Received: Added at production | Revised: Added at production | Accepted: Added at production

DOI: xxx/xxxx

RESEARCH ARTICLE

On the effect of non-linearity and Jacobian initialization on the convergence of the generalized Broyden quasi-Newton method

Toon Demeester[1] | Nicolas Delaissé[1] | E. Harald van Brummelen[2] | Rob Haelterman[3] | Joris Degroote[1,4]

[1]Department of Electromechanical, Systems and Metal Engineering, Ghent University, Ghent, Belgium
[2]Department of Mechanical Engineering, Eindhoven University of Technology, Eindhoven, the Netherlands
[3]Department of Mathematics, Royal Military Academy, Brussels, Belgium
[4]Flanders Make, Belgium

Correspondence
Toon Demeester, Department of Electromechanical, Systems and Metal Engineering, Ghent University, Sint-Pietersnieuwstraat 41, 9000 Gent, Belgium. Email: toon.demeester@ugent.be

Summary

This paper investigates two aspects of the generalized Broyden quasi-Newton method that have a major impact on its convergence: the initial approximation of the Jacobian and the presence of non-linearities in the secant conditions.

After reformulating the common representation of generalized Broyden, a straightforward interpretation is given. This leads to a natural extension of the method in which an application-dependent physics-based surrogate model is used as initial approximation of the (inverse) Jacobian. A carefully chosen surrogate has the potential to greatly reduce the required number of iterations.

The behavior of generalized Broyden depends strongly on the parameter that determines how many secant conditions are satisfied by the Jacobian approximation. Respecting all secant conditions reduces it to Anderson acceleration; a single one to Broyden's original method. An analysis demonstrates that these two variants behave very differently when non-linearities are present in the secant conditions: they are ignored by Broyden, but can destabilize Anderson. On the other hand, the analysis shows that Broyden tends to neglect small linear information, possibly reducing convergence speed. To mitigate stability problems with Anderson acceleration, a practical method to detect and remove non-linear secant information is introduced next.

Finally, we solve a steady free-surface-flow problem using several generalized Broyden variants, testing the influence of the surrogate, the non-linearities and the combination thereof. The results agree with the theoretical predictions, showing large differences in convergence behavior. Furthermore, the proposed method effectively negates the problems related to non-linearities in this case.

KEYWORDS:
quasi-Newton, Broyden's method, Anderson acceleration, surrogate model, free-surface flow

1 | INTRODUCTION

In many fields of science and engineering, non-linear systems of equations of the form $\mathcal{F}(x) = 0$ must be solved. This usually means that iterations need to be performed to find an input x for which the residual $\|\mathcal{F}(x)\|$ is smaller than a specified tolerance.

In this paper we focus specifically on systems that are expensive to evaluate, so that consequently the function evaluations of $\mathcal{F}(x)$ dominate the total cost of the iterative solution process.

To solve such systems two methods were developed in the sixties, which have since pervaded many application fields: Anderson acceleration[1] and Broyden's methods.[2] The former was developed to accelerate fixed-point iterations, the latter as a quasi-Newton method for solving non-linear systems. Although these methods seem distinctly different at first sight, both can actually be recognized as special cases of the *generalized Broyden* quasi-Newton method.[3] In this paper we investigate two aspects of this method that have a significant influence on its convergence behavior: the choice of initial Jacobian approximation and the presence of non-linearities in the secant information.

Section 2 discusses the generalized Broyden method in some detail and shows how Anderson and Broyden can be found as its limiting cases. It is then explained how the initial approximate Jacobian can be replaced by an application-specific physics-based surrogate model to boost convergence of the quasi-Newton iterations.

In the generalized Broyden method, inputs and outputs of $\mathcal{F}(x)$ from previous iterations are used to approximate the inverse Jacobian. If $\mathcal{F}(x)$ is non-linear however, some of the earlier collected data may not be relevant anymore for the local behavior of $\mathcal{F}(x)$ and can lead to an inaccurate approximation of the inverse Jacobian. To comprehend why and how this can impact convergence, Section 3 analyses how generalized Broyden behaves when non-linearities are encountered. Interestingly, we find a clear distinction between Anderson and Broyden; our new insights can be summarized as follows. Broyden tends to neglect small linear information, which may lead to slower convergence. More importantly, Anderson acceleration amplifies non-linearities present in the secant information, which can lead to convergence issues.

To mitigate these issues with Anderson acceleration, Section 4 proposes a new method to detect and systematically remove obstructive non-linear data on the fly.

In Section 5, the methods discussed in the earlier sections are applied to a steady free-surface-flow problem, where $\mathcal{F}(x)$ is a Reynolds-averaged Navier-Stokes solver. Attention is focused on the influence of the surrogate model, the impact of non-linearities and the combination thereof. The results support our conclusions about the impact of non-linearities on generalized Broyden, and demonstrate that the proposed method works.

2 | THE GENERALIZED BROYDEN METHOD WITH PHYSICS-BASED SURROGATE

Let us first introduce the notation conventions used in this paper. Curly letters such as $\mathcal{F}$ are used for non-linear functions. Matrices are denoted by upper case, vectors by lower case, e.g. F and f.

In this paper, systems of equations of the form

$$\mathcal{F}(x) = 0 \tag{1}$$

are considered, where $\mathcal{F} : \mathbb{R}^n \to \mathbb{R}^n$ is a continuously differentiable operation. The standard approach for solving non-linear systems of the form (1), is by means of Newton-Raphson iterations, where given an approximation x_k the next iterate x_{k+1} is calculated as

$$x_{k+1} = x_k - \mathcal{F}'(x_k)^{-1} f_k \tag{2}$$

with k the iteration number and $\mathcal{F}'(x_k)$ the Jacobian of $\mathcal{F}$ at x_k. The notation

$$f_i \equiv \mathcal{F}(x_i) \tag{3}$$

is introduced here to compactly denote function values. New iterates, x_k, are calculated and evaluated until the residual $\|f_k\|$ drops below a prescribed tolerance ε, where generally $\|\cdot\|$ corresponds to the Euclidean norm.

In many applications of practical importance, the Jacobian $\mathcal{F}'(x_k)$ is not explicitly available or its assembly may carry a prohibitive computational cost. Its inverse can then be replaced by an approximation, G_k, resulting in the quasi-Newton iteration

$$x_{k+1} = x_k - G_k f_k. \tag{4}$$

Only quasi-Newton methods that approximate the inverse of the Jacobian, equation (4), are discussed in this paper. For every such method, an analogous method exists that approximates the Jacobian itself. Approximating the inverse avoids having to solve a linear system with a (possibly dense) $\mathbb{R}^{n\times n}$ matrix in the quasi-Newton iteration, so that these methods can be executed in a *matrix-free* way. With this term, we refer to the fact that the new iterate can be constructed based on a low-cost procedure for the matrix-vector product $G_k f_k$, avoiding the expensive construction and storage of the matrix G_k itself. However, for some quasi-Newton methods the inverse of the approximate Jacobian can be formed cheaply using the Sherman–Morrison–Woodbury formula as long as $k \ll n$, which also results in a matrix-free quasi-Newton iteration.[4]

An important class of quasi-Newton methods is formed by the generalized Broyden methods as discussed by Eyert[3] and Fang and Saad,[4] where G_k is constructed using input-output information of $\mathcal{F}$ from previous iterations. This input-output information is stored and used in the form

$$\Delta x_i \equiv x_{i+1} - x_i \tag{5}$$

$$\Delta f_i \equiv f_{i+1} - f_i \tag{6}$$

for $i = 0, 1, \ldots, k-1$, where Δx_i and Δf_i are respectively iterate differences and corresponding function-value differences between consecutive iterations. If G_k, the approximate inverse Jacobian in iteration k, relates these differences to each other, it is said that G_k fulfills the *secant condition*

$$\Delta x_i = G_k \Delta f_i. \tag{7}$$

The pair $(\Delta x_i, \Delta f_i)$ will henceforth be called the *secant information* at iteration i. Symbolically, the ratio of the vector Δf_i to the vector Δx_i can be interpreted as a central difference approximation of $\mathcal{F}'$ in direction Δx_i at the midpoint

$$\bar{x}_i = \frac{x_i + x_{i+1}}{2}. \tag{8}$$

The secant information $(\Delta x_i, \Delta f_i)$ can also be conceived of as a one-sided finite difference in either x_i or x_{i+1}. The error with respect to those points is first order, while for the central difference it is second order, hence the choice in this paper to relate $(\Delta x_i, \Delta f_i)$ to the midpoint $\bar{x}_i$.

The secant information from the m latest iterations is collected in the matrices

$$X_k = \begin{bmatrix} \Delta x_{k-1} & \Delta x_{k-2} & \cdots & \Delta x_{k-m} \end{bmatrix} \in \mathbb{R}^{n \times m}, \tag{9}$$

$$F_k = \begin{bmatrix} \Delta f_{k-1} & \Delta f_{k-2} & \cdots & \Delta f_{k-m} \end{bmatrix} \in \mathbb{R}^{n \times m}. \tag{10}$$

If $k < m$, secant information for only k iterations is available, so the subscript $k-m$ should in fact be replaced with $\max(k-m, 0)$ in equations (9) and (10). For the sake of readability, the subscript $k-m$ will be used in the remainder of the paper instead of $\max(k-m, 0)$.

The secant conditions on G_k for the m latest iterations can be grouped together in a system

$$X_k = G_k F_k. \tag{11}$$

With these definitions, the generalized Broyden method defines G_k recursively as

$$G_k = G_{k-m} + \left(X_k - G_{k-m} F_k\right) \left(F_k^T F_k\right)^{-1} F_k^T \tag{12}$$

with G_0 an initial estimate of the inverse Jacobian provided by the user.[3,4] How expression (12) can be derived and interpreted is explained in detail in the next section. The complete quasi-Newton procedure with a generalized Broyden approximation G_k is given in Algorithm 1.

