Abstract

We consider iterative methods for approximating solutions of nonlinear equations, where the iteration cannot be computed exactly, but is corrupted by additive perturbations. The cost of computing each iteration depends on the size of the perturbation. For a class of cost functions, we show that the total cost of producing an $\varepsilon$-approximation can be made proportional to the cost $c(\varepsilon)$ of one single iterative step performed with the accuracy proportional to $\varepsilon$. We also demonstrate that for some cost functions the total cost is proportional to $c(\varepsilon)^2$. In both cases matching lower bounds are shown. The results find natural application to establishing the complexity of nonlinear boundary-value problems, where they yield an improvement over the known upper bounds, and remove the existing gap between the upper and lower bounds.

1. Introduction

We deal in this paper with the iterative solution of nonlinear equations, a subject that has been extensively studied in the framework of applied mathematics, optimization, engineering, etc., and has a huge bibliography devoted to both theory and applications. It is not our aim to survey here numerous research directions and results within the field. We only observe that a major part of the literature on the subject is concerned with the case when iterations constructed by a method are computed exactly, i.e., the only error that we deal with is that of the method itself, and no other possible perturbations of the process are considered. Such a case is
discussed, for instance, in the classical monograph [2], which is still an important source of fundamental results in the theory of numerical methods for nonlinear equations and minimization problems, or in the recent book [3], which contains the complexity analysis of nonlinear equations and provides an extensive up-to-date bibliography.

In real-life computations, we rarely deal with exact methods; in many cases we are forced to use their perturbed versions. In spite of this, the number of papers devoted to perturbed iterations is relatively small. Round-off errors, for example, introduce perturbations that spoil theoretical properties of a method and make further analysis necessary, see, e.g., [4] or [5] for results on stability of iterative methods for nonlinear or linear equations. Computing the degree of a mapping, the problem closely related to that of solving nonlinear equations, is analyzed in the perturbed case in [6].

In this paper we analyze the cost of iterative methods in the presence of perturbations. In many cases, the perturbations are not entirely out of our control; their size can be controlled or even selected. A typical example is the shooting method for solving boundary-value problems: the iterative process involved requires information that can be computed only approximately by means of an initial value problem solver. The constraint here is the amount of computational work (a cost) that we wish to expend in computing successive iterations (this example is discussed later in more detail). Another example could be the computation of successive iterations in variable precision arithmetic.

In general, we assume that the cost of computing each iteration depends on the size of the perturbation. For a given \( \varepsilon > 0 \), the aim is to compute an \( \varepsilon \)-approximation to the solution of a nonlinear equation with minimal total cost, by properly selecting the precision at which each successive iteration is computed. For superlinearly or linearly convergent methods, we show for a class of cost functions \( c(\cdot) \) that the total cost of producing an \( \varepsilon \)-approximation can be made proportional to the cost \( c(\varepsilon) \) of one single iterative step performed with accuracy proportional to \( \varepsilon \) (Theorems 1 and 2). We also demonstrate that for some cost functions the total cost is proportional to \( c(\varepsilon)^2 \). These bounds cannot be improved: in both cases matching lower bounds are shown (Theorem 3).

The obtained results may find an application in analyzing methods that involve, as a building block, the solution of nonlinear equations. In Section 5, we discuss in more detail an application of this type, the problem of establishing the complexity of nonlinear boundary-value problems. This topic has been recently studied in [1], where upper and lower bounds on the complexity are shown. The bounds, although close, differ by a logarithmic factor. We show that the results of this paper yield an improvement over the upper bound, and remove the existing gap between the upper and lower bounds.

2. Perturbed iterative methods

We deal with the iterative solution of nonlinear equations

\[
F(x) = 0,
\] (1)
where $F : \Omega \to \mathbb{R}^d$ and $\Omega$ is an open subset of $\mathbb{R}^d$. We assume that (1) has a solution $\alpha$ in $\Omega$, and that the function $F$ belongs to a certain class $\mathcal{F}$, usually defined by suitable regularity conditions.

We shall consider iterative methods for solving (1) with superlinear or linear convergence. For a given initial approximation $\hat{x}_0$, the successive approximations are given by

$$
\hat{x}_{s+1} = \phi(\hat{x}_s, F), \quad s = 0, 1, \ldots, (2)
$$

for some transformation $\phi$, see [2] for numerous possibilities of defining $\phi$. For superlinearly convergent methods, it is usually shown that there exists $R > 0$ such that if $\hat{x}_0 \in K(\alpha, R)$, then the sequence $\{\hat{x}_s\}$ is well defined, it remains in $K(\alpha, R)$, $\lim_{s \to \infty} \hat{x}_s = \alpha$, and the errors $\hat{e}_s = ||\hat{x}_s - \alpha||$ satisfy

$$
\hat{e}_{s+1} \leq \frac{1}{2} S \hat{e}_s^p, \quad s = 0, 1, \ldots (3)
$$

for some $p > 1$, and a constant $S > 1$ (the coefficient $\frac{1}{2}$ is here only for convenience). This means that $\phi$ has order of convergence at least $p$ (see [2]). The ball $K(\alpha, R)$ is called the ball of convergence of $\phi$. In the case of (at least) linear convergence, we have $\lim_{s \to \infty} \hat{x}_s = \alpha$, and

$$
\hat{e}_{s+1} \leq q \hat{e}_s, (4)
$$

for some constant $q \in (0, 1)$, and for (usually) an arbitrary $\hat{x}_0$.

