to the following clear corollary.

Corollary 2.5 *Let $Q = \widetilde{C}\widetilde{D}^T$, where $\widetilde{C}$ and $\widetilde{D}$ are $(n \times p)$-matrices. Suppose that $\widetilde{D}$ is orthogonal then the p largest singular values of Q, $\sigma_1 \geq \ldots \geq \sigma_p \geq 0$, are the square roots of the eigenvalues of $\widetilde{C}^T\widetilde{C}$. Additionally, if $w_1, \ldots, w_p$ are the eigenvectors of $\widetilde{C}^T\widetilde{C}$ then the corresponding right singular vectors of Q, $v_1, \ldots, v_p$, are given by*

$$v_l = \widetilde{D}w_l, \qquad l = 1, \ldots, p.$$

Furthermore, the BRRM is a quasi-Newton method, since

$$B_{k+1}s_k = \widetilde{B}s_k + y_k - \widetilde{B}s_k = y_k.$$

The error that is introduced by removing the pth singular value of Q equals

$$\begin{aligned} \|B_k - \widetilde{B}\| &= \|B_0 + Q - B_0 - \widetilde{Q}\| \\ &= \|\sigma_p u_p v_p^T\| = \sigma_p\|u_p\|\|v_p\| = \sigma_p \end{aligned}$$

Note that update (2.6) can be applied in the first p iterations of the BRR process as well. Because the smallest singular value σ_p in those iterations is equal zero, the algorithm is exactly the same as the method of Broyden.

Remark 2.6 *The same reduction method can be applied to other quasi-Newton methods. In particular to the inverse notation of the method of Broyden that handles with the approximations H_k to the inverse of the Jacobian. The process then is given by*

$$\begin{cases} x_{k+1} = x_k - H_k g(x_k), \\ \widetilde{H} = H_k - \sigma_p u_p v_p^T, \\ H_{k+1} = \widetilde{H} + (s_k - \widetilde{H}y_k)\frac{s_k^T\widetilde{H}}{s_k^T\widetilde{H}y_k}. \end{cases} \qquad k = 0, 1, \ldots,$$

where $H_k = H_0 + Q$ and rank $Q \leq p$. We call this variant the Broyden Rank Reduction Inverse method. The BRRI method is also a quasi-Newton method and has the advantage that no inverse has to be computed. The second difference is that no inverse has to be computed of a perturbed matrix. In sections 4 and 5 we compare the BRR method with the BRRI method.

Another approach of reduction could be only removing the first column of both matrices, C and D, i.e., the oldest update of the Broyden process.

Example 2.7 *Consider the period map f of the RFR that we mentioned in the introduction. The dimension of the problem is ($n = 200$). We use the method of Broyden to solve*

$$\|g(x)\| < 10^{-10},$$

starting with $B_0 = -I$ and $x_0 = (1, \ldots, 1)$, where after the pth iteration every next iteration the earliest update $\{c_1, d_1\}$ is removed from the sum that forms the update matrix. The new Broyden matrix $\widetilde{B}$ becomes

$$\widetilde{B} = B_1 - c_1 d_1^T = B_0 + \sum_{k=2}^{p} c_k d_k^T.$$

Renumber the update pairs of the Broyden matrix. For different values of p we have plotted the rate of convergence, i.e., the error ($\|g(x_k)\|$) against the iteration. We call this method the Broyden column space reduction method (BCRM).

Method	n	p	N_1	N_m	m	R	$\sigma_{\max}$
Broyden	200	-	6.29	$0.957 \cdot 10^{-10}$	50	0.498	-
BCRM	200	15	6.29	$0.857 \cdot 10^{-10}$	148	0.169	-
BCRM	200	10	6.29	$0.940 \cdot 10^{-10}$	133	0.187	-
BCRM	200	5	6.29	$0.538 \cdot 10^{-10}$	159	0.160	-
BCRM	200	4	6.29	$0.845 \cdot 10^{-10}$	145	0.173	-
BCRM	200	3	6.29	$0.817 \cdot 10^{-10}$	187	0.134	-
BCRM	200	2	6.29	$0.823 \cdot 10^{-10}$	201	0.125	-
BCRM	200	1	6.29	$0.226 \cdot 10^{-6}$	201	$0.853 \cdot 10^{-1}$	-

Table 3: The performance of the BCR method, for different values of p, in case of the period map of the RFR.

It is clear, see Table 3 and Figure 2, that for this problem the BCR method is not as efficient as the method of Broyden.

Instead of removing the smallest singular value we also could remove other singular values from the SVD of the update matrix Q

Example 2.8 *Consider the period map f of the RFR that we mentioned in the introduction. The dimension of the problem is ($n = 200$). We use the method of Broyden to solve*

$$\|g(x)\| < 10^{-10},$$

starting with $B_0 = -I$ and $x_0 = (1, \ldots, 1)$, where after the pth iteration every next iteration the largest singular value of Q is removed. Thus

$$\widetilde{B} = B - \sigma_1 u_1 v_1^T = B_0 + \sum_{k=2}^{p} \sigma_k u_k v_k^T.$$

We call this variant the Broyden First singular value Reduction Method (BFRM)

In Table 4 and Figure 3 we see again that the performance of the BFR method is worse than the performance of Broyden's method.

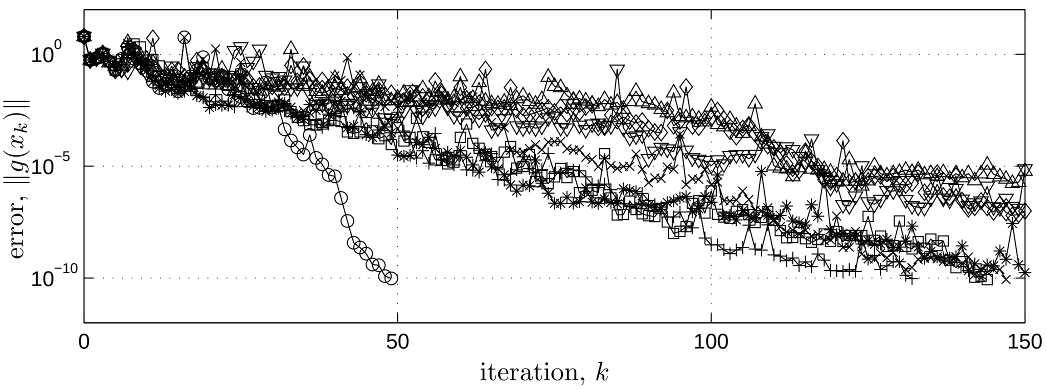


Figure 2: The convergence rate of Broyden and the BCR method in case of the period map of the RFR, for different values of p. ['$\circ$' (Broyden), '$\times$' ($p = 15$), '$+$' ($p = 10$), '$*$' ($p = 5$), '$\square$' ($p = 4$), '$\diamond$' ($p = 3$), '∇' ($p = 2$), '$\triangle$' ($p = 1$)]

Method	n	p	N_1	N_m	m	R	$\sigma_{\max}$
Broyden	200	-	6.29	$0.957 \cdot 10^{-10}$	50	0.498	-
BFRM	200	15	6.29	$0.520 \cdot 10^{-7}$	201	$0.926 \cdot 10^{-1}$	2.36
BFRM	200	10	6.29	$0.169 \cdot 10^{-6}$	201	$0.867 \cdot 10^{-1}$	2.39
BFRM	200	5	6.29	$0.168 \cdot 10^{-5}$	201	$0.753 \cdot 10^{-1}$	2.50
BFRM	200	4	6.29	$0.224 \cdot 10^{-5}$	201	$0.739 \cdot 10^{-1}$	2.06
BFRM	200	3	6.29	$0.421 \cdot 10^{-7}$	201	$0.936 \cdot 10^{-1}$	2.29
BFRM	200	2	6.29	$0.308 \cdot 10^{-5}$	201	$0.723 \cdot 10^{-1}$	2.61
BFRM	200	1	6.29	$0.106 \cdot 10^{-5}$	201	$0.776 \cdot 10^{-1}$	2.00

Table 4: The performance of Broyden and the BFR method in case of the period map of the RFR, for different values of p.

3 Convergence of the Broyden Rank Reduction Method

In this section we investigate the local convergence behavior of the Broyden Rank Reduction method introduced in Section 2. In [2] Broyden, Dennis and Moré have proved that the method of Broyden is locally and q-superlinearly convergent at x^*, the zero of the function g. The results of Lemma 3.1 and Theorem 3.2, taken from [2], are the main ingredients of this proof.

We use two different matrix norms, $\|.\|_2$ and $\|.\|_F$, respectively, the l_2-norm and Frobenius norm. For simplicity we write $\|.\|$ instead of $\|.\|_2$. By $J_g : \mathbb{R}^n \to \mathbb{R}^{n \times n}$ we denote the Jacobian of g in x.

Lemma 3.1 *Assume $g : \mathbb{R}^n \to \mathbb{R}^n$ is differentiable in the open, convex set $\mathcal{D}$, and suppose that for some x^* in $\mathcal{D}$ there exists a $K > 0$ such that*

$$\|J_g(x) - J_g(x^*)\| \leq K\|x - x^*\|, \qquad x \in \mathcal{D}. \tag{3.1}$$

Then, for every u and v in $\mathcal{D}$,

$$\|g(v) - g(u) - J_g(x^*)(v - u)\| \leq K \max\{\|v - x^*\|, \|u - x^*\|\}\|v - u\|. \tag{3.2}$$

Moreover, if $J_g(x^*)$ is invertible, there is an $\varepsilon > 0$ and a $\rho > 0$ such that $\max\{\|v-x^*\|, \|u-x^*\|\} \leq \varepsilon$ implies that u and v belong to $\mathcal{D}$ and

$$(1/\rho)\|v - u\| \leq \|g(v) - g(u)\| \leq \rho\|v - u\|. \tag{3.3}$$

Condition (3.1) means that J_g is Lipschitz in x^*. Equation (3.2) follows from a standard result in multivariable calculus, see [20]. For the inequality in (3.3) we need the continuity and the non-singularity of J_g in x^*, as well as the triangle inequality and (3.2). For more details see [5].