Algorithm 1 Quasi-Newton with generalized Broyden.

choose x_0, G_0, ε, m
$x_1 = x_0 - G_0 \mathcal{F}(x_0)$
$f_1 = \mathcal{F}(x_1)$
$k = 1$
while $\|f_k\| > \varepsilon$ **do**
  for $j \in \left[0, 1, \cdots, \text{floor}\left(\frac{k-1}{m}\right)\right]$ **do**
    $l = k - jm$
    $X_l = \begin{bmatrix} \Delta x_{l-1} & \Delta x_{l-2} & \cdots & \Delta x_{l-m} \end{bmatrix}$
    $F_l = \begin{bmatrix} \Delta f_{l-1} & \Delta f_{l-2} & \cdots & \Delta f_{l-m} \end{bmatrix}$
  end for
  $x_{k+1} = x_k - G_k f_k$ ▷ evaluate recursively (matrix-free) using expression (16)
  $f_{k+1} = \mathcal{F}(x_{k+1})$
  $k = k + 1$
end while

Interpreting generalized Broyden

Before we give an interpretation of the generalized Broyden quasi-Newton method, we convert expression (12) for G_k to a more amenable form in two steps.

First, the rightmost three terms of expression (12) can be recognized as the pseudo-inverse F_k^+ of the rectangular matrix F_k, defined as

$$F_k^+ = \left(F_k^T F_k\right)^{-1} F_k^T. \tag{13}$$

The standard method for evaluating the pseudo-inverse[5] uses the economy-size QR decomposition

$$F_k = Q_k R_k \qquad \text{with} \qquad Q_k \in \mathbb{R}^{n\times m}, R_k \in \mathbb{R}^{m\times m} \tag{14}$$

to simplify F_k^+, reducing expression (12) to

$$G_k = G_{k-m} + \left(X_k - G_{k-m}F_k\right) R_k^{-1} Q_k^T. \tag{15}$$

Note that the identity $Q_k^T Q_k = I$ can be used to obtain expression (15), as the columns of Q_k are orthonormal by definition; on the other hand $Q_k Q_k^T \neq I$, because Q_k is rectangular. The second step is to isolate the old Jacobian G_{k-m}, yielding the final form of generalized Broyden we will use in this paper:

$$\begin{aligned} G_k &= X_k R_k^{-1} Q_k^T + G_{k-m}\left(I - F_k R_k^{-1} Q_k^T\right) \\ &= X_k R_k^{-1} Q_k^T \left(Q_k Q_k^T\right) + G_{k-m}\left(I - Q_k Q_k^T\right). \end{aligned} \tag{16}$$

To evaluate the product $G_k f_k$ that appears in the quasi-Newton expression (4), the approximate inverse Jacobian G_k from expression (16) does not need to be evaluated explicitly. It is cheaper, both in terms of storage and computational complexity, to directly calculate the matrix-vector product in a matrix-free way. For this purpose, each term of the product $G_k f_k$ is evaluated from right to left (e.g. starting with $Q_k^T f_k$), avoiding the formation of large $n \times n$ matrices. As the previous approximation G_{k-m} is also not stored, this results in a recursive evaluation of expression (16).

Expression (16) admits a simpler interpretation than we could give based on expression (12). When the product $G_k f_k$ is taken in the quasi-Newton iteration (4), the vector f_k is first split in two orthogonal parts by the complementary orthogonal projectors $Q_k Q_k^T$ and $(I - Q_k Q_k^T)$. This yields a part that lies in range $\left(Q_k\right)$ and one that is orthogonal to range $\left(Q_k\right)$. We know that range $\left(Q_k\right)$ is equal to range $\left(F_k\right)$, which means that the part of $f_k \in$ range $\left(Q_k\right)$ is a linear combination of function-value differences Δf_{k-i} $(1 \leq i \leq m)$. After f_k has been split in two parts, a different approximate inverse Jacobian is used for each part. The part $(I - Q_k Q_k^T) f_k$ cannot be linked to any of the secant information included in F_k. Therefore the Jacobian G_{k-m} from m iterations ago is used. This can be expressed by the *no-change conditions* according to:

$$G_k v = G_{k-m} v, \ \forall v \notin \text{range}\left(F_k\right). \tag{17}$$

The part $Q_k Q_k^T f_k$ is a linear combination of differences Δf_i encountered in the m previous iterations, so we can look at the secant information stored in X_k and F_k to see how the function $\mathcal{F}(x)$ behaved in previous iterations and assume the same behavior in the current iteration. For this purpose the approximate inverse Jacobian $X_k R_k^{-1} Q_k^T$ is used, which is the solution to the secant conditions (11) with minimum Frobenius norm, and is therefore uniquely determined.

An important condition for generalized Broyden to work, is that R_k is non-singular. This requires that all Δf_i in F_k are linearly independent, which we assume to hold in the rest of this paper. In practice, if two linearly dependent columns are encountered in F_k, the oldest of the two columns is removed. More information about filtering out linearly dependent information can be found in the paper by Haelterman et al.[6]

To conclude, the expression (16) for G_k can be found as the only[†] matrix that satisfies both the m secant conditions (11) and the $(n - m)$ no-change conditions (17). This is not only the core idea behind generalized Broyden, but also behind Broyden's original methods for which $m = 1$.

Adding a physics-based surrogate

Efficiency of the iterative procedure demands that the number of iterations k is much smaller than the size n of the problem. Hence, only a low-rank approximate inverse can be built using the available secant information. The initial estimate G_0 is used

[†]Note that a total of n independent vector conditions is indeed required to uniquely determine the n^2 unknowns of the approximate inverse Jacobian $G_k \in \mathbb{R}^{n\times n}$.

to predict how the function behaves in all other directions. This makes the choice of G_0 crucial: a bad one can lead to excessively slow convergence.

Often G_0 is chosen as $G_0 = -\beta I$, where β is called the *mixing* or *relaxation* factor, which is between zero and one. When solving fixed-point problems with a quasi-Newton method, this choice is rational. The fixed-point problem $\mathcal{H}(x) = x$ can be reformulated to a non-linear root-finding problem by setting $\mathcal{F}(x) = \mathcal{H}(x) - x$. The quasi-Newton iteration with $G_0 = -\beta I$ can then be written as

$$\begin{aligned} x_{k+1} &= x_k + \beta f_k \\ &= (1-\beta)x_k + \beta \mathcal{H}(x_k). \end{aligned} \tag{18}$$

This clearly shows how the new vector x_{k+1} is obtained as a convex combination of the vectors x_k and $\mathcal{H}(x_k)$ in a ratio dictated by β, the simplest approach for solving fixed-point problems. If $\mathcal{H}$ is a linear operator with spectrum Σ, then the spectrum of the operator $x_k \mapsto x_{k+1}$ in equation (18) is $(1-\beta)+\beta\Sigma$. The coefficient β should in principle be selected such that the spectral radius of this operator is minimal. A good value β for a given application is typically chosen based on experience, in some cases taking system parameters into account.[7] For more general functions $\mathcal{F}$, whose input and output represent different physical quantities (e.g. position and pressure for the application in Section 5), this fixed-point interpretation is not applicable.

For specific applications, a physics-based surrogate model for the inverse Jacobian

$$G_{\text{sur}}(\Delta f) = \Delta x \tag{19}$$

may be available. This G_{sur} can for example originate from a reduced-physics or coarse-mesh approximation of $\mathcal{F}$. Such a model can replace the product of G_0 with a vector as encountered in the recursive definition of G_k for generalized Broyden (16), such that $G_0 v$ becomes $G_{\text{sur}}(v)$. This surrogate can come in a number of forms—a matrix, a linear operator or even a non-linear function—but should meet a few criteria in order to be useful:

- $G_{\text{sur}}(\Delta f)$ must evaluate significantly faster than $\mathcal{F}(x)$;
- To make sure that the vector $f_0 \in \mathbb{R}^n$ can converge completely to the zero-vector during the iterations, $G_{\text{sur}}(\Delta f)$ must be surjective, i.e. map to all of $\mathbb{R}^n$. As $G_{\text{sur}} : \mathbb{R}^n \to \mathbb{R}^n$, it must therefore be a bijection;
- $G_{\text{sur}}(\Delta f)$ must provide a better approximation to the inverse Jacobian than $-\beta I$ for most vectors in $\mathbb{R}^n$. In other words, if quasi-Newton iterations were done purely with G_0, then G_{sur} should provide faster convergence than $-\beta I$ for most inputs. If it provides a worse approximation for a suitably small number of inputs Δf, this need not be an issue as is demonstrated in Section 5.

In Section 5 a steady free-surface-flow problem is solved. If $G_0 = -\beta I$ is used as initial approximation, convergence of the quasi-Newton iterations becomes grid-dependent. This means that a more accurate discretization of the problem not only makes $\mathcal{F}(x)$ more expensive to evaluate, but also slows down convergence of the quasi-Newton iterations. Using a suitable surrogate can remove this grid-dependence.

The importance of the initialization of the approximate Jacobian in quasi-Newton methods is well-known in the field; it was investigated e.g. by Gilbert and Lemaréchal[8] and more recently by Brust et al.[9] These investigations have however been conducted without considering a physics-based surrogate.

One method, two faces: Anderson and Broyden

Since their development in the sixties, Anderson acceleration[1] and Broyden's methods[2] have been two of the most popular techniques for solving (nonsymmetric) non-linear systems.