Examples of both superlinearly and linearly convergent methods are numerous and well known. For a classical discussion of the theory, see [2]; the complexity point of view is analyzed in [3]. We shall study in this paper iterative methods in the presence of perturbations. For a given initial approximation $x_0$, the successive approximations are given by

$$
x_{s+1} = \phi(x_s, F) + \gamma_s, \quad s = 0, 1, \ldots, (5)
$$

where $\gamma_s$ is a perturbation corrupting an iteration. We assume that

$$
||\gamma_s|| \leq \frac{1}{2} w \delta, (6)
$$

where $w = 1$ for superlinearly convergent methods, and $w = 1 - q$ for linearly convergent methods (the coefficient $\frac{1}{2} w$ is used as a convenient normalization). The parameter $\delta$ is a given positive number (for a while independent of $s$) that controls the size of perturbations $\gamma_s$.

Let $e_s = ||x_s - \alpha||$ denote the error of the perturbed sequence. It is easy to show by induction that, for superlinearly convergent methods, if $S^{1/(p-1)} R \leq 1, \delta \leq R$ and $x_0 \in K(\alpha, R)$, then $\{x_s\} \subseteq K(\alpha, R)$, and therefore

$$
e_{s+1} \leq \frac{1}{2} S e_s^p + \frac{1}{2} \delta, \quad s = 0, 1, \ldots (7)
$$

In the case of linear convergence, for any $R > 0$ and $\delta \leq 2R$ the sequence remains in $K(\alpha, R)$ if the starting point does, and

$$
e_{s+1} \leq q e_s + \frac{1}{2} (1 - q) \delta. (8)
$$

In real-life computations we usually have $\gamma_s \neq 0$. For instance, perturbations may arise from round-off errors, or they may appear when the information about $F$
needed to construct $\phi(x_s, F)$ can be computed only approximately. In many cases, the accuracy $\delta$ can be controlled; the only constraint is the amount of computational work or memory (a cost) that we wish to involve in computing $x_{s+1}$. An illustration of such a situation is discussed in Section 5.

2.1. The cost function and the total cost

We shall associate with formula (5) the positive-valued cost function $c(\delta)$ defined for positive $\delta$. That is, we assume that the computation of $\phi(x_i, F)$ with accuracy $\frac{1}{2}\delta$ in the superlinear case, or $\frac{1}{2}(1 - q)\delta$ in the linear case, can be performed at cost $c(\delta)$. We assume that $c(\delta)$ is a decreasing function, and $\lim_{\delta \to 0^+} c(\delta) = +\infty$. This means that the cost grows as $\delta$ decreases, and approaches infinity as $\delta \to 0^+$. (Thus, the exact value of $\phi(x_i, F)$ is not available.) Consider the following examples.

Below, $C$ stands for a positive constant independent of $\delta$, and $\log = \log_2$.

1. $c(\delta) = C(1/\delta)^m$, where $m$ is a positive number. This cost function appears, e.g., when considering shooting-type methods for solving nonlinear boundary-value problems, see Section 5.

2. $c(\delta) = C\log 1/\delta$. This measure of cost is proportional to the number of binary bits necessary to represent the value of $\phi(x_i, F)$ with the precision $\delta$. In practical computations, a choice of $\delta$ is possible if our computing device is equipped with variable precision arithmetic.

3. $c(\delta) = C\log\log 1/\delta$. Although this example does not seem to have an obvious interpretation, it is included to illustrate a nonlinear lower bound for superlinearly convergent methods that will be shown in the sequel.

Our aim is to minimize the total cost of approximating $x$ to within a prescribed error, using the iterative method $\phi$. More precisely, let $\varepsilon$ be a given positive number representing the desired accuracy, and let $\{\delta_i\}$ be a decreasing positive sequence converging to $0$, with $\delta_0 = R$. Let $k$ be the minimal number such that $\delta_k \leq \varepsilon$. We apply the method $\phi$ as follows. For an initial approximation $x_0$ with $e_0 \leq \delta_0$, we arrive at an approximation $x_k$ such that $e_k \leq \varepsilon$ in $k$ stages. The $i$th stage $(i = 1, 2, \ldots, k-1)$ consists of $l_i$ iterative steps (5) performed with the (constant within the stage) precision $\delta = \delta_i$. This stage starts from the approximation $x_{i-1}$ having error $e_{i-1} \leq \delta_{i-1}$, and ends with the approximation $x_i$ with the error $e_i \leq \delta_i$. That is, the $i$th stage looks as follows:

\[
\begin{align*}
\text{start:} & \quad z_0^i = x_{i-1}, \\
\text{iteration:} & \quad z_{s+1}^i = \phi(z_s^i, F) + \gamma_s^i, \quad s = 0, 1, \ldots, l_i - 1, \quad \text{with} \quad \|\gamma_s^i\| \leq \frac{1}{2}l_i \delta_i \\
& \quad x_i = z_{l_i}^i.
\end{align*}
\]