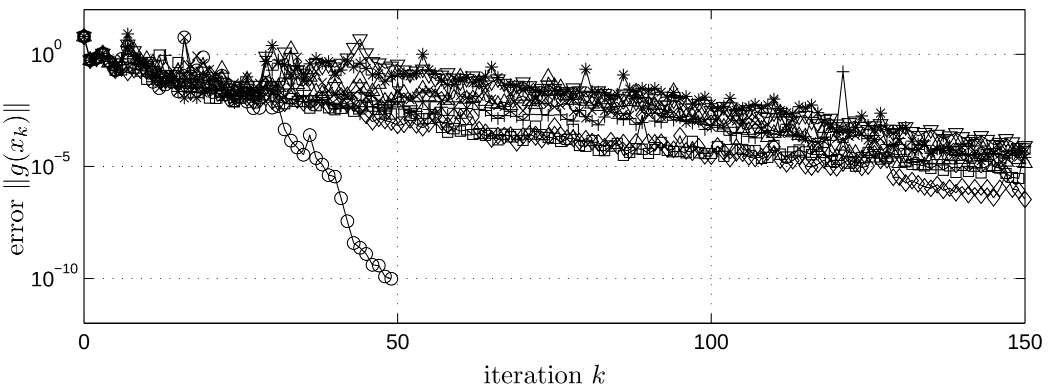


Figure 3: The convergence rate of Broyden's method and the BFR method in case of the period map of the RFR, for different values of p. ['$\circ$' (Broyden), '$\times$' ($p = 15$), '$+$' ($p = 10$), '$*$' ($p = 5$), '$\Box$' ($p = 4$), '$\diamond$' ($p = 3$), '∇' ($p = 2$), '$\triangle$' ($p = 1$)]

Theorem 3.2 *Let $g : \mathbb{R}^n \to \mathbb{R}^n$ be differentiable in the open, convex set $\mathcal{D}$, and assume that for some x^* in $\mathcal{D}$ and $K > 0$ inequality (3.1) holds, where $g(x^*) = 0$ and $J_g(x^*)$ is non-singular. Let $\Phi : \mathbb{R}^n \times \mathcal{L}(\mathbb{R}^n) \to \mathcal{P}\{\mathcal{L}(\mathbb{R}^n)\}$ be defined in a neighborhood $N = N_1 \times N_2$ of $(x^*, J_g(x^*))$ where N_1 is contained in $\mathcal{D}$ and N_2 only contains non-singular matrices. Suppose there are non-negative constants α_1 and α_2 such that for each (x, B) in N, and for $\bar{x} = x - B^{-1}g(x)$,*

$$\|\bar{B} - J_g(x^*)\|_F \leq [1 + \alpha_1 \max\{\|\bar{x} - x^*\|, \|x - x^*\|\}]\|B - J_g(x^*)\|_F + \alpha_2 \max\{\|\bar{x} - x^*\|, \|x - x^*\|\} \quad (3.4)$$

for each $\bar{B}$ in $\Phi(x, B)$. Then for arbitrary $r \in (0, 1)$, there are positive constants $\varepsilon(r)$ and $\delta(r)$ such that for $\|x_0 - x^*\| < \varepsilon(r)$ and $\|B_0 - J_g(x^*)\|_F < \delta(r)$, and $B_{k+1} \in \Phi(x_k, B_k)$, $k \geq 0$, the sequence

$$x_{k+1} = x_k - B_k^{-1} g(x_k) \quad (3.5)$$

is well defined and converges to x^*. Furthermore,

$$\|x_{k+1} - x^*\| \leq r\|x_k - x^*\| \quad (3.6)$$

for each $k \geq 0$, and $\{\|B_k\|\}, \{\|B_k^{-1}\|\}$ are uniformly bounded.

The proof of Theorem 3.2 can be found in [2], and therefore omitted here. As we can see the updated quasi-Newton matrix B_{k+1} should be in the set $\Phi(x_k, B_k)$, where $\Phi : \mathbb{R}^n \times \mathcal{L}(\mathbb{R}^n) \to \mathcal{P}\{\mathcal{L}(\mathbb{R}^n)\}$. In case of the method of Broyden, for example, this function in given by $\Phi(x, B) = \{\bar{B} : s \neq 0\}$, where

$$\bar{B} = B + \frac{(y - Bs)s^T}{s^T s},$$

see (2.2). Broyden, Dennis and Moré [2] indicated that a round-off can be taken into account in which case the function

$$\Phi(x, B) = \{\bar{B} + R : \|R\|_F \leq \|s\|, s \neq 0\} \quad (3.7)$$

may yield a reasonable algorithm. We show, according to [2] and [5], that the algorithm corresponding with the update function (3.7) is also locally q-superlinearly convergent. Thereafter we indicate the consequences of Theorem 3.3 for the BRR method.

Theorem 3.3 *Let $g : \mathbb{R}^n \to \mathbb{R}^n$ be differentiable in the open, convex set $\mathcal{D}$, and assume that for some x^* in $\mathcal{D}$ and $K > 0$ inequality (3.1) holds, where $g(x^*) = 0$ and $J_g(x^*)$ is non-singular. Then the update function $\Phi(x, B) = \{\bar{B} : \|R\|_F < \|s\|, s \neq 0\}$, where*

$$\bar{B} = B + (y - Bs)\frac{s^T}{s^T s} - R\left(I - \frac{ss^T}{s^T s}\right), \quad (3.8)$$

is well defined in a neighborhood $N = N_1 \times N_2$ of $(x^*, J_g(x^*))$, and the corresponding iteration

$$x_{k+1} = x_k - B_k^{-1} g(x_k) \tag{3.9}$$

with $B_{k+1} \in \Phi(x_k, B_k)$, $k \geq 0$, is locally and q-superlinearly convergent at x^*.

Before we prove the theorem we need some preparations. The idea of the proof of Theorem 3.3 is as follows. If $\bar{B}$ is given by (3.8), then Lemma 3.1 and standard properties of the matrix norms, $\|.\|_2$ and $\|.\|_F$ imply that there exists a neighborhood N of $(x^*, J_g(x^*))$ such that condition (3.4) is satisfied for every (x, B) in N. Subsequently Theorem 3.2 yields that iteration (3.9) is locally and linearly convergent. The q-superlinear convergence is a consequence of the following two lemma's of which the proofs can be found in [5].

Lemma 3.4 *Let $\mathcal{D} \subseteq \mathbb{R}^n$ be an open convex set, $g : \mathbb{R}^n \to \mathbb{R}^n$ differentiable, J_g Lipschitz in $x^* \in \mathcal{D}$ and $J_g(x^*)$ non-singular. Let $\{B_k\}$ be a sequence of nonsingular matrices in $\mathcal{L}(\mathbb{R}^n)$, and suppose for some $x_0 \in \mathcal{D}$ that the sequence of points generated by*

$$x_{k+1} = x_k - B_k^{-1} g(x_k)$$

remains in $\mathcal{D}$, and satisfies $x_k \neq x^*$ for any k, and $\lim_{k\to\infty} x_k = x^*$. Then $\{x_k\}$ converges q-superlinearly to x^* in some norm $\|.\|$, and $g(x^*) = 0$, if and only if

$$\lim_{k\to\infty} \frac{\|(B_k - J_g(x^*))s_k\|}{\|s_k\|} = 0 \tag{3.10}$$

where $s_k = x_{k+1} - x_k$.

Lemma 3.5 *Let $s \in \mathbb{R}^n$ be nonzero and $E \in \mathbb{R}^{n\times n}$. Then*

$$\begin{aligned} \left\| E\left(I - \frac{ss^T}{s^Ts}\right)\right\|_F &= \left(\|E\|_F^2 - \left(\frac{\|Es\|}{\|s\|}\right)^2\right)^{1/2} \\ &\leq \|E\|_F - \frac{1}{2\|E\|_F}\left(\frac{\|Es\|}{\|s\|}\right)^2. \end{aligned}$$

We are now able to prove the main theorem of this section.

Proof (of Theorem 3.3) In order to be able use both Theorem 3.2 and Lemma 3.4 we first derive an estimate for $\|\bar{B} - J_g(x^*)\|$. Assume that $\bar{x}$ and x are in $\mathcal{D}$ and $\|s\| \neq 0$. Define $\bar{E} = \bar{B} - J_g(x^*)$, $E = B - J_g(x^*)$, $\bar{e} = \bar{x} - x^*$, and $e = x - x^*$. Note that

$$\begin{aligned} \bar{E} &= \bar{B} - J_g(x^*) \\ &= B - J_g(x^*) + (y - Bs)\frac{s^T}{s^Ts} - R\left(I - \frac{ss^T}{s^Ts}\right) \\ &= (B - J_g(x^*))\left(I - \frac{ss^T}{s^Ts}\right) + (y - J_g(x^*))\frac{s^T}{s^Ts} - R\left(I - \frac{ss^T}{s^Ts}\right). \end{aligned}$$

Therefore,

$$\begin{aligned} \|\bar{E}\|_F &\leq \left\|(B - J_g(x^*))\left(I - \frac{ss^T}{s^Ts}\right)\right\|_F + \frac{\|y - J_g(x^*)s\|}{\|s\|} + \left\|R\left(I - \frac{ss^T}{s^Ts}\right)\right\|_F \\ &\leq \left\|E\left(I - \frac{ss^T}{s^Ts}\right)\right\|_F + K\max\{\|\bar{e}\|, \|e\|\} + \left\|R\left(I - \frac{ss^T}{s^Ts}\right)\right\|_F. \end{aligned} \tag{3.11}$$