Broyden's original methods were extended in the eighties to the rather complex modified Broyden method.[10,11] In the nineties Eyert[3] simplified this method by removing some nonessential parameters, resulting in the previously discussed generalized Broyden method. It was only around this time—three decades after they were both introduced—that the connection between Anderson acceleration and Broyden's methods became apparent. Based on work by van Leuken,[12] Eyert showed that Anderson acceleration is mathematically equivalent to generalized Broyden with $m = k$. This is not immediately apparent due to the very different ideas on which Anderson and Broyden originally based their methods. While on the topic of equivalent methods,

we mention that Anderson acceleration also corresponds to the recent quasi-Newton inverse least squares (QN-ILS) method,[13] which was developed independently in the fluid-structure interaction community.[‡]

Further on we will investigate the two limiting cases of generalized Broyden in terms of the parameter m. If $m = k$, secant information from all previous iterations is included in X_k and F_k and generalized Broyden (16) reduces to Anderson acceleration, denoted with a subscript A:

$$G_{k,\mathrm{A}} = X_k R_k^{-1} Q_k^T + G_0 \left(I - Q_k Q_k^T \right). \tag{20}$$

As all secant conditions are met here, the no-change conditions (17) directly apply to G_0 and the formula for Anderson acceleration is not recursive anymore.

If $m = 1$, only secant information from the previous iteration is included in X_k and F_k, so that generalized Broyden (16) reduces to Broyden's second (nicknamed "bad") method, denoted with a subscript B:

$$G_{k,\mathrm{B}} = \frac{\Delta x_{k-1} \Delta f_{k-1}^T}{\Delta f_{k-1}^T \Delta f_{k-1}} + G_{k-1,B} \left(I - \frac{\Delta f_{k-1} \Delta f_{k-1}^T}{\Delta f_{k-1}^T \Delta f_{k-1}} \right). \tag{21}$$

In the remainder of this paper Anderson acceleration (20) and Broyden's second method (21) will be referred to simply as *Anderson* and *Broyden*.

In literature where Anderson acceleration is used as a general non-linear solver, the amount of secant information that is retained is often quite small: X_k and F_k are then limited to only a few columns $(2-5)$ to avoid (near) linear dependence issues.[3] Such a conservative choice is safer, but limits the power of the method in terms of efficiency. In literature related to (steady) partitioned fluid-structure interaction simulations, typically all secant information is retained in Anderson (e.g. IQN-ILS). In this paper we do not focus on the optimal number of columns—which is problem dependent—but compare different generalized Broyden variants that each use the same amount of secant information to approximate the inverse Jacobian. The following section shows that the way in which the retained secant information is used, i.e. Anderson or Broyden, has important consequences for the influence of non-linearities.

3 | THE INFLUENCE OF NON-LINEARITIES

Although generalized Broyden has not been studied extensively as such, its limiting cases, Anderson and Broyden, have received ample consideration in the literature. Their properties when applied to linear systems are well understood[17,23,24,25] and relevant for non-linear systems too, as local convergence behavior is usually dominated by the tangent operator at the solution.[26]

Non-linearities can play an important role in the convergence process. During the first few iterations, the iterate can be quite far from the solution and the steps can be large. Non-linearities in $\mathcal{F}$ may then be picked up and appear in the secant information, which means they are interpreted incorrectly as linear information when forming G_k, which in turn impacts the convergence. The mechanisms behind this are investigated here for the generalized Broyden method. More specifically, we are interested in the impact of the parameter m, which corresponds to the number of secant equations that are satisfied by G_k. As Anderson and Broyden correspond to the maximum and minimum values of m, it is natural to compare these two methods.

In the following analysis, we will artificially create two pairs of secant information, by choosing three iterates x_0, x_1 and x_2. These will then yield corresponding function values f_0, f_1 and f_2, following definition (3). To be clear: the values x_1 and x_2 are not obtained through quasi-Newton iterations here, but are selected in advance to insert a well-defined non-linearity in the secant information. The differences between consecutive iterates and consecutive function values then yield the secant pairs $(\Delta x_0, \Delta f_0)$ and $(\Delta x_1, \Delta f_1)$, following definitions (5) and (6). For both Anderson and Broyden, we will use this secant information to construct the approximate inverse Jacobian G_2, and calculate a new iterate using the quasi-Newton equation $x_3 = x_2 - G_2 f_2$. If x_0, x_1 and x_2 are chosen carefully, this result can yield valuable insights into the effects of non-linearities on the convergence of Anderson and Broyden.

If $\mathcal{F}$ is non-linear, its Jacobian $\mathcal{F}'$ is not constant. This means that for a non-linear $\mathcal{F}$, if the values of x_0, x_1 and x_2 are chosen in such a way that Δf_0 is equal to Δf_1, then Δx_0 is not necessarily equal to Δx_1, as would be the case for a linear $\mathcal{F}$.Hence,

[‡]In partitioned fluid-structure interaction simulations, black-box flow and structure solvers are evaluated in an iterative scheme until their respective solution fields meet certain interface conditions. After the development of I-GMRES[14] to accelerate this fixed-point problem, several quasi-Newton schemes were introduced for the same purpose: IBQN-LS[15] and IQN-ILS[16,17] correspond to Anderson acceleration, MVQN[18] and IQN-MVJ[19] to generalized Broyden where the (inverse) Jacobian of the previous timestep is used as old Jacobian. Only recently the correspondence between these methods was discovered.[20,21,22] IQN-ILS has emerged as a universal acceleration method in many iterative approaches, and has for instance been adopted in the open-source preCICE coupling tool and the commercial tool ANSYS System Coupling.

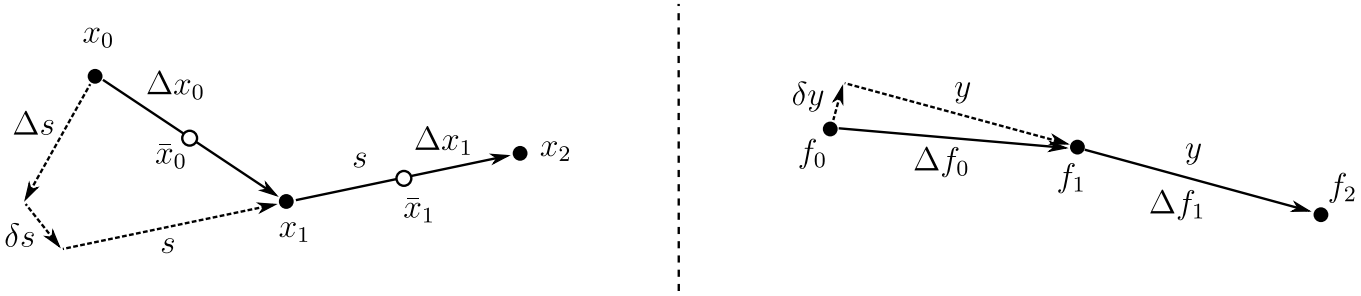

FIGURE 1 Illustration of secant information for the test problem with $n = 2$: domain (left), co-domain (right).

by choosing x_0, x_1 and x_2 such that $\Delta f_0 = \Delta f_1$, we could introduce a well-defined non-linearity in the secant information. However, we stated earlier that the columns of F_k defined in (10) must be linearly independent if generalized Broyden is used, so we may not choose the iterates such that $\Delta f_0 = \Delta f_1$. Instead we choose x_0, x_1 and x_2 such that Δf_0 and Δf_1 are slightly different, namely

$$\begin{aligned} f_1 - f_0 \equiv \Delta f_0 &= y + \delta y \\ f_2 - f_1 \equiv \Delta f_1 &= y \end{aligned} \tag{22}$$

where we assume that $\varepsilon \equiv \|\delta y\| \,/\, \|y\| \ll 1$ and $y^T \delta y = 0$, i.e. the difference δy is much smaller than Δf_1 and orthogonal to Δf_1. It may seem unconventional to put the δy term in Δf_0 instead of Δf_1; the reason is that this simplifies forthcoming calculations. The iterate differences are defined as

$$\begin{aligned} x_1 - x_0 \equiv \Delta x_0 &= s + \delta s + \Delta s \\ x_2 - x_1 \equiv \Delta x_1 &= s. \end{aligned} \tag{23}$$

The difference between Δx_1 and Δx_0 is split into two parts, namely δs and Δs. We choose the part δs to be caused purely by the difference between Δf_1 and Δf_0, i.e. it is a purely linear effect. Hence we define δs such that it corresponds to δy via the (unknown) Jacobian $\mathcal{F}'$ at x_2:

$$\delta y = \mathcal{F}'(x_2)\, \delta s. \tag{24}$$

The term Δs is then the non-linear effect related to the changing Jacobian and would be zero if $\mathcal{F}$ were linear. Figure 1 illustrates the relations between the iterates and the function values graphically for $n = 2$.

Based on the secant information $(\Delta x_0, \Delta f_0)$ and $(\Delta x_1, \Delta f_1)$, the approximate inverse Jacobian at $k = 2$ is calculated for both Anderson (20) and Broyden (21):

$$G_{2,\mathrm{A}} = \frac{s y^T}{\|y\|^2} + \frac{(\delta s + \Delta s)\, \delta y^T}{\|\delta y\|^2} \tag{25}$$

$$G_{2,\mathrm{B}} = \frac{s y^T}{\|y\|^2} + \left(\frac{(s + \delta s + \Delta s)(y + \delta y)^T}{\|y\|^2} \right) \left(I - \frac{y y^T}{\|y\|^2} \right) + \mathcal{O}(\varepsilon^2). \tag{26}$$

As we are not interested in the role of G_0, it was set to zero. The derivation of expressions (25) and (26) is given in Appendix A. In the upcoming analysis, we will neglect the $\mathcal{O}(\varepsilon^2)$ term in equation (26), as $\varepsilon \ll 1$.