The $k$th stage is somewhat nontypical, and consists of applying (9) with the final accuracy $\delta_k = \varepsilon$. We shall find in the next section the sufficient number of iterations $l_i$ that allows to reduce the error from $\delta_{i-1}$ to $\delta_i$. The total cost of getting the $\varepsilon$-approximation $x_k$ is given by

\[
TC(\varepsilon) = l_1 c(\delta_1) + l_2 c(\delta_2) + \cdots + l_{k-1} c(\delta_{k-1}) + l_k c(\varepsilon).
\]
The total cost depends on the selection of the intermediate precisions \( \delta_i \). For example, in extreme cases, we can pass from \( \delta_0 \) to \( \epsilon \) in one single stage consisting of \( l_1 \) steps or, on the contrary, we can perform only one step within each stage. Each selection of \( \{ \delta_i \} \) leads to a certain value of \( \text{TC}(\epsilon) \). We shall find a selection of \( \{ \delta_i \} \) that is optimal, in the sense that for the resulting value of \( \text{TC}(\epsilon) \) there is a matching lower bound. In order to find such a set \( \{ \delta_i \} \), we first analyze the error of iterative methods in the presence of perturbations.

2.2. Superlinearly convergent methods

Define the sequence \( \{ x_i \} \) by

\[
 x_0 = 0, \quad x_i = S x_{i+1}^p, \quad i = 0, 1, \ldots .
\]

Letting \( S_1 = S^{1/(p-1)} \) we easily see that \( \lim_{i \to \infty} x_i = 1/S_1 \). Moreover, if \( S_1 \delta \leq 1 \) (which we assume in the sequel), then \( x_i \leq 1/S_1 \) and the sequence \( \{ x_i \} \) is nondecreasing. The following two useful lemmas will serve as the basis for the analysis of perturbed iterations. We first observe that there is a simple (forward–backward) relation between the sequence of errors of the perturbed iteration \( \{ e_i \} \) and the sequence \( \{ x_i \} \).

**Lemma 1.** Let \( S_1 \delta \leq 1 \). If the initial approximation satisfies \( e_0 \leq x_0 \), then for \( i = 0,1, \ldots, 1 \) we have

\[
e_i \leq x_{i-j}.
\]

**Proof.** For \( l = 0 \) the lemma obviously holds true. Let \( l \geq 1 \). For \( i = 0 \) the implication is satisfied. Assuming that \( e_i \leq x_{i-j} \) for some \( i (i < l) \), we have that

\[
e_{i+1} \leq 1/2 x_{i+1}^p + 1/2 \delta \leq 1/2 S x_{i+1}^p + 1/2 x_0 = 1/2 x_{i-j-1} + 1/2 x_0 \leq x_{i-j-1},
\]

which completes the proof. □

A similar result for Newton’s method has been exploited in [1]. Based on Lemma 1, we now establish a sufficient number of iterations that allows us to obtain an approximation with the error \( \delta_{\text{new}} \), provided that the starting approximation lies within the distance \( \delta_{\text{old}} \) from the solution.

**Lemma 2.** Consider an iterative method of the form (5) with errors satisfying (7), with \( \delta = \delta_{\text{new}} > 0 \). Let an initial approximation be such that \( e_0 \leq \delta_{\text{old}} \), where \( \delta_{\text{new}} < \delta_{\text{old}} \leq R \), and let \( S_1 \delta_{\text{old}} < 1 \). Then for

\[
l \geq \left\lfloor \frac{\log \log \frac{1}{S_1 \delta_{\text{new}}} - \log \log \frac{1}{S_1 \delta_{\text{old}}}}{\log p} \right\rfloor
\]

we have

\[
e_l \leq \delta_{\text{new}}.
\]
Proof. Form Lemma 1 with $d = d_{\text{new}}$ we have that $e_l \leq a_0 = d_{\text{new}}$ if $e_0 \leq a_l$. The latter inequality holds if $d_{\text{old}} \leq a_l$. Since $S_1 a_l = (S_1 a_0)^{1/p_l}$, the inequality $d_{\text{old}} \leq a_l$ is equivalent to the inequality

$$l \geq \left\lceil \log \frac{1}{S_1 d_{\text{new}}} - \log \frac{1}{S_1 d_{\text{old}}} \right\rceil \log p,$$

which completes the proof.

2.3. Linearly convergent methods

To show results for linearly convergent methods corresponding to Lemmas 1 and 2, let $a_0 = d_0$, and $a_{i+1} = \frac{2}{1 + q} a_i$, $i = 0, 1, \ldots$.

The sequence $\{a_i\}$ is increasing and $\lim_{i \to +\infty} a_i = +\infty$. We now have the following lemmas, whose proofs are similar to those of the previous two lemmas.