For the last step Lemma 3.1 is used. Because $\left(I - \frac{ss^T}{s^Ts}\right)$ is an orthogonal projection and $\|s\| \leq 2\max\{\|\bar{e}\|, \|e\|\}$ inequality (3.11) can be reduced to

$$\begin{aligned} \|\bar{E}\|_F &\leq \|E\|_F \left\|\left(I - \frac{ss^T}{s^Ts}\right)\right\| + K\max\{\|\bar{e}\|, \|e\|\} + \|R\|_F \left\|\left(I - \frac{ss^T}{s^Ts}\right)\right\| \\ &\leq \|E\|_F + (K+2)\max\{\|\bar{e}\|, \|e\|\}. \end{aligned} \tag{3.12}$$

Define

$$N_2 = \left\{ B \in \mathcal{L}(\mathbb{R}^n) : \|J_g(x^*)^{-1}\| \|B - J_g(x^*)\| < \frac{1}{2} \right\}.$$

Then any $B \in N_2$ is non-singular and satisfies

$$\|B^{-1}\| \leq 2\|J_g(x^*)^{-1}\|.$$

To define N_1 choose $\varepsilon > 0$ and $\rho > 0$ as in Lemma 3.1 so that $\max\{\|\bar{x} - x^*\|, \|x - x^*\|\} \leq \varepsilon$ implies that x and $\bar{x}$ belong to $\mathcal{D}$ and that (3.3) holds. In particular, if $\|x - x^*\| \leq \varepsilon$ and $B \in N_2$ then $x \in \mathcal{D}$ and

$$\|s\| = \|B^{-1}g(x)\| \leq \|B^{-1}\| \|g(x) - g(x^*)\| \leq 2\rho\|J_g(x^*)^{-1}\| \|x - x^*\|.$$

Now define N_1 as the set of all $x \in \mathbb{R}^n$ such that $\|x - x^*\| < \frac{\varepsilon}{2}$ and

$$2\rho\|J_g(x^*)^{-1}\| \|x - x^*\| < \frac{\varepsilon}{2}.$$

If $N = N_1 \times N_2$ and $(x, B) \in N$, then

$$\|\bar{x} - x^*\| \leq \|s\| + \|x - x^*\| \leq \varepsilon.$$

Hence, $\bar{x} \in \mathcal{D}$ and moreover, (3.3) shows that $s = 0$ if and only if $x = x^*$. The update function is well defined in N. Equation (3.12) then shows that the update function associated with iteration (3.9) satisfies the hypotheses of Theorem 3.2 are therefore, algorithm (3.9) is locally convergent at x^*. In addition we can choose $r \in (0, 1)$ in (3.6) arbitrary. We take $r = \frac{1}{2}$, thus

$$\|x_{k+1} - x^*\| \leq \frac{1}{2}\|x_k - x^*\| \tag{3.13}$$

Considering Lemma 3.4, a sufficient condition for $\{x_k\}$ to converge q-superlinearly to x^* is

$$\lim_{k\to\infty} \frac{\|E_k s_k\|}{\|s_k\|} = 0. \tag{3.14}$$

In order to justify Equation (3.14) we write Equation (3.11) as

$$\|E_{k+1}\|_F \leq \left\| E_k \left(I - \frac{s_k s_k^T}{s_k^T s_k} \right) \right\|_F + (K + 2)\max\{\|e_{k+1}\|, \|e_k\|\}. \tag{3.15}$$

Using Equation (3.13) and Lemma 3.5 in (3.15), gives

$$\|E_{k+1}\|_F \leq \|E_k\|_F - \frac{\|E_k s_k\|^2}{2\|E_k\|_F \|s_k\|^2} + (K + 2)\|e_k\|,$$

or

$$\frac{\|E_k s_k\|^2}{\|s_k\|^2} \leq 2\|E_k\|_F \left(\|E_k\|_F - \|E_{k+1}\|_F + (K + 2)\|e_k\| \right). \tag{3.16}$$

Theorem 3.2 gives that $\{\|B_k\|\}$ is uniformly bounded, $k \geq 0$. This implies that there exists an $M > 0$ independently of k such that

$$\|E_k\| = \|B_k - J_g(x^*)\| \leq \|B_k\| + \|J_g(x^*)\| \leq M. \tag{3.17}$$

By Equation (3.13)

$$\sum_{k=0}^{\infty} \|e_k\| \leq 2\varepsilon.$$

Thus from (3.16),

$$\frac{\|E_k s_k\|^2}{\|s_k\|^2} \leq 2M\left(\|E_k\|_F - \|E_{k+1}\|_F + (K+2)\|e_k\|\right), \tag{3.18}$$

and summing the left and right sides of (3.18) for $k = 0, 1, \ldots, m$,

$$\begin{aligned}\sum_{k=0}^{m} \frac{\|E_k s_k\|^2}{\|s_k\|^2} &\leq 2M\left(\|E_0\|_F - \|E_{m+1}\|_F + (K+2)\sum_{k=0}^{m}\|e_k\|\right)\\ &\leq 2M\left(\|E_0\|_F + 2\varepsilon(K+2)\right)\\ &\leq 2M\left(M + 2\varepsilon(K+2)\right). \end{aligned}\tag{3.19}$$

Since (3.19) is true for any $m \geq 0$, we have

$$\sum_{k=0}^{\infty} \frac{\|E_k s_k\|^2}{\|s_k\|^2} < \infty.$$

This implies (3.14) and completes the proof. □

In case of the BRR method the round-off matrix R in Theorem 3.3 is dependent on k and given by the pth term in the singular value decomposition of the update matrix $(B_{k-1} - B_0)$, $k \geq 0$. Thus in the kth iteration of the BRR method, the singular value decomposition of the update matrix is given by

$$B_{k-1} - B_0 = U\Sigma V^T = \sum_{l=1}^{p} \sigma_l u_l v_l^T,$$

where U and V are orthogonal and $\sigma_1 \geq \ldots \geq \sigma_p \geq 0$. Then, the round-off of the kth update becomes $R_{k-1} = \sigma_p u_p v_p^T$. Note that

$$\|R_{k-1}\|_F = \sigma_p \|u_p v_p^T\|_F = \sigma_p \|u_p\|\|v_p\| = \sigma_p.$$

As long as the removed singular value σ_p is smaller than the size of the current step, i.e.,

$$\sigma_p \leq \|s_{k-1}\|, \qquad k \in \mathbb{N} \tag{3.20}$$

the condition that $\|R_{k-1}\|_F \leq \|s_{k-1}\|$ is fulfilled and the algorithm is locally and q-superlinearly convergent. Simulations show, however, that after several initial iterations of the BRR method the pth singular value of the update matrix remains more or less of the same size. If the process converges, the step size $\|s_{k-1}\|$ approaches zero. Unless $\sigma_p \equiv 0$ for all $k \in \mathbb{N}$ there exits a smallest $k_0 \in \mathbb{N}$ such that (3.20) is violated. This implies that the local and q-superlinear convergence of the BRR method cannot be proved. However, the larger the number p is taken, the more iterations can be made with (3.20) satisfied. In many applications, a reasonable number p can be chosen such that k_0 is large enough for the process to converge to the zero x^* upto the wanted precision, see Section 4. But, as can be seen in Section 5, it is also possible that the algorithm has still good performance after the k_0th iteration. This lead us to the following conclusion.

Theorem 3.3 can also be formulated using the inverse notation of the method of Broyden. The proof is only slightly different and therefore omitted here.

Theorem 3.6 *Let $g : \mathbb{R}^n \to \mathbb{R}^n$ be differentiable in the open, convex set $\mathcal{D}$, and assume that for some x^* in $\mathcal{D}$ and $K > 0$ inequality* (3.1) *holds, where $g(x^*) = 0$ and $J_g(x^*)$ is non-singular. Then the update function $\Phi(x, H) = \{\bar{H} : \|R\|_F < \|s\|, s \neq 0\}$, where*

$$\bar{H} = H + (s - Hy)\frac{s^T H}{s^T H y} - R\left(I - \frac{ss^T}{s^T s}\right),$$

is well defined in a neighborhood $N = N_1 \times N_2$ of $(x^*, J_g^{-1}(x^*))$, and the corresponding iteration

$$x_{k+1} = x_k - H_k g(x_k)$$

with $H_{k+1} \in \Phi(x_k, H_k)$, $k \geq 0$, is locally and q-superlinearly convergent in x^*.

4 Testfunctions

This section is devoted to a discussion of the behavior of the Broyden Rank Reduction method and the Broyden Rank Reduction Inverse method when applied to a number of test cases. For every testfunction, we consider both methods for different values of p, the number of stored update pairs. If possible we compare the results with those of Broyden's method itself. Because Broyden's method is not globally converging and the area of convergence can be small, we have chosen some specific testfunctions, taken from the CUTE collection, cf. [15, 5]. The performance of the methods are summarized for each test problem in a table. Every time the method of Broyden or the BRR(I) method was applied, we stated the initial error N_1, the final error N_m, and the number of iterations needed to converge, m. The rate of convergence R is defined as $R = \log(N_1/N_m)/m$, to obtain a measure to compare the methods. Finally $\sigma_{\max}$ denotes the maximum value of all singular values σ_p that are removed during the process.

4.1 Discrete integral equation function

The nonlinear integral equation

$$u(t) + \int_0^1 H(s,t)(u(s)+s+1)^3 ds = 0$$

where

$$H(s,t) = \begin{cases} s(1-t), & s < t, \\ t(1-s), & s \geq t, \end{cases}$$

can be discretized by considering the equation at the points $t = t_i, i = 1, \dots, n$, and then replacing the integral by an n-point rectangular rule based on the points $\{t_i\}$. Denoting $h = 1/(n+1)$ and $t_i = i \cdot h, i = 1, \dots, n$, the resulting system of equations is defined by

$$g_i(x) \equiv x_i + \frac{h}{2}\left((1-t_i)\sum_{j=1}^{i} t_j(x_j+t_j+1)^3 + t_i \sum_{j=i+1}^{n}(1-t_j)(x_j+t_j+1)^3\right) = 0, \qquad i = 1, \dots, n,$$

where $x = (x_i), x_i = u(t_i), i = 1, \dots, n$. This function, the so-called Discrete integral equation function, was first used as a testfunction by Moré and Cosnard [14] to test the methods of Brent and of Brown. Remark that the Jacobian of this function has a dense structure.