To understand the difference between the Jacobian based on Anderson (25) and the one based on Broyden (26), we will analyze what values they predict when used in a quasi-Newton iteration. For this purpose, we will calculate the iterate change $\Delta x_2 = x_3 - x_2$ that is obtained from a quasi-Newton step $\Delta x_2 = -G_2 f_2$ as defined in (4), for both approximate inverse Jacobians. This is done for three different values of $-f_2$: the most recent function-value difference y, the previous one $y + \delta y$ and their difference δy. Table 1 summarizes the results. The values in the table are straightforward to derive, using the definition of ε and taking into account that $y \perp \delta y$.

The first row shows that the latest secant condition $G_2 \Delta f_1 = \Delta x_1$ is fulfilled for both methods, as required. On the second row it can be seen that for Anderson the previous secant condition $G_2 \Delta f_0 = \Delta x_0$ is fulfilled as well, as it should. For Broyden this is not the case: it uses a no-change condition for this f_2 instead of the previous secant condition. Because Δf_0 and Δf_1 are very close, the effect of the difference δy is effectively ignored (the ε^2 term is negligible) so that a vector along Δf_0 is treated the same as one along Δf_1 by $G_{2,\mathrm{B}}$.

$-f_2$	$\Delta x_2 = -G_{2,\mathrm{A}} f_2$	$\Delta x_2 = -G_{2,\mathrm{B}} f_2$
y	s	s
$y + \delta y$	$s + \delta s + \Delta s$	$s + \epsilon^2(s + \delta s + \Delta s)$
δy	$\delta s + \Delta s$	$\epsilon^2(s + \delta s + \Delta s)$

TABLE 1 Approximate effect of non-linearities present in secant information. The values represent the quasi-Newton step Δx_2 for different function values f_2, for both Anderson and Broyden.

For the third row f_2 lies along the direction of the difference δy. This yields the most interesting results, as Anderson and Broyden behave in a very different way here: both methods clearly have advantages and disadvantages. For Broyden, the update Δx_2 is negligible due to the presence of the ϵ^2 factor. The advantage is that a large non-linearity Δs has no effect on the convergence process. On the other hand, no use is made of the linear information δs. With Anderson, the linear information δs is correctly taken into account because all secant conditions are fulfilled, potentially speeding up convergence compared to Broyden. However, for non-linear functions Δs can be much larger than δs, so that Δx_2 potentially yields a large step in the wrong direction for Anderson. This may well introduce more non-linearities in the secant information, leading to instabilities in the quasi-Newton process.

We think it is reasonable to extend the observations from this model problem to the general use of the Anderson and Broyden schemes. Both are linear methods and can therefore not take non-linear behavior correctly into account, namely that the Jacobian $\mathcal{F}'$ depends on the vector x. However, Broyden seems to be better fit for dealing with non-linearities than Anderson: it will effectively ignore the non-linear information, while Anderson uses it actively due to its requirement to fulfill all secant conditions. On the other hand, Broyden ignores "small" linear information, which is correctly taken into account by Anderson and may therefore speed up convergence.

4 | A METHOD TO DEAL WITH NON-LINEARITIES

The previous section argues that both Anderson and Broyden have distinct advantages and disadvantages: Anderson uses the secant information more efficiently, but non-linearities can have an adverse effect; the opposite is true for Broyden. Therefore, we cannot conclude for which value of m generalized Broyden performs best: this will depend on the function $\mathcal{F}(x)$ and the initial guess x_0. In this section a strategy is outlined that seeks to prevent non-linearities from entering the secant information used to construct G_k for generalized Broyden. The goal is to improve convergence when m is large, but the method is equally applicable to smaller m.

Non-linearities influence G_k when they appear in the secant information $(\Delta x_i, \Delta f_i)$, as caused by the following two mechanisms. Firstly, the pair $(\Delta x_i, \Delta f_i)$ is representative for the Jacobian at point $\bar{x}_i$ as defined in equation (8), but may not be representative for the Jacobian at the current point x_k if the distance $\|x_k - \bar{x}_i\|$ is too large. Secondly, the points x_i and x_{i+1} may be so far apart that $(\Delta x_i, \Delta f_i)$ does not provide an accurate finite difference approximation to the Jacobian at point $\bar{x}_i$. In this case the distance $\|\Delta x_i\|$ is too large.

During the quasi-Newton iterations, both mechanisms can be checked: only secant information that corresponds to a close-enough point $\bar{x}_i$ and which is made up of points x_i and x_{i+1} that are near enough should be included in X_k and F_k. To check these criteria, a reference distance is required to compare $\|x_k - \bar{x}_i\|$ and $\|\Delta x_i\|$ with. This reference distance d must be obtained in advance, i.e. before the actual quasi-Newton iterations are started. An iterative procedure to calculate a suitable value for d is outlined next. Please note that these iterations should not be confused with the actual quasi-Newton iterations; to make a clear distinction, iterates and function values are therefore denoted with a hat.

We start by introducing the differences between iteration i and iteration 0 of this procedure as

$$\Delta \hat{x}_i \equiv \hat{x}_i - x_0 \tag{27}$$

where the inputs $\hat{x}_i$ are chosen—independently of the quasi-Newton iterations—such that consecutive iterate differences satisfy

$$\Delta \hat{x}_k = \sigma \Delta \hat{x}_{k-1} = \sigma^{k-1} \Delta \hat{x}_1 \qquad \text{with} \qquad \sigma > 1. \tag{28}$$

Subsequently, the output $\hat{f}_k = \mathcal{F}(\hat{x}_k)$ is calculated for each $\hat{x}_k$, upon which the difference with f_0 is calculated according to

$$\Delta\hat{f}_i \equiv \hat{f}_i - f_0. \tag{29}$$

For a linear function $\mathcal{F}$, the $\Delta\hat{f}_k$ corresponding with $\Delta\hat{x}_k$ would change with the same factor, hence one would obtain $\Delta\hat{f}_k = \sigma^{k-1}\Delta\hat{f}_1$. For a non-linear function $\mathcal{F}$, we expect this to be approximately true as long as $\Delta\hat{x}_k$ is small enough, as linear behavior is locally dominant. For larger $\Delta\hat{x}_k$, $\Delta\hat{f}_k$ will deviate from this linear behavior. A quantitative way to measure this deviation is provided by the coefficient

$$c_k = \frac{\left\|\sigma^{1-k}\,\Delta\hat{f}_k - \Delta\hat{f}_1\right\|}{\left\|\Delta\hat{f}_1\right\|} \tag{30}$$

which gives the relative size of the deviation in $\Delta\hat{f}_k$. When c_k exceeds a prescribed maximum value c_d, we can conclude that we have found the threshold for d. The distance d that corresponds to c_d can then be interpolated from the stored values of c_i and $\left\|\Delta\hat{x}_i\right\|$, e.g. with the linear interpolation rule

$$\begin{aligned} d &= \frac{c_d - c_k}{c_{k-1} - c_k}\left\|\Delta\hat{x}_{k-1}\right\| + \frac{c_d - c_{k-1}}{c_k - c_{k-1}}\left\|\Delta\hat{x}_k\right\| \\ &= \left(\frac{c_d - c_k}{c_{k-1} - c_k}\sigma^{k-2} + \frac{c_d - c_{k-1}}{c_k - c_{k-1}}\sigma^{k-1}\right)\left\|\Delta\hat{x}_1\right\|. \end{aligned} \tag{31}$$

For our algorithm to work properly, the value $\Delta\hat{x}_1$ must be small enough that the response $\Delta\hat{f}_1$ is representative for the Jacobian at point x_0. We propose to choose $\Delta\hat{x}_1$ in the direction $G_0 f_0$, scaled with a factor a to make it appropriately small. This procedure is detailed in Algorithm 2 and needs to be executed only once, in advance of the quasi-Newton iterations. The user can limit the number of iterations in Algorithm 2 to $k_{\max}$, ensuring that at most $k_{\max}$ additional function evaluations of $\mathcal{F}$ must be made to determine d. If the Jacobian changes more quickly in certain regions of the function $\mathcal{F}$ (i.e. large second-order derivatives), the distance d will not be representative further away from x_0 and the method will not correctly cope with non-linear secant information, limiting its applicability.

Algorithm 2 Determining the distance d, with parameter a chosen small enough such that the response to $\Delta\hat{x}_1$ is close to linear.

choose suitable values for a, σ and c_d
$f_0 = \mathcal{F}(x_0)$
$\Delta\hat{x}_1 = -aG_0 f_0$ ▷ use the same G_0 as chosen for the quasi-Newton iterations
$k = 1$
while $k \leq k_{\max}$ **do**
    $\Delta\hat{x}_k = \sigma^{k-1}\Delta\hat{x}_1$
    $\hat{x}_k = x_0 + \Delta\hat{x}_k$
    $\hat{f}_k = \mathcal{F}(\hat{x}_k)$
    c_k from definition (30)
    if $c_k \geq c_d$ **then**
        break
    end if
    $k = k + 1$
end while
d from definition (31)

Even though this procedure requires several expensive evaluations of $\mathcal{F}(x)$, improved convergence of the quasi-Newton iterations may justify its use for certain applications. Specifically, if many quasi-Newton iterations are performed, e.g. in nested iterations in time-integration processes, the computational overhead associated with Algorithm 2 may be negligible. This procedure will be demonstrated in Section 5 with recommended values for the parameters σ and c_d.