Lemma 3. If the initial approximation satisfies $e_0 \leq a_l$ then $e_i \leq a_{l-i}$ for $i = 0, 1, \ldots, l$.

Lemma 4. Consider an iterative method of the form (5) with errors satisfying (8), with $\delta = \delta_{\text{new}} > 0$. Let an initial approximation be such that $e_0 \leq \delta_{\text{old}}$, where $\delta_{\text{new}} < \delta_{\text{old}}$. Then we have:

$$\text{if } l \geq \left\lceil \log \frac{\delta_{\text{old}}}{\delta_{\text{new}}} \right\rceil \log \frac{2}{1 + q} \text{ then } e_l \leq \delta_{\text{new}}.$$

3. Selected precision sequences—upper bounds on the total cost

Based on Lemmas 2 and 4, we now complete the definition of the $i$th stage (9) by specifying the number of iterative steps $l_i$ that are performed. For superlinearly convergent methods, we take

$$l_i = \left\lceil \log \log \frac{1}{S_1 \delta_i} - \log \log \frac{1}{S_1 \delta_{i-1}} \right\rceil \log p,$$

(17)
for \( i = 0, 1, \ldots, k - 1 \), and

\[
l_i = \left[ \log \log \frac{1}{S_i} - \log \log \frac{1}{S_i \delta_{i-1}} \right] \frac{\log p}{\log \frac{1}{1 + q}}.
\]

For linearly convergent methods, we take

\[
l_i = \left[ \frac{\log \delta_{i-1}}{\delta_i} \right] \frac{\log \frac{1}{\delta_i}}{\log \frac{1}{1 + q}} \quad \text{for } i = 0, 1, \ldots, k - 1, \quad \text{and} \quad l_k = \left[ \frac{\log \delta_{k-1}}{\delta_k} \right] \frac{\log \frac{1}{\delta_k}}{\log \frac{1}{1 + q}}.
\]

Note that, for instance, if we choose \( k = 1 \), that is, if we pass from \( \delta_0 \) to \( \varepsilon \) in one stage, then

\[
TC(\varepsilon) = l_1 c(\varepsilon) = O_c(\varepsilon) \log \log \frac{1}{c(\varepsilon)}, \quad \text{as } \varepsilon \to 0
\]

for superlinearly convergent methods, and

\[
TC(\varepsilon) = l_1 c(\varepsilon) = O_c(\varepsilon) \log \log \frac{1}{c(\varepsilon)}, \quad \text{as } \varepsilon \to 0
\]

for linearly convergent methods. We shall show that in each case, under some assumptions on \( c(\delta) \), bounds (20) and (21) can be improved by selecting better precision sequence \( \{\delta_i\} \).

### 3.1. Superlinearly convergent methods

The total cost \( TC(\varepsilon) \) is now given by (10) with \( l_i \) defined by (17) and (18). Consider one of the following two possibilities of defining \( \{\delta_i\} \). Let \( M > 1 \). Our first possible definition is

\[
\delta_0 = R, \quad \delta_i = \delta_{i-1}^M, \quad i = 0, 1, \ldots
\]

assuming that \( R < 1 \). Our second possible definition (which requires a knowledge of \( S \)) is

\[
\delta_0 = R, \quad \delta_i = S_i^{M-1} \delta_{i-1}^M, \quad i = 0, 1, \ldots
\]

assuming that \( S_i R < 1 \).

We shall consider cost functions \( c(\delta) \) satisfying the following condition: for any positive \( L \) and \( M > 1 \) there exits \( K > 1 \) and \( \hat{\delta} \) such that for all \( 0 < \delta \leq \hat{\delta} \), we have

\[
c(L \delta^M) \geq K c(\delta).
\]

(Note that (24) always holds with \( K = 1 \).) It is easy to check that (24) holds for the cost functions \( c(\delta) = C(1/\delta)^m \) as well as for \( c(\delta) = C \log 1/\delta \), but it does not for \( c(\delta) = C \log \log 1/\delta \), since \( \lim_{\delta \to 0} c(L \delta^M) / c(\delta) = 1 \). As we shall see this is not a coincidence; the behavior of \( TC(\varepsilon) \) is different for the latter function.
Let $I$ be the maximal number $i$ such that $\delta_i > \hat{\delta}$. We write
\[ TC(\varepsilon) = \sum_{i=1}^{I} l_i c(\delta_i) + \sum_{i=I+1}^{k-1} l_i c(\delta_i) + l_k c(\varepsilon). \]
Note that the first sum above is bounded by a constant that depends only on the parameters of the problem, and is independent of $\varepsilon$. To find an upper bound on the remaining terms, we first note that for any of the two selections of $\delta_i$, the number of iterative steps $l_i$ is bounded by a constant. For choice (22), we have
\[ l_i \leq \left\lceil \frac{1 + \log M}{\log p} \right\rceil, \tag{25} \]
provided that $R$ is sufficiently small, while for (23), we have
\[ l_i \leq \left\lfloor \frac{\log M}{\log p} \right\rfloor. \tag{26} \]
(Note that in the latter case $l_i = 1$ if we take $M \leq p$.)