We start with the initial vector x_0, given by

$$x_i = t_i(t_i - 1), \qquad i = 1, \dots, n.$$

We search for a zero of the function g, more precisely, a solution to the inequality

$$\|g(x)\| < 10^{-10}.$$

If the dimension of the system is not too large ($n = 100$), we can apply Broyden's method. The process converges within 20 iterations. We choose $n = 10,000$ and apply the BRR method for different values of p. It seems that we don't have to choose p larger than 7; for $p > 7$ the rate convergence is not higher than for $p = 7$, cf. Table 5. If we choose $p \leq 6$, the process doesn't converge within 200 iterations. In the last column of the table, we see that for $p = 8$ the largest removed singular value during the process is equal to $\sigma_{\max} = 1.42$. For $p = 9$ this value is much smaller.

In Figure 4(a) we have plotted the error, $\|g(x_k)\|$, of the BRR process for the first 30 iterations, for $p = 1, \dots, 8$. One by one the processes leave the trajectory of the BRR process with $p = 8$, and then slowly diverges. Before the process for $p = 7$ can be disturbed, it is converged. In the second part of the figure, we have plotted the quotient $\sigma_p/\|s_{k-1}\|$ for the same values of p. For every p this quotient increases until it has reached a certain maximum, where it stays more or less

Method	n	p	N_1	N_m	m	R	$\sigma_{\max}$
BRRM	100000	10	$0.238 \cdot 10^2$	$0.157 \cdot 10^{-10}$	22	1.27	$0.127 \cdot 10^{-7}$
BRRM	100000	9	$0.238 \cdot 10^2$	$0.158 \cdot 10^{-10}$	22	1.27	$0.218 \cdot 10^{-5}$
BRRM	100000	8	$0.238 \cdot 10^2$	$0.158 \cdot 10^{-10}$	22	1.27	1.42
BRRM	100000	7	$0.238 \cdot 10^2$	$0.118 \cdot 10^{-10}$	22	1.29	1.73

Table 5: The performance of the BRR method, for different values of p, in case of the Discrete integral equation function.

constant. For every p we see that the moment where σ_p is no longer smaller than $\|s_{k-1}\|$ is just one or two iterations before the process diverges. The larger we choose p, the more iterations the BRR process can make before $\sigma_p > \|s_{k-1}\|$.

The same computations have been done for the BBRI method, see Table 6. We observe the same behavior. For $p > 7$ we have the same rate of convergence as for $p = 7$. If $p \leq 6$ then the rate of convergence is lower. But unlike the results of the BRR method, here the process is still converging for $p \leq 6$.

Method	n	p	N_1	N_m	m	R	$\sigma_{\max}$
BRRIM	100000	10	$0.238 \cdot 10^2$	$0.134 \cdot 10^{-10}$	22	1.28	0.0
BRRIM	100000	9	$0.238 \cdot 10^2$	$0.133 \cdot 10^{-10}$	22	1.28	$0.220 \cdot 10^{-5}$
BRRIM	100000	8	$0.238 \cdot 10^2$	$0.132 \cdot 10^{-10}$	22	1.28	1.56
BRRIM	100000	7	$0.238 \cdot 10^2$	$0.160 \cdot 10^{-10}$	22	1.27	1.62
BRRIM	100000	6	$0.238 \cdot 10^2$	$0.292 \cdot 10^{-10}$	24	1.14	1.63
BRRIM	100000	5	$0.238 \cdot 10^2$	$0.764 \cdot 10^{-10}$	51	0.519	1.78
BRRIM	100000	4	$0.238 \cdot 10^2$	$0.563 \cdot 10^{-11}$	117	0.248	1.82

Table 6: The performance of the BRRI method, for different values of p, in case of the Discrete integral equation function.

We conclude that in this situation we can choose $p = 7$ without losing the high rate of convergence of the Broyden method. This implies that we obtain a reduction of $n^2 = 10^{10}$ to $2pn = 1.4 \cdot 10^6$ storage locations to store the Broyden matrices of our scheme.

4.2 Discrete boundary value function

In the same article [14], Moré and Cosnard also considered the Discrete boundary value function. This function is obtained by applying the standard $\mathcal{O}(h^2)$ discretization to the two-point boundary value problem

$$u''(t) = \frac{1}{2}(u(t) + t + 1)^3, \qquad 0 < t < 1, \quad u(0) = u(1) = 0.$$

Denoting $h = 1/(n+1)$ and $t_i = i \cdot h, i = 1, \ldots, n$, the resulting system of equations is defined by

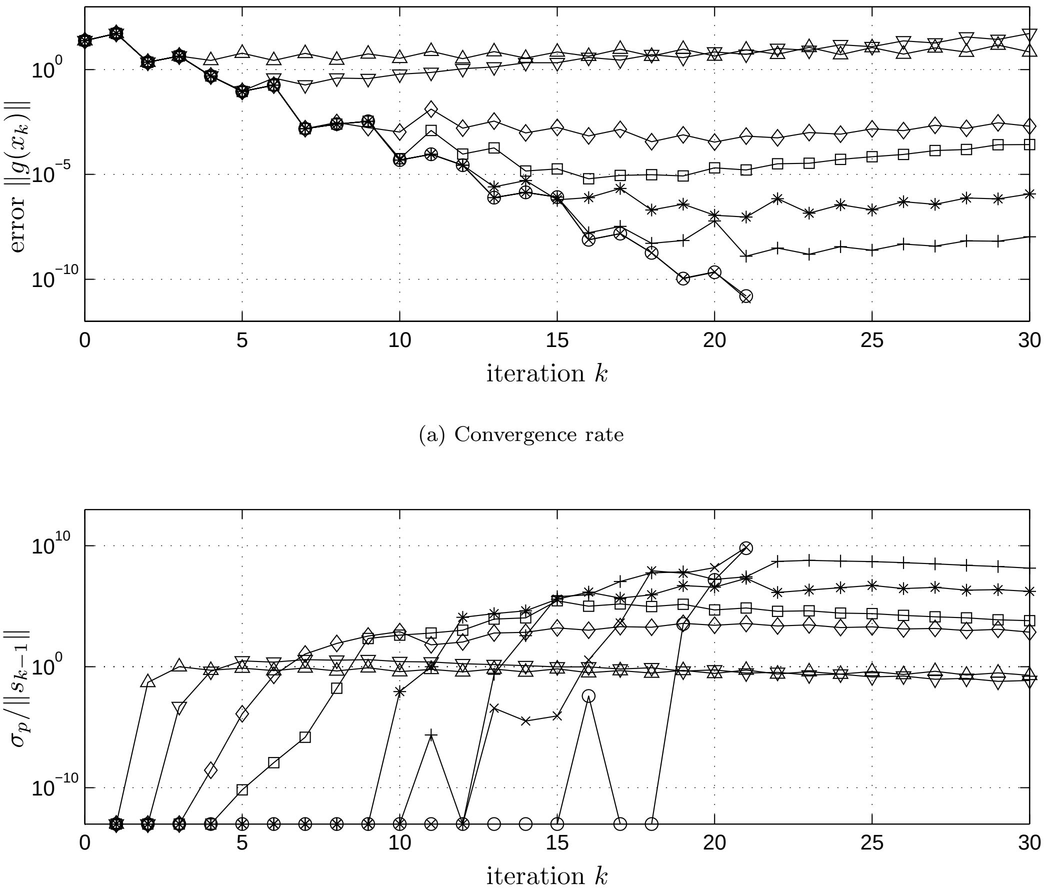
$$g_i(x) = 2x_i - x_{i-1} - x_{i+1} + \frac{h^2}{2}(x_i + t_i + 1)^3, \qquad i = 1, \ldots, n,$$

where $x = (x_i), x_i = u(t_i), i = 1, \ldots, n$. The Jacobian of this function has a band structure with on both subdiagonals the element -1. As in Section 4.1 we start with the initial vector x_0, given by

$$x_i = t_i(t_i - 1), \qquad i = 1, \ldots, n,$$

and try to approximate a zero of the function g, which has to satisfy the inequality

$$\|g(x)\| < 10^{-12}. \tag{4.1}$$



(a) Convergence rate

(b) Quotient of σ_p and $\|s_{k-1}\|$

Figure 4: The convergence rate of the BRR method, in case of the Discrete integral equation function and additionally the quotient $\sigma_p/\|s_{k-1}\|$, for different values of p. ['$\circ$' (p = 8), '$\times$' (p = 7), '$+$' (p = 6), '$*$' (p = 5), '$\Box$' (p = 4), '$\diamond$' (p = 3), '$\triangledown$' (p = 2), '$\triangle$' (p = 1)]

It turns out that Broyden's method can only be applied to solve (4.1) if n is small. For $n = 12$, Broyden's method takes 35 iterations to converge. If we consider the last update matrix $(B_{35} - B_0)$ of the Broyden process we see that all singular values are larger than one. This implies that the update matrix has full rank. For the Discrete integral equation function we have seen that only relatively small singular values ($\sigma_p < \|s_{k-1}\|$) can be removed. Table 7 shows that the BRR method and the BRRI method have difficulties in solving (4.1). For $p < 12$ both the BRR method as the BRRI method diverge. Remark that in case of $p = n$, every iteration step still one singular value is removed. Therefore the results of the BRR method and Broyden's method don't have to be identical. To support the hypotheses that $\sigma_p < \|s_{k-1}\|$ is a strong condition for a good performance of the BRR method, we have plotted in Figure 5 the error and the quotient $\sigma_p/\|s_{k-1}\|$ during the first 30 iterations, for several values of p.