Using the distance d as determined by Algorithm 2, the generalized Broyden scheme from Algorithm 1 can be adapted to exclude secant information which has a too large value for $\left\|\Delta x_i\right\|$ or $\left\|x_k - \bar{x}_i\right\|$, resulting in Algorithm 3. Note that for the latter criterion a distance $d/2$ is used, which is in line with how d is determined and $\bar{x}_i$ is defined. The latest secant pair is always

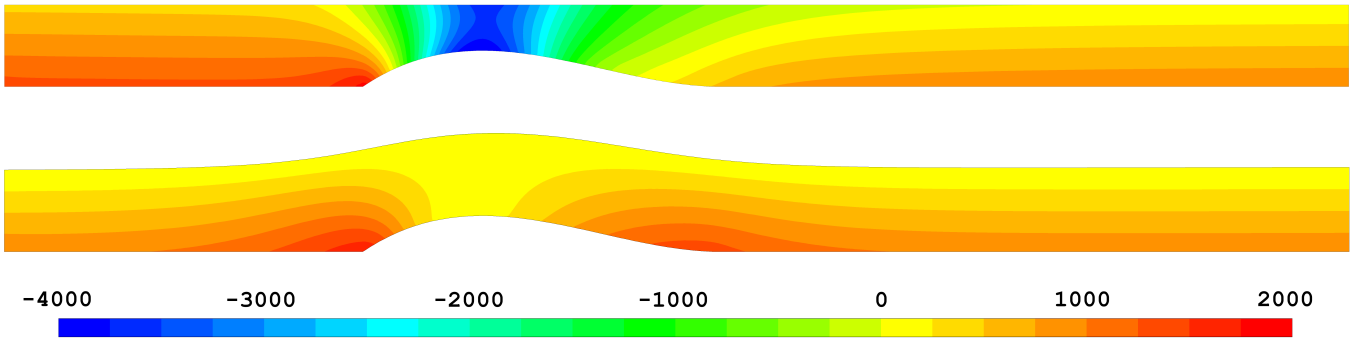

FIGURE 2 Pressure contours [Pa] for initial guess (top) and final solution (bottom). Flow from left to right.

added to the matrices X_l and F_l, because only non-linearities in the other secant pairs are amplified by Anderson according to the analysis in Section 3. As a consequence Algorithm 3 reduces to Algorithm 1 for $m = 1$.

Algorithm 3 Quasi-Newton with generalized Broyden, adapted to deal with non-linearities.

choose x_0, G_0, ε, m
get d using Algorithm 2
$x_1 = x_0 - G_0\mathcal{F}(x_0)$
$f_1 = \mathcal{F}(x_1)$
$k = 1$
while $\|f_k\| > \varepsilon$ **do**
  for $j \in \left[0, 1, \cdots, \text{floor}\left(\frac{k-1}{m}\right)\right]$ **do**
    $l = k - jm$
    $X_l = \left[\Delta x_{l-1}\right]$
    $F_l = \left[\Delta f_{l-1}\right]$
    for $i \in [2, 3, \cdots, \min(m, l)]$ **do**
      if $\|x_k - \bar{x}_{l-i}\| < d/2$ **and** $\|\Delta x_{l-i}\| < d$ **then**
        $X_l = \left[X_l\ \Delta x_{l-i}\right]$
        $F_l = \left[F_l\ \Delta f_{l-i}\right]$
      end if
    end for
  end for
  $x_{k+1} = x_k - G_k f_k$ ▷ evaluate recursively (matrix-free) using expression (16)
  $f_{k+1} = \mathcal{F}(x_{k+1})$
  $k = k + 1$
end while

5 | APPLICATION: SOLVING STEADY FREE-SURFACE FLOW

Introducing the test case

Steady free-surface flows are encountered in the fields of marine and hydraulic engineering. An important example is the calculation of the wave pattern around a ship that sails at constant speed. We can usually neglect the influence of the air for these cases, so that the steady free-surface problem reduces to a free boundary problem, viz. solving simultaneously for a single-phase flow with appropriate boundary conditions at the free surface, and the domain on which this flow is defined. The challenge is then to determine the free-surface position for which the corresponding flow field satisfies the governing partial differential equations

and all free-surface conditions. One way to do this is to solve a non-linear system $\mathcal{F}(x) = 0$, where x contains the (discretized) vertical position of the free-surface and $\mathcal{F}$ is a flow solver which returns the free-surface pressure $\mathcal{F}(x)$.[27] This corresponds to finding the free-surface shape that yields zero (atmospheric) pressure at the free-surface.[§]

Especially at higher flow velocities, it is important to take viscous and turbulent effects into account to obtain the correct shape. Hence the flow solver $\mathcal{F}$ must solve the Reynolds-averaged Navier-Stokes (RANS) equations. In this paper we use ANSYS Fluent for this purpose.

The test case we will consider is the 2D steady free-surface flow of water over an obstacle as introduced by Cahouet,[28] more specifically the flow with depth-based Froude number equal to 2.05 and Reynolds number 1.9×10^{-5}. The initial configuration and the converged solution are both shown in Figure 2, where the pressure fields as calculated in the flow solver are plotted. In the notation used throughout this paper, the position of the free-surface (the top boundary) corresponds to the iterate x, while the pressure at the free-surface corresponds to the function value $\mathcal{F}(x)$. The obstacle's height is given by

$$\frac{27}{4}\frac{H_\mathrm{b}}{L_\mathrm{b}^3}x(x - L_\mathrm{b})^2 \qquad \text{for} \qquad 0 \leq x \leq L_\mathrm{b} \tag{32}$$

with $L_b = 0.42\,\mathrm{m}$ and $H_b = 0.042\,\mathrm{m}$. The domain stretches from $-L_\mathrm{b}$ to $2.75L_\mathrm{b}$, the inlet height is 0.0955 m. At the inlet (left) a velocity profile is imposed,[28] at the outlet (right) a hydrostatic pressure, at the bottom a no-slip condition and at the free-surface a free-slip condition. The flow field is discretized with 57 600 cells, with $n = 481$ nodes at the free-surface. Turbulence modeling is done with the kω-SST model.[29] To avoid noise in the secant information, a convergence criterion of 10^{-12} was used in ANSYS Fluent for all conservation equations.

Adding a physics-based surrogate model

Section 2 conveyed that a surrogate model G_sur can replace $G_0 = -\beta I$ to improve the convergence of generalized Broyden. For steady free-surface flow, we base G_sur on an analytical investigation of the inviscid free-surface flow over a horizontal surface.[30] From this analysis follows a frequency domain relation between height differences Δx and pressure differences Δf. The discrete-time Fourier transform (DTFT) is used to transform an approximation of this frequency domain relation to the spatial domain. The convolution theorem is then employed to create a surrogate model.[31] The Python-code to create and use this surrogate is provided as supplementary material, as a detailed explanation goes beyond the scope of this paper.

Two variants of this surrogate model will be used: a *good* one denoted G_g, which accurately represents the analytical relation, and a *bad* one G_b, which uses a coarser approximation of the analytical relation.[¶] As a consequence, the latter model gives a bad prediction for some vectors in $\mathbb{R}^n$.

Apart from these two surrogate models, the free-surface problem will also be solved with the standard $G_0 = -\beta I$. The relaxation factor β is chosen based on the analytical investigation mentioned earlier, in such a way that iterations with simple relaxation ($G_k = -\beta I$) are stable while converging as fast as possible (details can be found in this paper[30]). For finer free-surface discretizations, β must decrease to retain stability, resulting in slower convergence of the quasi-Newton iterations.

Dealing with non-linear information

To remove secant information that might be contaminated by non-linearitites, the distance d as introduced in Section 4 is calculated using Algorithm 2. This must be done once, as a pre-computation before the start of the quasi-Newton iterations in Algorithm 3.

In Figure 3, the criterion c_k is plotted against the factor σ^{k-1} for $\sigma = 2$ and $\sigma = 10$. The value of c_1 is not shown as $c_1 = 0$ by definition (30) and therefore cannot be plotted on a logarithmic scale. The distance d can be interpolated from these data points using equation (31) based on either $\sigma = 2$ or $\sigma = 10$, giving practically the same result because c_k changes slowly. This shows that a large value can be used for σ to reduce the required number of function evaluations in Algorithm 2.

[§]The problem is actually a little more complicated, but this does not impact any of our results or conclusions. We need in fact only a *constant* free-surface pressure, not a zero one. Furthermore, a fixed inlet height is usually given as boundary condition to the problem. As the value of the constant pressure depends on the inlet height condition, this results in one less degree of freedom in both input and output of $\mathcal{F}$. In practice, this is dealt with by adapting $\mathcal{F}$ slightly: in each iteration a constant value is added to the input height to satisfy the inlet condition, and the average is subtracted from the output pressure. See this paper[27] for a more detailed explanation.

[¶]Note that the terms "good" and "bad" are no reference whatsoever to Broyden's first and second method; we use these adjectives in the literal sense, to distinguish between a very accurate and a rather poor surrogate model.