To find a bound on $c(\delta_i)$ we use condition (24). For $\delta_i \leq \delta$, we have
\[ c(\delta_i) \leq \frac{1}{K^{k-i-1}} c(\delta_{k-1}), \quad i = 1, 2, \ldots, k - 1. \]
Using this and $c(\delta_{k-1}) \leq c(\varepsilon)$, we get that
\[ \sum_{i=I+1}^{k-1} l_i c(\delta_i) + l_k c(\varepsilon) \leq \left\lceil \frac{1 + \log M}{\log p} \right\rceil \frac{2K - 1}{K - 1} c(\varepsilon). \]

We have shown the following:

**Theorem 1.** Let $F \in \mathcal{F}$ and let $\phi$ be a superlinearly convergent iterative method implemented as in (9) with the precisions $\delta_i$ given by (22) or (23). Let the cost function satisfy (24). Then there exists a constant $C_1 > 0$ such that the total cost satisfies
\[ TC(\varepsilon) \leq C_1 c(\varepsilon), \quad \text{as} \quad \varepsilon \to 0. \tag{27} \]

Recall that assumption (24) holds true for the cost functions considered in Examples 1 and 2. In addition to Theorem 1, we recall that for the cost function $c(\delta) = C \log \log 1/\delta$ (which does not satisfy (24)), a one-stage passage from $\delta_0$ to the final accuracy $\varepsilon$ leads to
\[ TC(\varepsilon) = O(c(\varepsilon)^2) \quad \text{as} \quad \varepsilon \to 0, \tag{28} \]
see (20).

Theorem 1 shows that for a class of cost functions, an implementation of the iterative method $\phi$ is possible such that the total cost of computing an $\varepsilon$-approximation to $x$ is proportional to the cost of one iterative step, performed with the accuracy $1/\varepsilon$. We now state a result corresponding to Theorem 1 for linearly convergent methods.
3.2. Linearly convergent methods

Recall that the total cost $TC(\varepsilon)$ is now given by (10) with $l_i$ given by (19). We define the precisions $\{\delta_i\}$ by

$$\delta_0 > 0 \text{ arbitrary, } \delta_i = \frac{1}{2}(1 + q)\delta_{i-1}, \quad i = 1, 2, \ldots$$  \hspace{1cm} (29)

Then we have $l_1 = l_2 = \ldots = l_k = 1$. Consider cost functions $c(\delta)$ satisfying the following condition:

for any $\tilde{q} \in (0, 1)$ there exists $K > 1$ and $\hat{\delta}$ such that for $0 < \delta \leq \hat{\delta}$, we have

$$c(\tilde{q}\delta) \geq Kc(\delta).$$ \hspace{1cm} (30)

(Note that (30) always holds with $K = 1$.) It is easy to see that (30) holds for the cost function $c(\delta) = C(1/\delta)^m$, but it does not for $c(\delta) = C \log 1/\delta$.

Assumption (30) with $\tilde{q} = \frac{1}{2}(1 + q)$ allows us to derive an upper bound on $TC(\varepsilon)$ much as we did for superlinearly convergent methods. We get

**Theorem 2.** Let $F \in \mathcal{F}$ and let $\phi$ be a linearly convergent method, implemented as in (9) with the precisions $\delta_i$ given by (29). Let the cost function $c(\delta)$ satisfy (30). Then the total cost of the method satisfies

$$TC(\varepsilon) \leq \frac{2(2K - 1)}{K - 1}c(\varepsilon) \text{ as } \varepsilon \rightarrow 0.$$ \hspace{1cm} (31)

In addition to Theorem 2, we recall that for the cost function $c(\delta) = C \log 1/\delta$ (which does not satisfy (30)), a one-stage passage to the final accuracy $\varepsilon$ leads to

$$TC(\varepsilon) = O(c(\varepsilon)^2) \text{ as } \varepsilon \rightarrow 0,$$ \hspace{1cm} (32)

see (21).

4. Arbitrary precision sequences—lower bounds on the total cost

Given $\varepsilon > 0$, and an initial approximation $x_0$ such that $\varepsilon < ||x_0 - x||$ (in the case of superlinearly convergent methods, $x_0$ is assumed sufficiently close to $x$), we assume that the accuracy $\varepsilon$ is achieved by a perturbed method in some number, say $k$, of stages

$$x_0 \rightarrow x_1 \rightarrow \ldots \rightarrow x_k.$$ 

That is, denoting the $i$th error $e_i = ||x_i - x||$ by $\delta_i$ (in correspondence to the notation of the previous section), we have that $\delta_k \leq \varepsilon$, and $k$ is the smallest number with this property. The general form of $\phi$ is given by (9) with

$$||v'_i|| \leq \frac{1}{2}w_\sigma_i,$$ \hspace{1cm} (33)
where \(\sigma_i\) are some parameters controlling the precision of computing \(x_{s+1}\) that must be related to the errors \(\delta_i\). The total cost of \(\phi\) is given by

\[
TC(\varepsilon) = \sum_{i=1}^{k} l_i c(\sigma_i).
\]

(34)