4.3 Extended Rosenbrock function

Let n be even and define the function $g : \mathbb{R}^n \to \mathbb{R}^n$ by

$$\begin{cases} g_{2i-1}(x) = 10(x_{2i} - x_{2i-1}^2), \\ g_{2i}(x) = 1 - x_{2i-1}, \end{cases} \qquad i = 1, \dots, n/2.$$

Method	n	p	N_1	N_m	m	R	$\sigma_{\max}$
Broyden	12	-	0.140	$0.352 \cdot 10^{-12}$	36	0.742	-
BRRM	12	12	0.140	$0.190 \cdot 10^{-13}$	57	0.520	1.0
BRRIM	12	12	0.140	$0.685 \cdot 10^{-12}$	110	0.237	1.25

Table 7: The performance of Broyden's method and the BRR method and BRRI method for $p = n$, in case of the Discrete boundary value function

This implies that the equation

$$g(x) = 0 \tag{4.2}$$

is just $n/2$ copies of a system in the two dimensional space. The structure of the equation allows us to solve parts of the system simultaneously which accelerates the computation. Nevertheless we will use the BRR method to find a vector $x \in \mathbb{R}^n$ such that

$$\|g(x)\| < 10^{-10}.$$

Note that the Jacobian of the Extended Rosenbrock function is a block diagonal matrix. The initial vector x_0 is given by

$$\begin{cases} x_{2i-1} = -1.2, \\ x_{2i} = 1, \end{cases} \qquad i = 1, \ldots, n/2.$$

The dimension of the problem is chosen $n = 100,000$. The results of the BRR method are given in Table 8. It turns out that for $p > 3$ the BRR method shows the same behavior as for $p = 3$. For $p = 2$ and $p = 1$ the rate of convergence is decreasing. Note that for $p = 1$ the maximum of all removed singular values from the update matrix is large. Still the process converges.

Method	n	p	N_1	N_m	m	R	$\sigma_{\max}$
BRRM	100000	10	$0.110 \cdot 10^4$	$0.720 \cdot 10^{-11}$	12	2.72	0.0
BRRM	100000	3	$0.110 \cdot 10^4$	$0.720 \cdot 10^{-11}$	12	2.72	0.0
BRRM	100000	2	$0.110 \cdot 10^4$	$0.681 \cdot 10^{-10}$	30	1.01	1.15
BRRM	100000	1	$0.110 \cdot 10^4$	$0.348 \cdot 10^{-11}$	440	$0.759 \cdot 10^{-1}$	$0.116 \cdot 10^6$

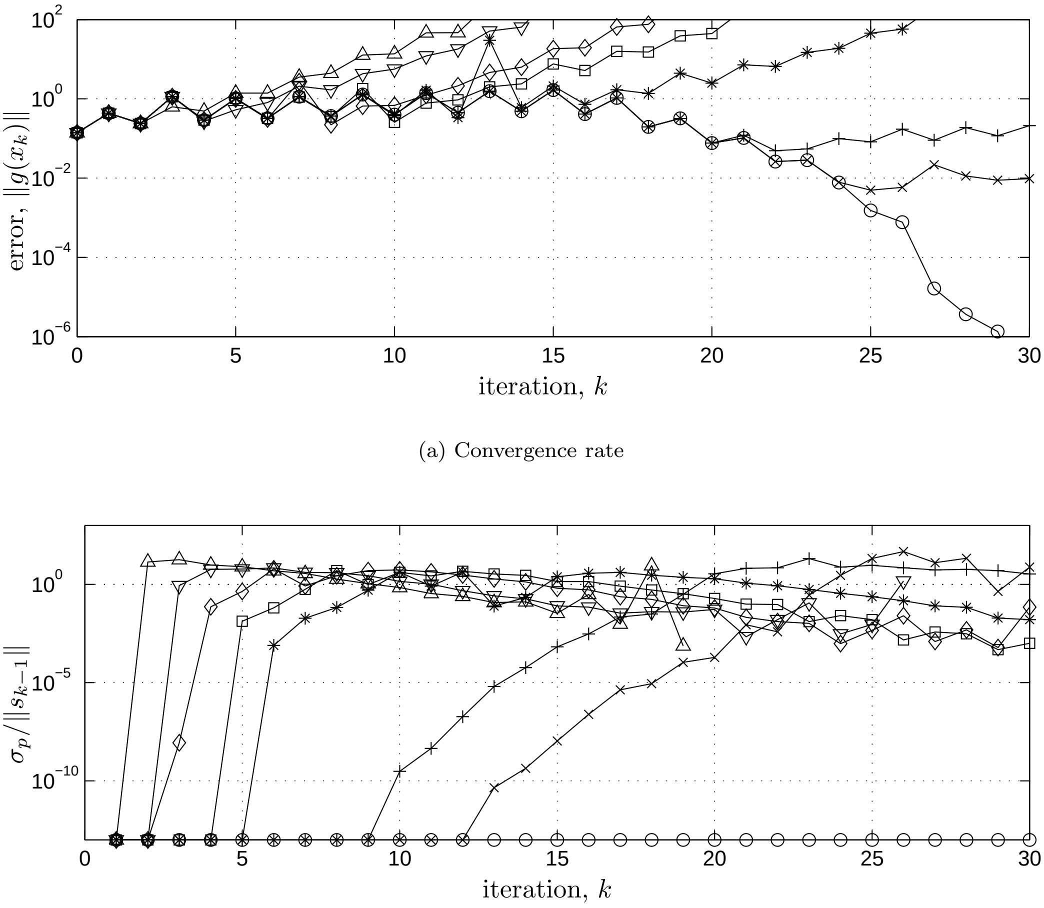
Table 8: The performance of the BRR method in case of the Extended Rosenbrock function, for different values of p.

In Figure 6 again we plotted the error $\|g(x_k)\|$ and the quotient $\sigma_p/\|s_{k-1}\|$ during the first 30 iterations for $p = 1, 2, 3$. The connection between both graphs is not as strong as in sections 4.1 and 4.2, but still we can see that in case of $p = 1$ and $p = 2$ the quotient $\sigma_p/\|s_{k-1}\|$ increases while for $p \geq 3$ it does not. For $p \geq 4$ the graphs are identical to the one of $p = 3$.

We performed the same computations using the BRRI method. The results are stated in Table 9. The only important difference is that the rate of convergence in case of ($p = 1$) is much higher than for the BRR method.

Method	n	p	N_1	N_m	m	R	$\sigma_{\max}$
BRRIM	100000	10	$0.110 \cdot 10^4$	$0.293 \cdot 10^{-10}$	13	2.40	0.0
BRRIM	100000	3	$0.110 \cdot 10^4$	$0.919 \cdot 10^{-10}$	13	2.32	0.0
BRRIM	100000	2	$0.110 \cdot 10^4$	$0.156 \cdot 10^{-10}$	35	0.911	1.02
BRRIM	100000	1	$0.110 \cdot 10^4$	$0.413 \cdot 10^{-10}$	60	0.515	$0.523 \cdot 10^2$

Table 9: The performance of the BRRI method, for different values of p, in case of the Extended Rosenbrock function.



(a) Convergence rate

(b) Quotient of σ_p and $\|s_{k-1}\|$

Figure 5: The convergence rate of the BRR method in case of the Discrete boundary value function and additionally the quotient $\sigma_p/\|s_{k-1}\|$, for different values of p. ['$\circ$' (Broyden), '$\times$' (p = 12), '$+$' (p = 10), '$*$' (p = 5), '$\Box$' (p = 4), '$\diamond$' (p = 3), '$\triangledown$' (p = 2), '$\triangle$' (p = 1)]

The tables and figures in this section show that only two singular values are significant. Therefore we can choose ($p = 3$), which implies a reduction from $n^2 = 10^{10}$ to $2pn = 6 \cdot 10^5$ storage locations.

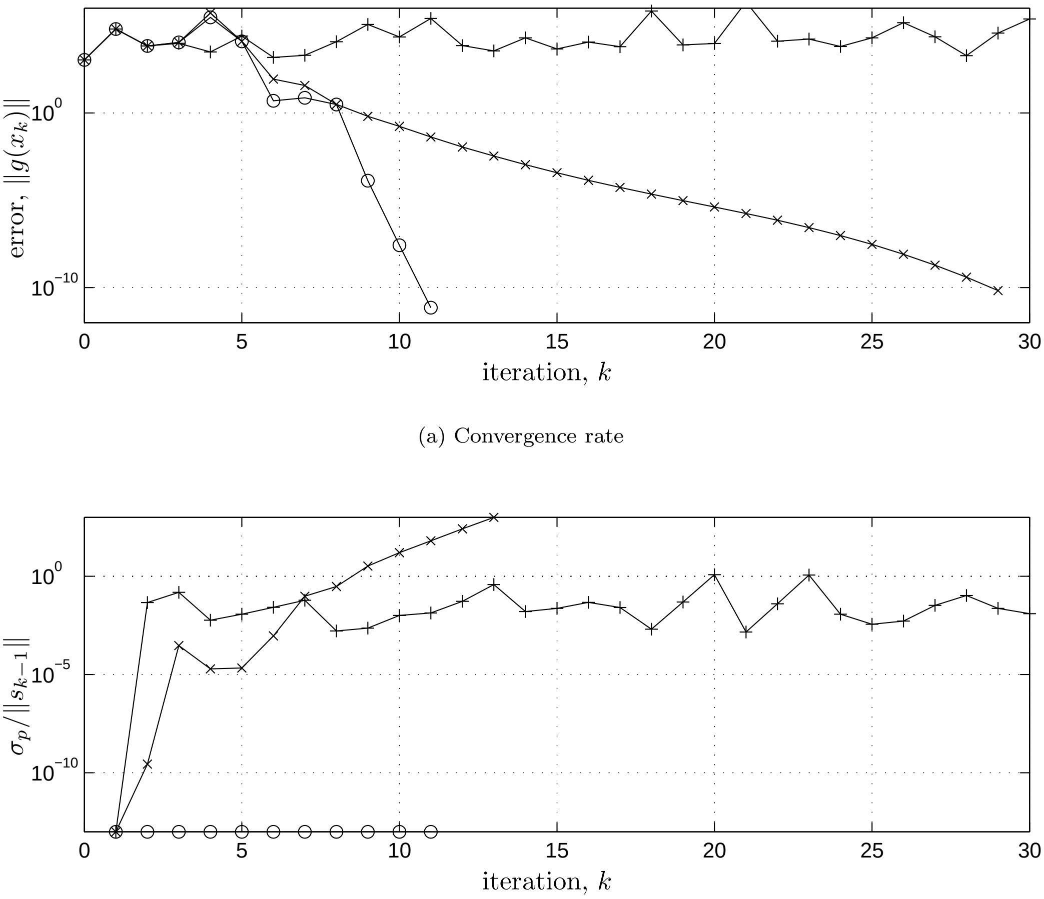
4.4 Extended Powell singular function

Similarly to the Extended Rosenbrock function, the Extended Powell singular function contains $n/4$ copies of the same function in the four-dimensional space. Let n be a multiple of 4 and define the function $g : \mathbb{R}^n \to \mathbb{R}^n$ by

$$\begin{cases} g_{4i-3}(x) = x_{4i-3} + 10x_{4i-2}, \\ g_{4i-2}(x) = \sqrt{5}(x_{4i-1} - x_{4i}), \\ g_{4i-1}(x) = (x_{4i-2} - 2x_{4i-1})^2, \\ g_{4i}(x) = \sqrt{10}(x_{4i-3} - x_{4i})^2, \end{cases} \qquad i = 1, \ldots, n/4.$$