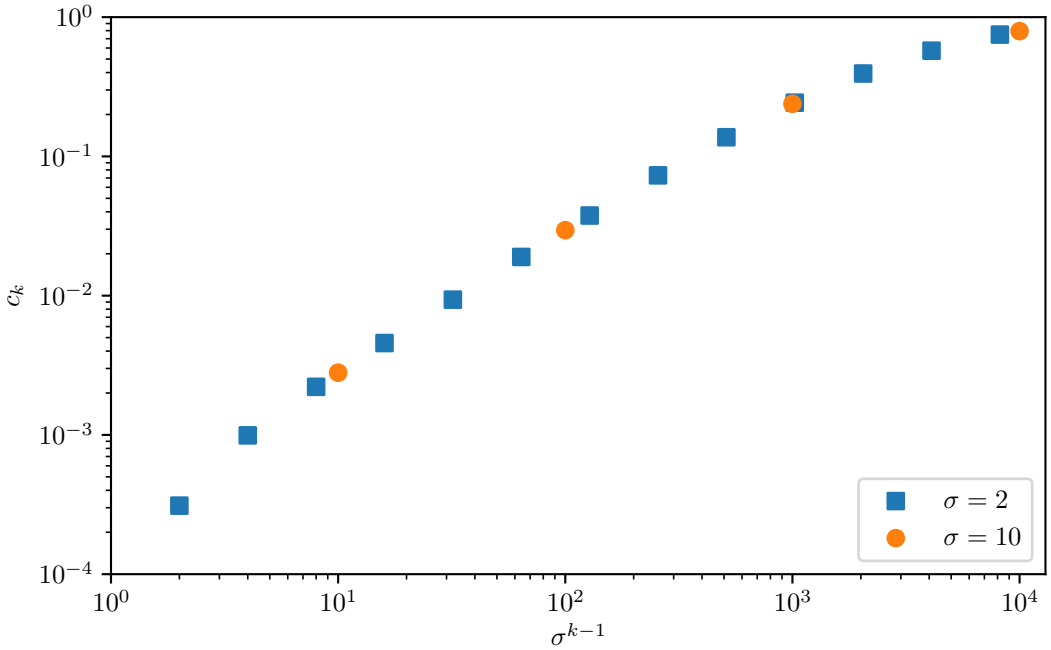

FIGURE 3 Determining the distance criterion d based on different values of σ.

The effect of the choice of c_d (and correspondingly d) on the convergence of Anderson acceleration is analyzed in Figures 4 and 5 for the steady free-surface problem. Figure 4 shows the residual $\|f_k\|$ normalized with respect to its initial value $\|f_0\|$. Note that the curve $c_d \to \infty$ corresponds to the original generalized Broyden scheme from Algorithm 1, where all secant information is used to construct G_k. Figure 5 summarizes the results of Figure 4 by plotting the average convergence speed as a function of c_d. The average convergence rate is based on the residual in iteration k and is defined as

$$\xi_k = -\frac{\log_{10} \frac{\|f_k\|_2}{\|f_0\|_2}}{k}. \tag{33}$$

It expresses (on a logarithmic scale) how much the residual decreases on average in each quasi-Newton iteration. If a large distance d is used ($c_d \geq 0.5$), convergence is poor. It is noteworthy that the convergence behavior is non-uniform in c_d: if all secant information is used ($c_d \to \infty$), the convergence behavior is better than for $c_d = 0.5$ and $c_d = 0.6$. We do not expect this to be a general trend, but rather a coincidence due to the unpredictable behavior of non-linearities in the secant information. For all $c_d \leq 0.4$, the convergence behavior is however significantly better and a clear trend does appear. For decreasing c_d, an optimal value $c_d = 0.4$ emerges for this application. When c_d decreases further, the initial convergence slows down, but the slope of the residuals curve in later iterations does not change. Furthermore, for lower c_d the residuals curve becomes smoother, which indicates that oscillations in the residual during the calculation are caused by the non-linear information picked up in the first few iterations.

The optimal choice for c_d will certainly be application dependent. As Figure 5 clearly demonstrates, a too high value is bad for convergence while a too low value has a very limited negative impact. Therefore we recommend a conservative choice $c_d \in [0.1, 0.2]$. In the remainder of this section, the distance d corresponding to $c_d = 0.2$ will be used for all results.

In the simulations reported in this paper, no filtering[6] was applied to remove linearly-dependent columns. This avoids interaction with the new method to remove non-linearities. In practice, these two techniques are expected to be complementary, as they have a different purpose and working principle.

Comparing all methods

The convergence of several methods is compared in Figure 6 for the test problem. The goal is to demonstrate the influence of the physics-based surrogate and the non-linearities on convergence behavior. The legend has several entries per method to

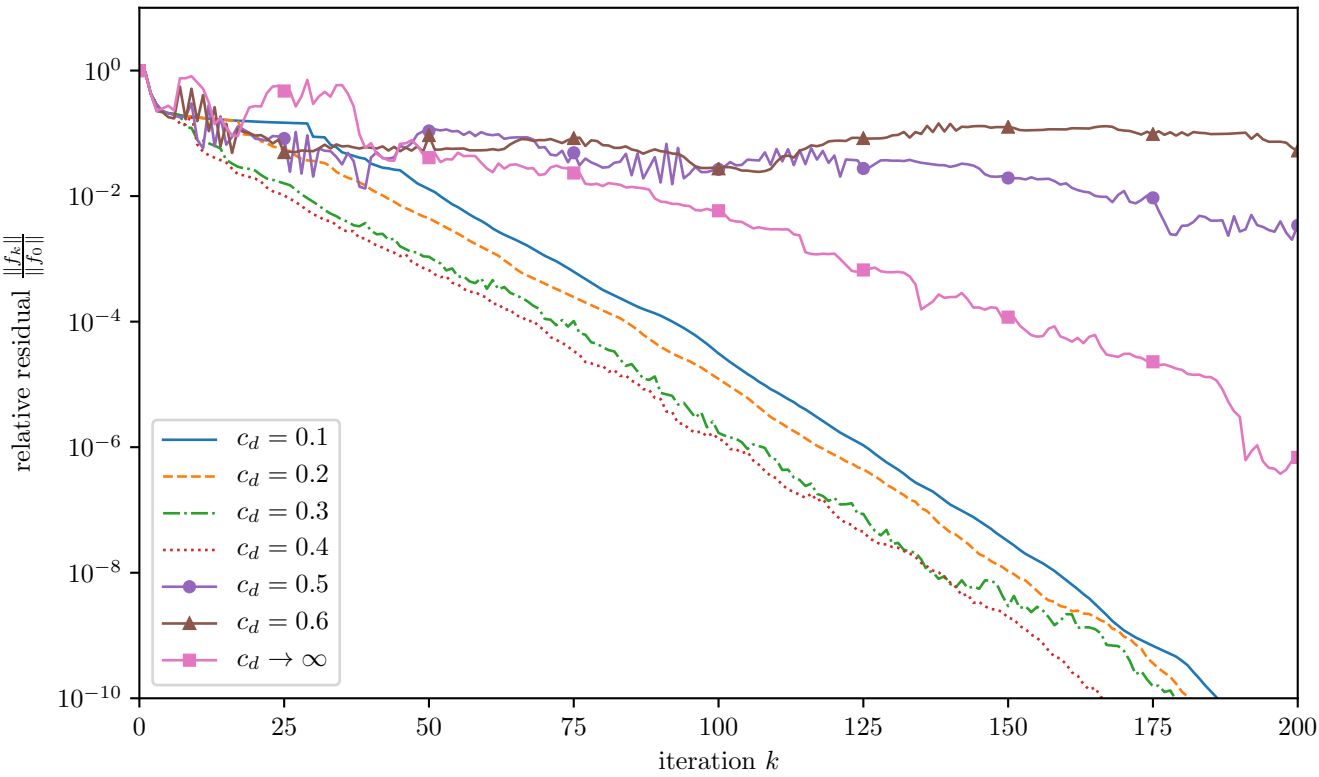

FIGURE 4 Convergence of Anderson (Algorithm 3 with $m = k$) for different values of c_d used in Algorithm 2.

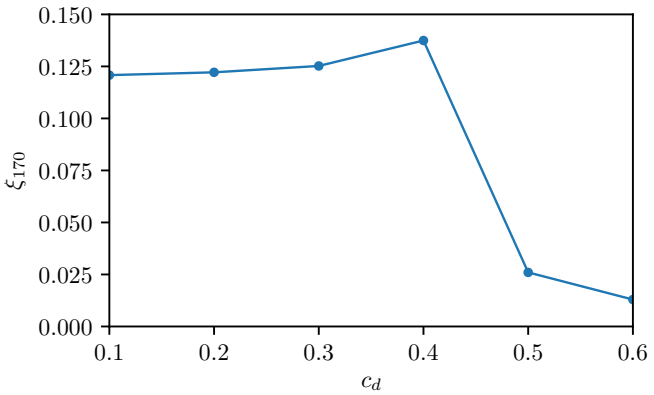

FIGURE 5 Comparison of the average convergence speed ξ_{170} for the calculations in Figure 4. Higher values correspond to faster convergence.

distinguish them. The first entry specifies which initial inverse Jacobian G_0 is used: a simple relaxation denoted by G_β, the good surrogate G_g or the bad surrogate G_b. The second entry specifies the method that is used to build G_k, which is either Anderson (20), Broyden (21) or $G_k = G_0$, denoted respectively as A, B and 0. In addition to these two entries, for the calculations with Anderson, the value of c_d is given to distinguish between the generalized Broyden method from Algorithm 1 and the adapted method from Algorithm 3.