Our aim is to find lower bounds on \(TC(\varepsilon)\). Let us discuss assumptions that we shall need in order to do this. First of all, we have to be sure that perturbations \(\gamma_i\) affecting each iteration are not ‘structured’, in the sense that they do not accelerate the convergence of the method. If, for instance, \(\phi\) is a quadratically convergent method, and \(\{y_s\}\) is a cubically convergent sequence, then for the perturbations defined by \(\gamma_s = y_{s+1} - \phi(y_s, F)\), we get \(x_s = y_s\) for all \(s\), so that the perturbed sequence \(\{x_s\}\) converges faster than the exact sequence \(\{x_s\}\). To avoid such situations, we assume that if the perturbed sequence gives a \(\kappa\)-approximation, then the exact sequence does the same job. More precisely, we assume that for any \(F \in \mathcal{F}\), any starting point \(x_0 = \hat{x}_0 \in K(\alpha, R)\) any \(\kappa > 0\), and any \(l\), the implication

\[
\text{if } ||x_l - \alpha|| \leq \kappa \text{ then } ||\hat{x}_l - \alpha|| \leq \kappa
\]

(35)

holds. The next condition that we must adopt assures that the cost function does not decrease too fast as \(\delta\) grows. Namely, we assume that for any \(D > 1\) there exists \(K_1 > 0\) and \(\delta' > 0\) such that

\[
c(D\delta) \geq K_1 c(\delta),
\]

(36)

for all \(\delta \leq \delta'\). (Note that we must have \(K_1 \leq 1\).) Assumption (36) holds true for each cost function defined in Examples 1–3.

Conditions (35) and (36) allow us to establish the (very intuitive) fact that any iterative method achieving the accuracy \(\varepsilon\) must perform at least one step with the accuracy proportional to \(\varepsilon\). More specifically, consider the \(k\)th step of the method that produces an \(x_k\) such that \(||x_k - \alpha|| \leq \varepsilon\). If \(y\) is a starting point for this step (||\(y - \alpha\)|| > \(\varepsilon\)), i.e.,

\[
x_k = \phi(y, F) + \gamma, \quad ||\gamma|| \leq \frac{1}{w} \sigma,
\]

(37)

then by (35) we have that \(||\phi(y, F) - \alpha|| \leq \varepsilon\), which yields \(||\gamma|| \leq 2\varepsilon\). Hence, \(\sigma \leq 4\varepsilon/w\). From (36), this implies that \(c(\sigma) \geq c(4\varepsilon/w) \geq K_1 c(\varepsilon)\). This leads to the bound

\[
TC(\varepsilon) \geq K_1 c(\varepsilon) \quad \text{as} \quad \varepsilon \to 0.
\]

(38)

This result provides a matching lower bound for cost functions considered in Theorems 1 and 2 and satisfying (36).

We shall now analyze two cost functions for which the obtained upper bounds are quadratic in \(c(\varepsilon)\). The first is \(c(\delta) = C \log \log 1/\delta\) for superlinearly convergent methods, and the second is \(c(\delta) = C \log 1/\delta\) for linearly convergent methods. First, let \(\phi\) be a superlinearly convergent method. We assume that its order is bounded from above, i.e., the exact iteration \(\hat{x}_{s+1} = \phi(\hat{x}_s, F)\) satisfies the following condition: for some \(F \in \mathcal{F}\) and a positive constant \(\hat{S}\), for any starting point \(\hat{x}_0\) from some neighborhood of \(\alpha\), we have

\[
||\hat{x}_{s+1} - \alpha|| \geq \hat{S} ||\hat{x}_s - \alpha||^\rho, \quad s = 0, 1, \ldots.
\]

(39)
For instance, this condition holds with \( p = 2 \) for Newton’s method. We shall see that condition (39) can be relaxed by assuming that inequality (39) holds with \( p \) replaced by some \( \tilde{p} \geq p \). This will only influence a constant in the final inequality (42).

We may assume, without loss of generality, that \( \delta_0 > \delta_1 > \cdots > \delta_k \). Indeed, if this condition is not satisfied, then we restrict our attention to a decreasing subsequence \( \{\delta_i\} \) defined by

\[
i_{s+1} = \min \{ i > i_s : \delta_i < \delta_{i_s} \}, \quad s = 0, 1, \ldots, l - 1,
\]

with \( i_0 = 0 \) and \( i_l = k \), for some \( 1 \leq l \leq k \). It is possible to show that such a restriction leads to a lower bound on \( TC(\varepsilon) \).