Again we want to approximate a zero of g with a vector $x \in \mathbb{R}^n$ such that

$$\|g(x)\| < 10^{-10}. \tag{4.3}$$



(a) Convergence rate

(b) Quotient of σ_p and $\|s_{k-1}\|$

Figure 6: The convergence rate of the BRR method in case of the Extended Rosenbrock function and additionally the quotient $\sigma_p/\|s_{k-1}\|$, for different values of p. ['$\circ$' (p = 3), '$\times$' (p = 2), '+' (p = 1)]

The initial point x_0 is given by

$$\begin{cases} x_{2i-3} = 3, \\ x_{2i-2} = -1, \\ x_{2i-1} = 0, \\ x_{2i} = 1, \end{cases} \qquad i = 1, \ldots, n/4.$$

We have to remark that for low dimensions it is not possible to solve inequality (4.3) using the method of Broyden, where the BRR method does solve the problem. For $n = 100,000$ the results of the BRR method are put in Table 10. The number of BRR iterations seems to be arbitrary. Note that no large singular values are removed for $p \geq 5$. For $p \geq 9$ more or less the same behavior is observed. If $p = 4$ the process is not converging within 1000 iterations. For $p \leq 3$ the process even diverges. This would imply that four singular values are significant. If we consider the rate of convergence of the BRR method for different values of p we observe that the process passes a meta stable phase, cf. Figure 7. It is remarkable that for every $p \geq 5$ the process reaches the meta stable phase at the same iteration of the BRR process, but leaves the plateau at different instances, not monotone in p.

The graph of the quotient $\sigma_p/\|s_{k-1}\|$ is not interesting since this quotient is very small for $p \geq 5$. Finally, the BRRI method doesn't converge for any value of p.

Method	n	p	N_1	N_m	m	R	$\sigma_{\max}$
BRRM	100000	8	$0.232 \cdot 10^4$	$0.170 \cdot 10^{-10}$	232	0.140	0.0
BRRM	100000	7	$0.232 \cdot 10^4$	$0.449 \cdot 10^{-10}$	141	0.224	0.0
BRRM	100000	6	$0.232 \cdot 10^4$	$0.398 \cdot 10^{-10}$	164	0.193	$0.228 \cdot 10^{-11}$
BRRM	100000	5	$0.232 \cdot 10^4$	$0.569 \cdot 10^{-12}$	158	0.227	$0.306 \cdot 10^{-6}$

Table 10: The performance of the BRR method, for different values of p, in case of the Extended Powell singular function.

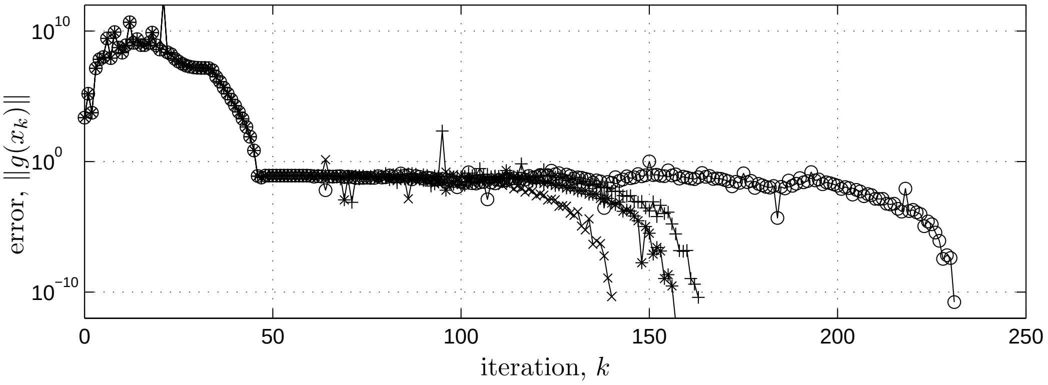


Figure 7: The convergence rate of the BRR method in case of the Extended Powell singular function for different values of p. ['$\circ$' (p = 8), '$\times$' (p = 7), '$+$' (p = 6), '$*$' (p = 5)]

5 The Reverse Flow Reactor

We now consider the example of the Reverse Flow Reactor in more detail. In the previous section we saw that the method of Broyden and the BRR method are not applicable to every function. It turns out, however, that the method of Broyden is well suited for many physical problems, especially from Chemical Reactor Engineering, see the application to the RFR in the introduction. In Section 5.1 we check the performance of the BRR(I) method for the one-dimensional model of the RFR. We furthermore show what happens to the singular values of the process when we reduce the number p. In Section 5.2 we consider the two-dimensional model of the RFR and shortly mention the importance of the radial dimension of the model for the RFR, by showing the difference in heat distribution over the reactor for thick and thin reactors.

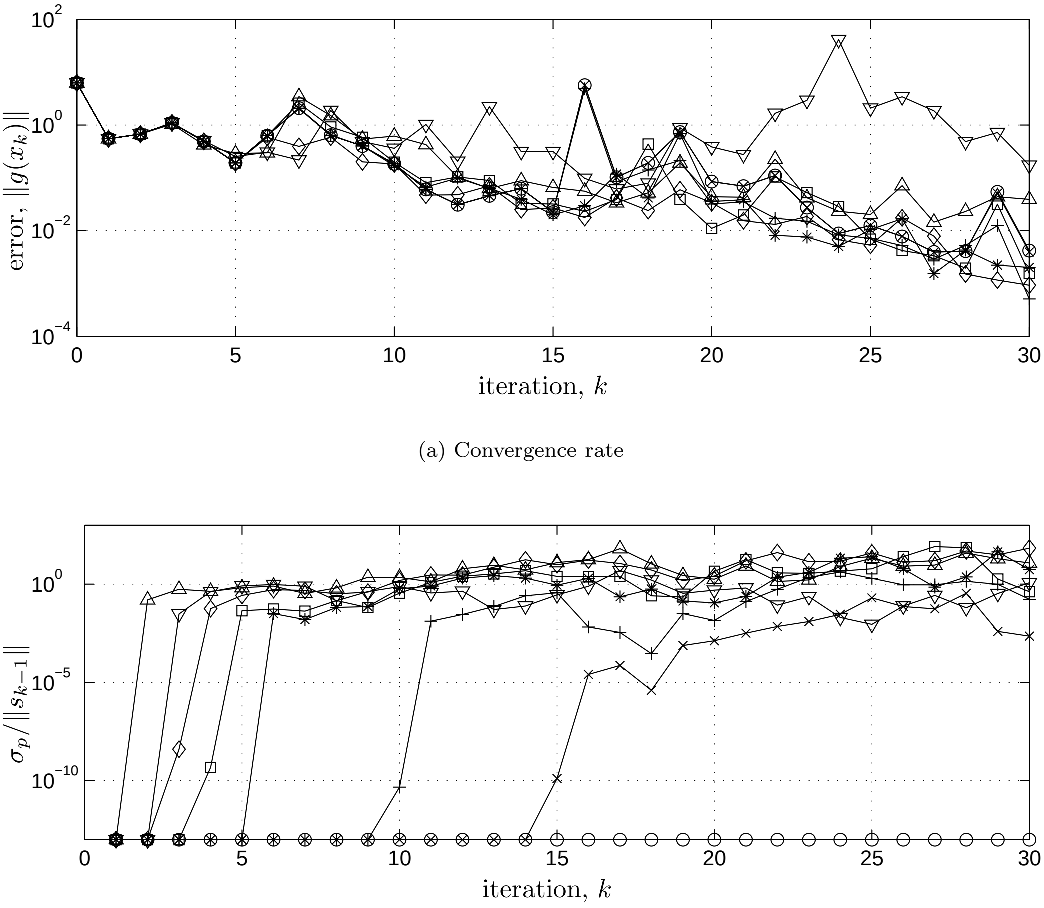
5.1 Performance of the BRR(I) method on 1D-model

As before let $f : \mathbb{R}^n \to \mathbb{R}^n$ be the period map of the one-dimensional model for the RFR ($n = 200$), and define $g : \mathbb{R}^n \to \mathbb{R}^n$ by $g(x) = f(x) - x$. We try to approximate a cyclic steady state of the reactor by finding a vector $x \in \mathbb{R}^n$ such that

$$\|g(x)\| < 10^{-10}. \tag{5.1}$$

In Table 2 of the introduction the results are given of the BRR method, starting from the initial condition $x_0 = (1, \ldots, 1)$. We obtain that even for relatively small values of p the results of the BRR method are not much worse than those of the method of Broyden. However, note that for every p the maximum of all removed singular value cannot be neglected.

Figure 8 shows the rate of convergence and the quotient $\sigma_p/\|s_{k-1}\|$ for the first 30 iterations of the BRR process. Here again one can see that the BRR method separates from the path of Broyden's method when $\sigma_p/\|s_{k-1}\|$ becomes too large. In most cases, however, the rate of convergence remains high, cf. Figure 1. Furthermore during the BRR process the quotient $\sigma_p/\|s_{k-1}\|$ increases until it has becomes larger than one. Thereafter it shows no uniform behavior; it stays constant, decreases monotonically or oscillates.