Comparing the calculations with initial approximation $G_0 = G_\beta$, it can be seen that simple relaxation converges very slowly for this problem. Using Broyden accelerates convergence greatly. Anderson is significantly slower than Broyden as it has convergence problems initially and keeps oscillating later on. When we use the distance d that corresponds to $c_k = 0.2$, to keep

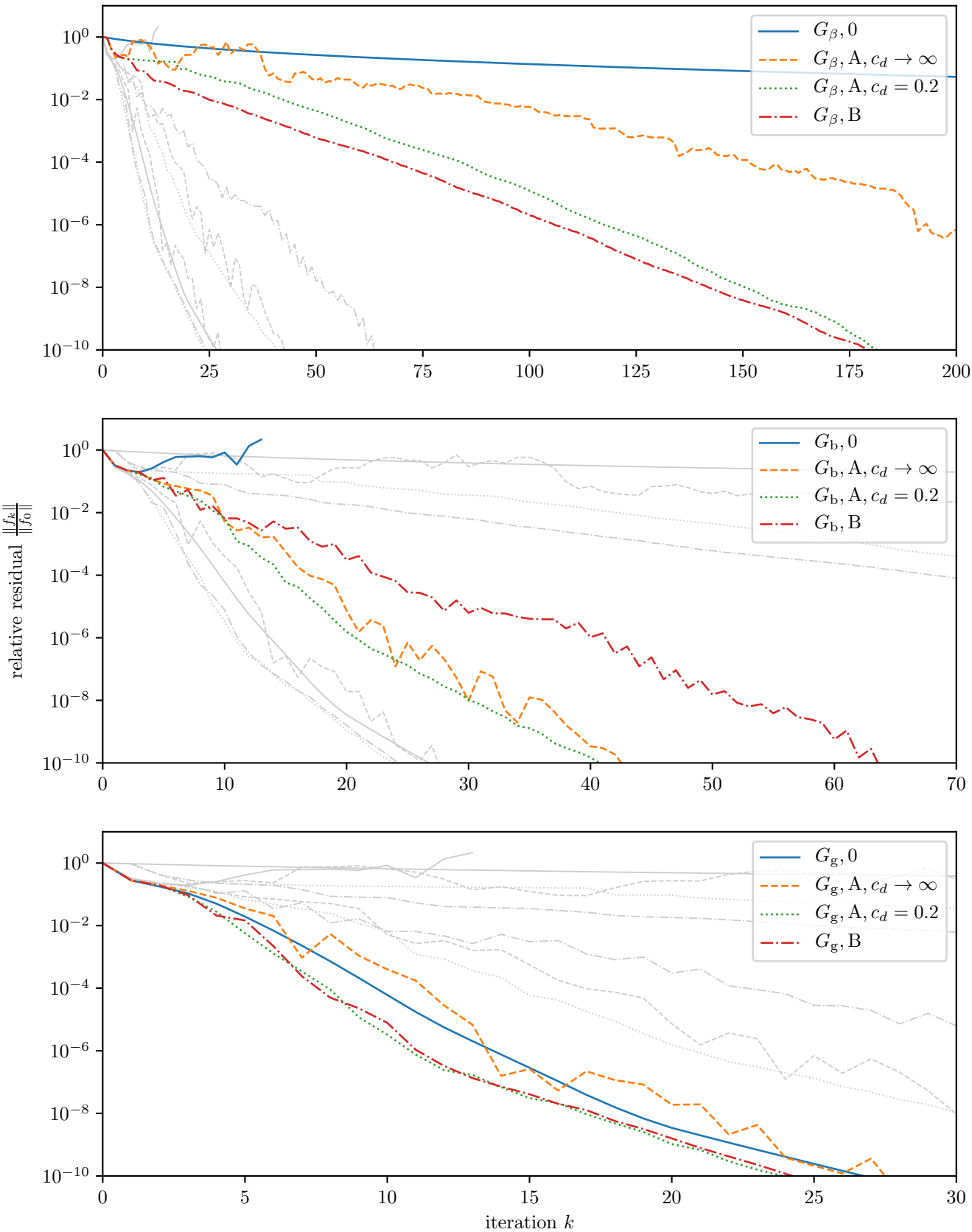

FIGURE 6 Convergence comparison for a range of quasi-Newton methods. The number of iterations equals the number of function calls $\mathcal{F}(x)$, except for the dotted lines where the execution of Algorithm 2 requires 5 additional function calls. The three plots focus on respectively the simulations with the simple relaxation G_β, the bad surrogate G_b and the good surrogate G_g.

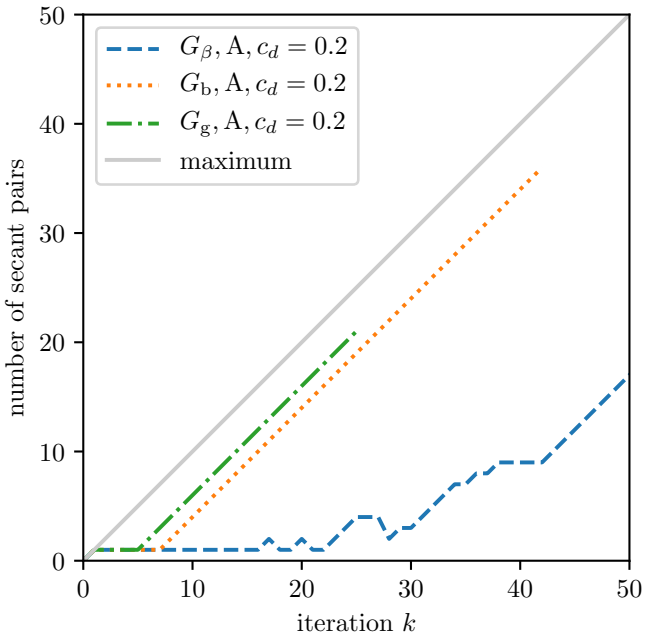

FIGURE 7 Number of secant pairs used to construct G_k when $c_d = 0.2$. The gray line represents the maximum available number of pairs in each iteration.

non-linear information from appearing in X_k and F_k, this behavior changes completely: Anderson then converges with the same speed as Broyden, although initial convergence is a bit slower. These results confirm the conclusions from Section 3 and the effectivity of the solution proposed in Section 4.

With the bad surrogate G_b as initial approximation, there are significant differences in the results. With simple relaxation, the iterations in fact diverge because G_b is inherently unstable. It is noteworthy that both Anderson and Broyden restore stability of the iteration. Comparing Anderson and Broyden, the former is faster in this case, perhaps because Anderson makes better use of small linear information. Removing non-linear information makes less of a difference here than with $G_0 = G_\beta$. The reason is most likely that the number of non-linear secant pairs is smaller to begin with, as convergence is very fast from the start. For Anderson enhanced with a distance criterion, the convergence becomes smoother though, as was observed with $G_0 = G_\beta$, too.

All methods that use the good surrogate G_g converge very fast in approximately the same number of iterations. The surrogate is such a good approximation of the actual Jacobian that the use of secant information yields very little improvement in convergence. The wiggles in Anderson again disappear with $c_d = 0.2$.

In the calculations with the adapted Anderson method from Algorithm 3, not all the secant information is used to construct G_k. Figure 7 plots the number of used secant pairs as a function of the iteration counter, for the three calculations with $c_d = 0.2$ in Figure 6. The available number of pairs is equal to the iteration count, shown as a gray line on the figure. Note that for the calculations with $c_d \to \infty$, this maximum number of pairs is used. The figure shows that for $c_d = 0.2$, only a single secant pair (which is the minimum) is used in the first few iterations, because the steps are large. For the calculations with the good and the bad surrogate, the behavior of the algorithm abruptly changes after respectively 5 and 7 iterations: as the method is converging well, the steps Δx_i become small and the iterate x_i sees only small variations, hence from this point on, all new secant pairs are kept. For $G_0 = -\beta I$, the change is more gradual, as the calculation is not converging as quickly. It can thus be observed that the behavior of the non-linearity criterion is different from a fixed window.

Figure 7 also helps to understand the effect on convergence of adding the distance criterion: the residuals converge slowly at first, but faster and smoother in a later stage. The slow initial convergence is caused by the fact that only a single secant pair is used in the first iterations. The smoother convergence later is caused by the absence of the first few secant pairs, which contain significant non-linearities. This last observation corroborates the conclusion drawn about Anderson in Section 3, namely that it suffers much more from non-linearities than Broyden.

Comparing all the results, it is clear that the choice of the initial inverse Jacobian approximation has the largest impact on convergence: the more of the physics is captured by the surrogate, the better. The performance comparison between Anderson and Broyden is inconclusive. Figure 6 demonstrates that the best choice does not only depend on the test-case, but also on the surrogate and whether non-linear secant information is retained or removed.

6 | CONCLUSIONS

This paper addresses two important aspects of the generalized Broyden method: the initial Jacobian approximation and the presence of non-linearities in the secant information. A steady free-surface-flow problem was used to compare several generalized Broyden variants and to support the analyses made in the paper.

It was shown how the generalized Broyden method can be extended naturally by using a physics-based surrogate model as initial approximate Jacobian. Depending on the application, this has the potential to greatly reduce the required number of quasi-Newton iterations. This statement was confirmed by the free-surface test case, even for a surrogate model that gave a bad prediction for some inputs.

To investigate the effect of non-linearities in the secant information, the two limiting cases of generalized Broyden were compared: Anderson acceleration and Broyden's second method, which satisfy respectively all secant equations and only a single one. It was shown that Broyden neglects small linear information which may lead to slower convergence, but more importantly that Anderson acceleration amplifies non-linearities present in the secant information. A method was therefore proposed to prevent non-linear information from being used in the construction of the inverse Jacobian in generalized Broyden. Note that the new technique is equally applicable to more specific methods, like Anderson acceleration or IQN-ILS. The test case showed that non-linearities indeed give convergence issues for Anderson, which can be effectively dealt with using the proposed method. Even though this method may not be optimal yet—it requires some extra function calls—applying it to the test case clearly demonstrated the predicted behavior, which substantiates our conclusions about non-linearities in generalized Broyden. As the new method was developed to work for generalized Broyden, it is equally applicable to submethods such as Anderson acceleration and IQN-ILS.

ACKNOWLEDGMENTS

This work was funded by the Special Research Fund (BOF) of Ghent University.

DATA AVAILABILITY STATEMENT

The data that support the findings of this study are available from the corresponding author upon reasonable request.