We bound \( c(\sigma_i) \) in (34) from below. From assumption (35), since \( ||x_i - x|| \leq \delta_i \), we have that \( ||\phi(x_i) - x|| \leq \delta_i \). Hence, \( ||\gamma_i|| \leq ||x_i - x|| + ||\phi(x_i) - x|| \leq 2\delta_i \), so that \( \sigma_i \leq 4\delta_i \), and \( c(\sigma_i) \geq c(4\delta_i) \). To find a bound on \( l_i \), the number of steps of the perturbed method which allows us to pass from \( \delta_{i-1} \) to \( \delta_i \) we again use (35), which yields that \( ||\hat{x} - x|| \leq \delta_i \). Hence, the number \( l_i \) is at least as large as the number of steps of the exact method. By (39), we have

\[
l_i \geq \frac{\log \log \frac{1}{S_1 \delta_i} - \log \log \frac{1}{S_1 \delta_{i-1}}}{\log \tilde{p}}
\]

for some \( F \), where \( \tilde{S}_1 = \tilde{S}_1^{1/(p-1)} \). Summarizing, we have found that

\[
TC(\varepsilon) \geq \frac{1}{\log \tilde{p}} \sum_{i=1}^{k} \left( \log \log \frac{1}{S_1 \delta_i} - \log \log \frac{1}{S_1 \delta_{i-1}} \right) c(4\delta_i).
\]

Since the cost function \( c(\delta) = C \log \log 1/\delta \) satisfies (36), we have that

\[
TC(\varepsilon) \geq K_1 \frac{C}{\log \tilde{p}} \sum_{i=1}^{k} (a_i - a_{i-1}) b_i,
\]

where \( a_i = \log \log 1/(\tilde{S}_1 \delta_i) \) and \( b_i = \log \log 1/\delta_i \). It is easy to see that \( b_i/a_i \geq \frac{1}{2} \) for sufficiently small \( \delta_i \), so that

\[
\sum_{i=1}^{k} (a_i - a_{i-1}) b_i = \sum_{i=1}^{k} \frac{a_i - a_{i-1}^2}{a_i} b_i \
\begin{aligned}
&= \frac{1}{2} \sum_{i=1}^{k} \frac{a_i^2 - a_{i-1}^2}{a_i} \\
&\geq \frac{1}{4} \sum_{i=1}^{k} (a_i^2 - a_{i-1}^2) = \frac{1}{4} (a_k^2 - a_0^2),
\end{aligned}
\]

which yields

\[
TC(\varepsilon) \geq \frac{K_1 C}{4 \log \tilde{p}} \left( \log \log \frac{1}{\tilde{S}_1 \varepsilon} \right)^2 - \left( \log \log \frac{1}{\tilde{S}_1 \delta_0} \right)^2.
\]

For linearly convergent methods and the cost function \( c(\delta) = C \log 1/\delta \), we proceed similarly. The only difference is that assumption (39) is now replaced by the requirement that for some \( F \in \mathcal{F} \) and a constant \( \hat{q} \in (0, 1) \), for any starting point \( \hat{x}_0 \)
from some neighborhood of $x$, we have
\[ ||\hat{x}_{s+1} - x|| \geq \hat{q}||\hat{x}_s - x||, \quad s = 0, 1, \ldots. \] 
(43)

We get in this case that $TC(\epsilon) = \Omega((\log 1/\epsilon)^2)$. The following theorem summarizes the discussion above.

**Theorem 3.** (i) Let $\phi$ be any iterative method (with superlinear or linear convergence) satisfying (35), and let the cost function $c(\delta)$ fulfill condition (36). Then there exists $C_2 > 0$ such that for any $F \in \mathcal{F}$ and for an arbitrary precision sequence $\{\delta_i\}$ in (9), we have
\[ TC(\epsilon) \geq C_2 c(\epsilon) \quad \text{as} \quad \epsilon \to 0. \] 
(44)

(ii) If, in addition to (35), a superlinearly convergent method satisfies (39) for a function $F$, and the cost function is given by $c(\delta) = C\log \log 1/\delta$, or a linearly convergent method satisfies (43) and $c(\delta) = C\log \log 1/\delta$, then for this $F$ there exists $C_3 > 0$ such that
\[ TC(\epsilon) \geq C_3 c(\epsilon)^2 \quad \text{as} \quad \epsilon \to 0. \] 
(45)

Let us remark that from the proof of Theorem 3 it follows that for superlinearly convergent methods and any cost function satisfying $c(\delta) \geq C\log \log 1/\delta$, we have $TC(\epsilon) = O(c(\epsilon)\log \log 1/\epsilon c(\epsilon))$ for the one-stage passage from $\delta_0$ to $\epsilon$ as in (20), and $TC(\epsilon) = \Omega((\log^2 1/\epsilon^2))$ for an arbitrary precision sequence $\{\delta_i\}$.

Similarly, for linearly convergent methods and any cost function satisfying $c(\delta) \geq C\log 1/\delta$, we have $TC(\epsilon) = O(c(\epsilon)\log \log 1/\epsilon c(\epsilon))$ if we jump from $\delta_0$ to $\epsilon$ in one stage as in (21), and $TC(\epsilon) = \Omega((\log 1/\epsilon)^2)$ for arbitrary $\{\delta_i\}$.