(a) Convergence rate

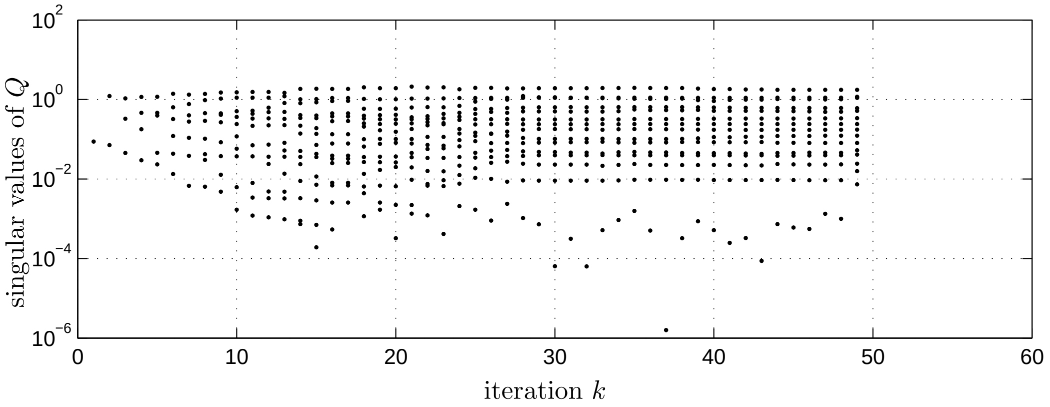
(b) Quotient of σ_p and $\|s_{k-1}\|$

Figure 8: The convergence rate of Broyden's method and the BRR method, in case of the period map of the one-dimensional model for the RFR, and additionally the quotient $\sigma_p/\|s_{k-1}\|$, for different values of p. ['$\circ$' (Broyden), '$\times$' ($p = 15$), '+' ($p = 10$), '$*$' ($p = 5$), '$\square$' ($p = 4$), '$\diamond$' ($p = 3$), '$\bigtriangledown$' ($p = 2$), '$\bigtriangleup$' ($p = 1$)]

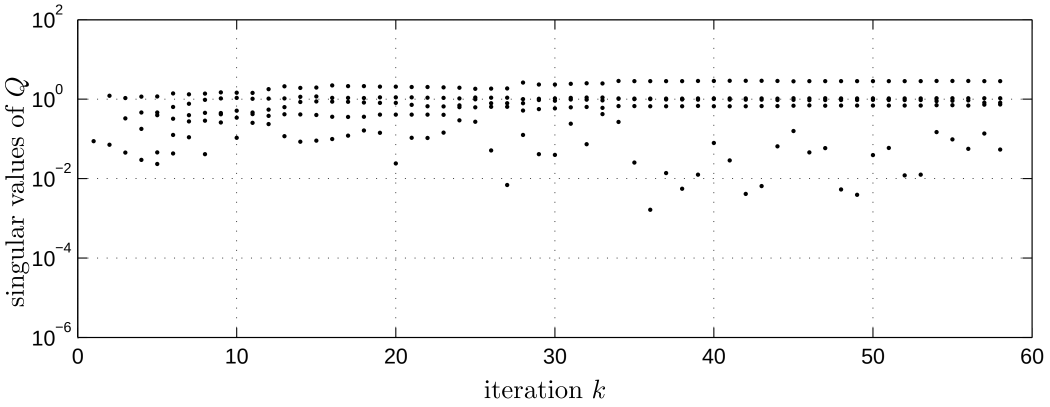
Thus far we only considered the pth singular value of the update matrix. Figure 9 shows the influence of removing the pth singular value on the distribution of the singular values of the update matrix. For every iteration we have plotted all singular values on a logscale for two different values of p.

In Figure 9(a) we see that in logscale the singular values finally are uniformly distributed over the interval $[10^{-2}, 10^0]$. Every iteration a smaller singular value is added and the other singular values shift above. After the 15th iteration step the largest ($p - 1 = 14$) singular values converge and fill more or less the interval $[10^{-2}, 10^0]$. The pth singular value fluctuates, and doesn't become extremely small or converges to the ($p - 1$)st singular value. In Figure 9(b) we see the same behavior for ($p = 5$). Here the largest ($p - 1 = 4$) singular values converge and the 5th singular value fluctuates. The most important difference is that the first singular value is larger in case of ($p = 5$) than for ($p = 15$). All other singular values are not influenced by the reduction step of the BRR method.

We can conclude the following. In order to obtain comparable performance for the BRR method as for the method of Broyden itself, in case of the period map of the RFR, it is sufficient to choose $p = 5$. This implies a reduction from $n^2 = 40,000$ to $2pn = 2,000$ storage locations to store the



(a) ($p = 15$)



(b) ($p = 5$)

Figure 9: The distribution of the singular values of the update matrix, Q, for $p = 15$ and $p = 5$.

Broyden matrix.

The performance of the BRRI method is comparable with the BRR method, cf. Table 11 and Figure 10. It is remarkable that for some values of p the BRR method is sometimes faster than the method of Broyden itself.

5.2 Performance of the BRR method on a 2D-model

Let $f : \mathbb{R}^n \to \mathbb{R}^n$ be the period map of the RFR corresponding to the two-dimensional model. Because ($N = 100$) gridpoints are used in axial direction and ($M = 25$) is radial direction, the dimension of the problem becomes ($n = 2 \cdot M \cdot N = 5000$). Define $g : \mathbb{R}^n \to \mathbb{R}^n$ by $g(x) = f(x) - x$. We search for a vector $x \in \mathbb{R}^n$ that satisfies

$$\|g(x)\| < 10^{-8}. \tag{5.2}$$

As we can see in Table 12 (and Table 2) the performance of the BRR method is for two-dimensional model similarly good as for the one-dimensional model. Also the maximum of all removed singular values, $\sigma_{\max}$, is of the same order in both cases. Remark that the tolerance imposed on the error for the two-dimensional model is larger than for the one-dimensional model. The reason is that due to the increased dimension of the problem, the error $\|g(x_k)\|$ increases with the same factor.

Method	n	p	N_1	N_m	m	R	$\sigma_{\max}$
Broyden	200	-	6.29	$0.957 \cdot 10^{-10}$	50	0.498	-
BRRIM	200	15	6.29	$0.540 \cdot 10^{-10}$	49	0.520	$0.651 \cdot 10^{-2}$
BRRIM	200	10	6.29	$0.704 \cdot 10^{-10}$	52	0.485	$0.821 \cdot 10^{-1}$
BRRIM	200	5	6.29	$0.955 \cdot 10^{-10}$	76	0.328	0.642
BRRIM	200	4	6.29	$0.311 \cdot 10^{-10}$	59	0.441	0.603
BRRIM	200	3	6.29	$0.676 \cdot 10^{-10}$	66	0.383	0.646
BRRIM	200	2	6.29	$0.940 \cdot 10^{-10}$	85	0.293	1.59
BRRIM	200	1	6.29	$0.253 \cdot 10^{-6}$	201	$0.847 \cdot 10^{-1}$	204

Table 11: The performance of Broyden's method and the BRRI method, in case of the period map of the one-dimensional model for the RFR, for different values of p.

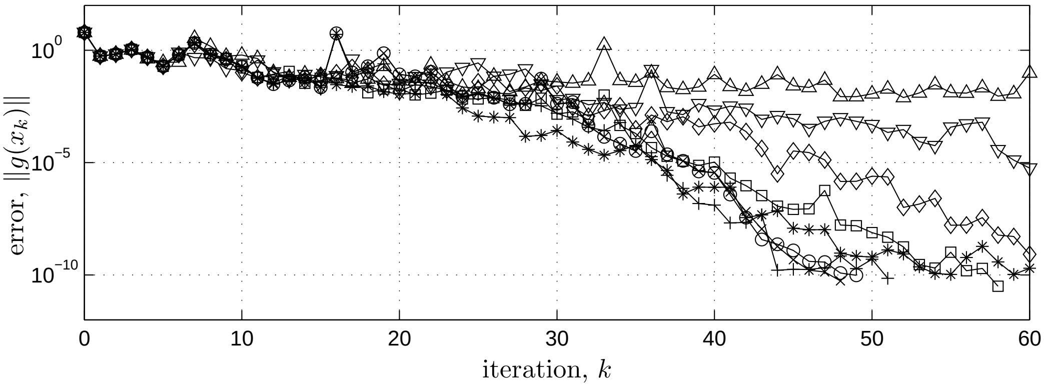


Figure 10: The convergence rate of Broyden's method and the BRRI method, in case of the period map of the one-dimensional model for the RFR, and additionally the quotient $\sigma_p/\|s_{k-1}\|$, for different values of p. ['$\circ$' (Broyden), '$\times$' ($p = 15$), '+' ($p = 10$), '$*$' ($p = 5$), '$\Box$' ($p = 4$), '$\diamond$' ($p = 3$), '$\bigtriangledown$' ($p = 2$), '$\bigtriangleup$' ($p = 1$)]

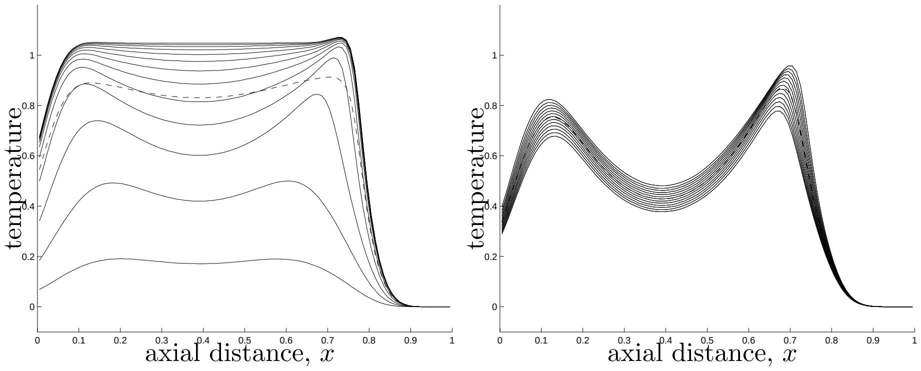
Additionally, integration routines to evaluate the function g at x_k introduce larger rounding error for higher dimensions.