References

1. Anderson DG. Iterative procedures for nonlinear integral equations. *Journal of the ACM* 1965; 12(4): 547–560. doi: 10.1145/321296.321305

2. Broyden CG. A class of methods for solving nonlinear simultaneous equations. *Mathematics of Computation* 1965; 19(92): 577–593. doi: 10.1090/S0025-5718-1965-0198670-6

3. Eyert V. A comparative study on methods for convergence acceleration of iterative vector sequences. *Journal of Computational Physics* 1996; 124(2): 271–285. doi: 10.1006/jcph.1996.0059

4. Fang Hr, Saad Y. Two classes of multisecant methods for nonlinear acceleration. *Numerical Linear Algebra with Applications* 2009; 16(3): 197–221. doi: 10.1002/nla.617

5. Trefethen LN, Bau III D. *Numerical linear algebra*. 50. Siam . 1997.

6. Haelterman R, Bogaers AE, Scheufele K, Uekermann B, Mehl M. Improving the performance of the partitioned QN-ILS procedure for fluid–structure interaction problems: Filtering. *Computers & Structures* 2016; 171: 9–17. doi: 10.1016/j.compstruc.2016.04.001

7. Marks L, Luke D. Robust mixing for ab initio quantum mechanical calculations. *Physical Review B* 2008; 78(7): 075114. doi: 10.1103/PhysRevB.78.075114

8. Gilbert JC, Lemaréchal C. Some numerical experiments with variable-storage quasi-Newton algorithms. *Mathematical Programming* 1989; 45(1): 407–435. doi: 10.1007/BF01589113

9. Brust J, Burdakov O, Erway JB, Marcia RF. A dense initialization for limited-memory quasi-Newton methods. *Computational Optimization and Applications* 2019; 74(1): 121–142. doi: 10.1007/s10589-019-00112-x

10. Vanderbilt D, Louie SG. Total energies of diamond (111) surface reconstructions by a linear combination of atomic orbitals method. *Physical Review B* 1984; 30(10): 6118. doi: 10.1103/PhysRevB.30.6118

11. Johnson DD. Modified Broyden's method for accelerating convergence in self-consistent calculations. *Physical Review B* 1988; 38(18): 12807. doi: 10.1103/PhysRevB.38.12807

12. van Leuken H. *Electronic structure of metallic multilayers*. PhD thesis. University of Amsterdam, 1991.

13. Haelterman R, Degroote J, Van Heule D, Vierendeels J. On the similarities between the quasi-Newton inverse least squares method and GMRes. *SIAM Journal on Numerical Analysis* 2010; 47(6): 4660–4679. doi: 10.1137/090750354

14. Michler C, van Brummelen EH, De Borst R. An interface Newton–Krylov solver for fluid–structure interaction. *International Journal for Numerical Methods in Fluids* 2005; 47(10-11): 1189–1195. doi: 10.1002/fld.850

15. Vierendeels J, Lanoye L, Degroote J, Verdonck P. Implicit coupling of partitioned fluid–structure interaction problems with reduced order models. *Computers & structures* 2007; 85(11): 970–976. doi: 10.1016/j.compstruc.2006.11.006

16. Degroote J, Bathe KJ, Vierendeels J. Performance of a new partitioned procedure versus a monolithic procedure in fluid–structure interaction. *Computers & Structures* 2009; 87(11): 793–801. doi: 10.1016/j.compstruc.2008.11.013

17. Haelterman R, Degroote J, Van Heule D, Vierendeels J. The quasi-Newton least squares method: a new and fast secant method analyzed for linear systems. *SIAM Journal on Numerical Analysis* 2009; 47(3): 2347–2368. doi: 10.1137/070710469

18. Bogaers AEJ, Kok S, Reddy B, Franz T. Quasi-Newton methods for implicit black-box FSI coupling. *Computer Methods in Applied Mechanics and Engineering* 2014; 279: 113–132. doi: 10.1016/j.cma.2014.06.033

19. Lindner F, Mehl M, Scheufele K, Uekermann B. A comparison of various quasi-Newton schemes for partitioned fluid-structure interaction. In: ; 2015: 1–12.

20. Blom D, Zuijlen vA, Bijl H. Multi-level acceleration with manifold mapping of strongly coupled partitioned fluid–structure interaction. *Computer Methods in Applied Mechanics and Engineering* 2015; 296: 211–231. doi: 10.1016/j.cma.2015.08.004

21. Scheufele K. Robust Quasi-Newton methods for partitioned fluid-structure simulations. Master's thesis. University of Stuttgart. 2015.

22. Uekermann BW. *Partitioned fluid-structure interaction on massively parallel systems*. PhD thesis. Technische Universität München, 2016.

23. Gay DM. Some convergence properties of Broyden's method. *SIAM Journal on Numerical Analysis* 1979; 16(4): 623–630. doi: 10.1137/0716047

24. Walker HF, Ni P. Anderson acceleration for fixed-point iterations. *SIAM Journal on Numerical Analysis* 2011; 49(4): 1715–1735. doi: 10.1137/10078356X

25. Toth A, Kelley CT. Convergence Analysis for Anderson Acceleration. *SIAM Journal on Numerical Analysis* 2015; 53(2): 805–819. doi: 10.1137/130919398

26. Martınez JM. Practical quasi-Newton methods for solving nonlinear systems. *Journal of Computational and Applied Mathematics* 2000; 124(1-2): 97–121. doi: 10.1016/S0377-0427(00)00434-9

27. Demeester T, van Brummelen EH, Degroote J. An efficient quasi-Newton method for two-dimensional steady free surface flow. *International Journal for Numerical Methods in Fluids* 2020; 92: 785–801. doi: 10.1002/fld.4806

28. Cahouet J. *Etude numérique et expérimentale du problème bidimensionnel de la résistance de vagues non-linéaire*. Ecole Nationale Supérieure de Techniques Avançées . 1984.

29. Menter FR. Two-equation eddy-viscosity turbulence models for engineering applications. *AIAA Journal* 1994; 32(8): 1598–1605. doi: 10.2514/3.12149

30. Demeester T, Degroote J, Vierendeels J. Stability analysis of a partitioned iterative method for steady free surface flow. *Journal of Computational Physics* 2018; 354: 387–392. doi: 10.1016/j.jcp.2017.10.053

31. Demeester T, van Brummelen EH, Degroote J. Extension of a fast method for 2D steady free surface flow to stretched surface grids. In: ; 2019: 235–246.

□

APPENDIX

A DERIVATION OF EXPRESSIONS (25) AND (26)

This appendix gives the step-by-step derivation of expressions (25) and (26), based on the secant information defined in (22) and (23) that is collected in the matrices

$$X_2 = \begin{bmatrix} s & s+\delta s+\Delta s \end{bmatrix} \tag{A1}$$

$$F_2 = \begin{bmatrix} y & y+\delta y \end{bmatrix}. \tag{A2}$$

For $G_{2,A}$ we need the economy-size QR decomposition of F_2. This is made easy by the choice of secant pairs:

$$F_2 = \begin{bmatrix} y & y+\delta y \end{bmatrix} = \begin{bmatrix} \frac{y}{\|y\|} & \frac{\delta y}{\|\delta y\|} \end{bmatrix} \begin{bmatrix} \|y\| & \|y\| \\ 0 & \|\delta y\| \end{bmatrix} = Q_2 R_2. \tag{A3}$$

We can now invert R_2:

$$R_2^{-1} = \frac{1}{\|y\|\,\|\delta y\|} \begin{bmatrix} \|\delta y\| & -\|y\| \\ 0 & \|y\| \end{bmatrix} = \begin{bmatrix} \frac{1}{\|y\|} & -\frac{1}{\|\delta y\|} \\ 0 & \frac{1}{\|\delta y\|}. \end{bmatrix} \tag{A4}$$

Expression (25) then follows from expression (20) with $G_0 = 0$:

$$G_{2,A} = X_2 R_2^{-1} Q_2^T \tag{A5}$$

$$= \begin{bmatrix} s & s+\delta s+\Delta s \end{bmatrix} \begin{bmatrix} \frac{1}{\|y\|} & -\frac{1}{\|\delta y\|} \\ 0 & \frac{1}{\|\delta y\|} \end{bmatrix} \begin{bmatrix} \frac{y^T}{\|y\|} \\ \frac{\delta y^T}{\|\delta y\|} \end{bmatrix} \tag{A6}$$

$$= \frac{s y^T}{\|y\|^2} + \frac{(\delta s+\Delta s)\delta y^T}{\|\delta y\|^2}. \tag{A7}$$

For $G_{2,B}$ we use the recursive formula (21), hence we first calculate $G_{1,B}$:

$$G_{1,B} = \frac{\Delta x_0 \Delta f_0^T}{\|\Delta f_0\|^2} = \frac{(s+\delta s+\Delta s)(y^T+\delta y^T)}{\|y+\delta y\|^2} = \frac{(s+\delta s+\Delta s)(y^T+\delta y^T)}{(1+\varepsilon^2)\|y\|^2}. \tag{A8}$$

In the last step the expression

$$\|y+\delta y\|^2 = \|y\|^2 + \|\delta y\|^2 = \left(1+\varepsilon^2\right)\|y\|^2 \tag{A9}$$

can be used, because $y \perp \delta y$. Expression (26) is then given by:

$$G_{2,B} = \frac{\Delta x_1 \Delta f_1^T}{\|\Delta f_1\|^2} + G_{1,B}\left(I - \frac{\Delta f_1 \Delta f_1^T}{\|\Delta f_1\|^2}\right) \tag{A10}$$

$$= \frac{sy^T}{\|y\|^2} + \frac{(s + \delta s + \Delta s)(y^T + \delta y^T)}{(1+\varepsilon^2)\|y\|^2}\left(I - \frac{yy^T}{\|y\|^2}\right) \tag{A11}$$

$$= \frac{sy^T}{\|y\|^2} + \frac{(s + \delta s + \Delta s)(y^T + \delta y^T)}{\|y\|^2}\left(I - \frac{yy^T}{\|y\|^2}\right) + \mathcal{O}(\varepsilon^2). \tag{A12}$$