To show the necessity of a two-dimensional model, we consider the Reverse Flow Reactor for two different values of the reactor width. Figure 11 shows the distribution of heat in the reactor in cyclic steady state, i.e., the state that returns every period. Because the state of the reactor is assumed to be cylindrically symmetric, we have plotted for different positions in radial direction the temperature against the axial distance. The upper lines correspond to the state around the axis of the reactor. The lower lines correspond to the state next to the wall of the reactor. The dotted line is the averaged temperature over the cross section of the reactor. If the reactor has a large radius, see Figure 11(a) the cooling has less influence of the state around the axis of the reactor; here the process is more or less adiabatic (without cooling). At the wall, on the other hand, the reactor is rather cold. Considering the thin reactor, Figure 11(b), the temperature gradients in radial direction are much smaller and cooling also affects the state around the axis of the reactor. For the cyclic steady states of the one dimensional model we refer to the article of Khinast et al. [9].

The reaction in the reactor strongly depends on the temperature. To overcome a mixture of reactants and products at the end of the reactor it is preferable to have no radial temperature gradients. In that case a reactor with small diameter turns out to be more attractive. For more details see [21].

Method	n	p	N_1	N_m	m	R	$\sigma_{\max}$
BRRM	5000	15	$0.315 \cdot 10^2$	$0.974 \cdot 10^{-8}$	39	0.561	$0.109 \cdot 10^{-1}$
BRRM	5000	10	$0.315 \cdot 10^2$	$0.382 \cdot 10^{-8}$	40	0.571	$0.751 \cdot 10^{-1}$
BRRM	5000	5	$0.315 \cdot 10^2$	$0.768 \cdot 10^{-8}$	57	0.388	0.376
BRRM	5000	4	$0.315 \cdot 10^2$	$0.890 \cdot 10^{-8}$	54	0.407	0.581
BRRM	5000	3	$0.315 \cdot 10^2$	$0.518 \cdot 10^{-8}$	174	0.129	1.08
BRRM	5000	2	$0.315 \cdot 10^2$	$0.586 \cdot 10^{-8}$	107	0.209	1.20
BRRM	5000	1	$0.315 \cdot 10^2$	$0.953 \cdot 10^{-8}$	170	0.129	2.62

Table 12: The performance of the BRR method, in case of the period map of the two-dimensional model for the RFR, for different values of p.



(a) Reactor with large radius

(b) Reactor with small radius

Figure 11: Qualitative temperature distribution in the cooled Reverse-Flow Reactor. Every line shows the distribution in axial direction for fixed radial position. The dotted line is the average of the cross section.

6 Conclusion

In Section 5 we have presented qualitative differences between one- and two-dimensional models for the RFR. If the full two-dimensional model must be discretized on a fine grid to obtain a sufficiently accurate solution, the dimension of the problem increases significantly. Therefore memory reduction methods are essential. We discussed several limited memory quasi-Newton methods to solve high-dimensional systems of nonlinear equations. In particular our examples showed that the BRR method is rather promising. With p chosen in advance, a reduction in memory can be made from n^2 to $2pn$ storage locations to store the quasi-Newton matrix. This in combination with the same rate of convergence as the original method of Broyden. The BRR method can be seen as an extension to dynamical simulation in which extra storage is used to accelerate convergence and simultaneously as an implementation of a quasi-Newton method in which storage is restricted. Comparable to the L-BFGS method of Nocedal [16] the simplicity of the BRR method is one of the main appeals: It does not require knowledge of the sparsity structure of the Jacobian nor of the separability of the objective function.

The example of the Discrete boundary value function, see Section 4.2, clearly shows that the BRR method is not usable for every function. But it might also tell us why Broyden's method has difficulties in approximating the zero of this function; the update matrix of the Broyden process has full rank.

If a singular value σ_p is removed, a singular value σ_p of the same order returns the next iteration of the BRR method. If the quotient $\sigma_p/\|s_{k-1}\|$ is large for some iteration, it will stay large unless the process diverges. If the quotient $\sigma_p/\|s_{k-1}\|$ remains small the BRR method is identical to the method of Broyden. Therefore we propose the following algorithm to solve a general large scale system of nonlinear equations, using the BRR method.

Algorithm 6.1 *Let $g : \mathbb{R}^n \to \mathbb{R}^n$ be given and choose initial condition $x_0 \in \mathbb{R}^n$ and tolerance $\varepsilon > 0$ appropriately. Perform the following steps.*

1. *Choose $\eta \in (0, 1]$ and set $p = 1$.*
2. *If $p > n/2$ then stop the algorithm.*
3. *Apply the BRR method as long as $\sigma_p/\|s_{k-1}\| < \eta$ and $\|g(x_k)\| > \varepsilon$.*
4. *If $\sigma_p/\|s_{k-1}\| \geq \eta$ then increase p by one, define a new initial condition by*

$$\begin{cases} x_k & \text{if } \|g(x_k)\| < \|g(x_0)\|, \\ x_0 & \text{otherwise,} \end{cases}$$

and return to step 2.

If it turns out that p must be chosen larger than $n/2$, then it is preferable to use the method of Broyden, if possible.

Simulations indicate that the limiting states of a reverse flow reactor belong to a low dimensional set in the full space of all possible states. See [17] for more information on the role of the low dimensional dynamics in iterative algorithms. The fact that the appropriate integer p in the BRR method can be rather small might actually be a consequence of this low dimensional dynamics. We hope to return to these issues in the future.

Acknowledgment

We would like to thank Alfred Bliek for many stimulating discussions. The analysis of the full two dimensional model for the reverse flow reactor give us inspiration for the Broyden Rank Reduction method.

References

[1] C.G. Broyden. A class of methods for solving nonlinear simultaneous equations. *Math. Comput.*, 19:577–593, 1965.

[2] C.G. Broyden, J.E. Dennis, and J.J. Moré. On the local and superlinear convergence of quasi-Newton methods. *J. Inst. Maths. Applics.*, 12:223–245, 1973.

[3] R.H. Byrd, J. Nocedal, and R.B. Schnabel. Representations of quasi-Newton matrices and their use in limited memory methods. *Math. Program.*, 63:129–156, 1994.

[4] J.E. Dennis and J.J. Moré. Quasi-Newton methods, motivation and theory. *SIAM Rev.*, 19(1):46–89, 1977.

[5] J.E. Dennis and R.B. Schnabel. *Numerical methods for unconstrained optimization and nonlinear equations.* Prentice-Hall, Englewood Cliffs, NJ, 1983.

[6] D.M. Gay. Some convergence properties of Broyden's method. *SIAM J. Numer. Anal.*, 16(4):623–630, 1979.

[7] R.R. Gerber and F.T. Luk. A generalized Broyden's method for solving simultaneous linear equations. *SIAM J. Numer. Anal.*, 18(5):882–890, 1981.

[8] G.H. Golub and C.F. Van Loan. *Matrix computations.* Johns Hopkins Studies in the Mathematical Sciences. Johns Hopkins University Press, third edition, 1996.

[9] J. Khinast, Y.O. Jeong, and D. Luss. Dependence of cooled Reverse-Flow Reactor dynamics on reactor model. *AIChE Journal*, 45(2):299–309, 1999.

[10] T.G. Kolda, D.P. O'Leary, and L. Nazareth. BFGS with update skipping and varying memory. *SIAM J. Optim.*, 8(4):1060–1083, 1998.

[11] D.C. Liu and J. Nocedal. On the limited memory BFGS method for large scale optimization. *Math. Program.*, 45:503–528, 1989.

[12] J.M. Martínez. Practical quasi-Newton methods for solving nonlinear systems. *J. Comput. Appl. Math.*, 124:97–121, 2000.

[13] J.L. Morales and J. Nocedal. Automatic preconditioning by limited memory quasi-Newton updating. *SIAM J. Optim.*, 10(4):1079–1096, 2000.

[14] J.J. Moré and M.Y. Cosnard. Numerical solution of nonlinear equations. *ACM Trans. Math. Soft.*, 5(1):64–85, 1979.

[15] J.J. Moré, B.S. Garbow, and K.E. Hillstrom. Testing Unconstrained Optimization Software. *ACM Trans. Math. Soft.*, 7(1):17–41, 1981.

[16] J. Nocedal. Updating quasi-Newton matrices with limited storage. *Math. Comput.*, 35(151):773–782, 1980.

[17] T.L. van Noorden. *New algorithms for parameter-swing reactors.* PhD thesis, Free University of Amsterdam, 2002.

[18] T.L. van Noorden, S.M. Verduyn Lunel, and A. Bliek. Acceleration of the determination of periodic states of cyclically operated reactors and separators. *Chem. Eng. Sci.*, 57:1041–1055, 2002.

[19] D.P. O'Leary. Why Broyden's nonsymmetric method terminates on linear equations. *SIAM J. Optim.*, 5(2):231–235, 1995.

[20] J.M. Ortega and W.C. Rheinboldt. *Iterative solution of nonlinear equations in several variables.* Computer Science and Applied Mathematics. Academic Press, 1970.

[21] B.A. van de Rotten, S.M. Verduyn Lunel, and A. Bliek. Dependence of cooled Reverse-Flow Reactor dynamics on reactor width. In preparation.

[22] L.K. Schubert. Modification of a quasi-Newton method for nonlinear equations with a sparse Jacobian. *Math. Comput.*, 24(109):27–30, 1970.

[23] Ph.L. Toint. On sparse and symmetric matrix updating subject to a linear equation. *Math. Comput.*, 31(140):954–961, 1977.

[24] Ph.L. Toint. A sparse quasi-Newton update derived variationally with a nondiagonally weighted Frobenius norm. *Math. Comput.*, 37(156):425–433, 1981.