5. Application

We show in this section that an application of the results above allows us to improve complexity bounds for nonlinear two-point boundary-value problems, recently established in [1]. Let us briefly recall the formulation of the problem and the main result. The details are omitted; for a full discussion, the reader is referred to [1]. We have considered in [1] a general boundary-value problem
\[ z'(t) = f(t, z(t)), \quad t \in [a, b], \quad p(z(a), z(b)) = 0, \] 
(46)
as well as a scalar second-order problem with separated boundary conditions
\[ z''(t) = f(t, z(t)), \quad t \in [a, b], \quad z(a) = A, \quad z(b) = B, \] 
(47)
where $f: [a, b] \times \mathbb{R}^d \to \mathbb{R}^d$ (with $d = 1$ in (47)) is a function with $r \geq 2$ continuous bounded partial derivatives, belonging to a certain class $\mathcal{F}$, and $p: \mathbb{R}^d \times \mathbb{R}^d \to \mathbb{R}^d$ has continuous bounded partial derivatives of first and second order. Our aim is to approximate the solution $z$ using some information about $f$, with the final goal of finding the $\epsilon$-complexity of problems (46) and (47). We assume that an approximation to $z$ can be computed on the basis of linear information about $f$.
given by \( n \) evaluations of linear functionals at \( f \) (possibly adaptively chosen), i.e.,
\[
N(f) = [L_1(f), L_2(f), \ldots, L_n(f)].
\] (48)
If \( L_\varepsilon(f) \) are values of \( f \) or its partial derivatives, then information is called standard. Given \( \varepsilon > 0 \), the \( \varepsilon \)-complexity of the problem is defined as the minimal number of functionals in (48) that allows the computation of an \( \varepsilon \)-approximation to \( z \) (in the supremum norm), i.e.,
\[
\text{comp}(\varepsilon) = \min \{ n : \text{an} \ \varepsilon \text{-approximation can be computed using (48)} \}. \] (49)
In [1], upper and lower bounds on the \( \varepsilon \)-complexity have been established, of order \((1/\varepsilon)^{1/(r+n)} \log \log 1/\varepsilon\) and \((1/\varepsilon)^{1/(r+n)}\), respectively. Here, \( \eta = 0 \) for both (46) and (47) if standard information is only allowed, \( \eta = 1 \) if arbitrary linear information can be used for (46), and \( \eta = 2 \) if arbitrary linear information can be used for (47). Since the logarithmic factor is essentially negligible compared to \((1/\varepsilon)^{1/(r+n)}\), such a result is often considered satisfactory. However, from a theoretical point of view, the asymptotic behavior of the \( \varepsilon \)-complexity as \( \varepsilon \to 0 \) is still unknown until there is a gap between the upper and lower bounds. We now use the ideas described in the previous sections to get rid of the logarithmic factor in the upper bound, thus removing the gap. The complexity of the problem turns out to be of order \((1/\varepsilon)^{1/(r+n)}\). The way in which we get rid of the logarithmic factor is similar for both problems (46) and (47), so that we shall concentrate on problem (46). As usual in shooting methods, the solution of (46) is expressed as \( z(t) = y(t, x) \), where \( y(t, x) \) is the solution of the initial value problem
\[
y'(t) = f(t, y(t)), \quad t \in [a, b], \quad y(a) = x,
\] (50)
and \( x \) is the solution of the equation \( F(x) = p(x, y(b, x)) = 0 \). We assume that this equation has a unique solution \( x \) and that \( F(x) \) is a nonsingular matrix.

The problem of approximating \( z \) is in main part reduced to approximating the solution \( x \), which can be done by means of Newton’s method \( \tilde{x}_{s+1} = \tilde{x}_s - F'(\tilde{x}_s)^{-1}F(\tilde{x}_s), \ s = 0, 1, \ldots \). To compute the value of \( F(\tilde{x}_s) \), we have to solve the initial value problem (50) with \( x = \tilde{x}_s \) for \( y(b, \tilde{x}_s) \). Similarly, to compute \( F'(\tilde{x}_s) \) we have to solve for \( Y(t, x) = \partial y(t, x)/\partial x \) the initial value problem of the form
\[
Y'(t, x) = \frac{\partial f}{\partial y}(t, y(t, x)) Y(t, x), \quad t \in [a, b] \quad Y(a, x) = I,
\] (51)
where \( x = \tilde{x}_s \), and \( I \) is the identity matrix. We face here a typical situation described in the previous sections: the exact values of \( F(\tilde{x}_s) \) and \( F'(\tilde{x}_s) \) are not available; we can only compute approximations whose errors depend on the amount of computational work that is involved when solving the initial value problems (50) and (51). Hence, instead of exact Newton’s method, we are forced to use its perturbed version
\[
x_{s+1} = x_s - A(x_s)^{-1}\tilde{F}(x_s), \ s = 0, 1, \ldots,
\] (52)
where \( \tilde{F}(x) \) and \( A(x) \) are approximations to \( F(x) \) and \( F'(x) \), respectively. By dividing the interval \([a, b]\) into \( m \) subintervals of length \((b-a)/m\), it is possible to compute
approximations such that \( \| F(x) - \hat{F}(x) \| = O(m^{\nu + \eta}) \) and \( \| F'(x) - A(x) \| = O(m^{\nu + \eta - 1}) \). (The constant in the \( O \) notation does not depend on \( x \)). Computing \( \hat{F}(x) \) and \( A(x) \) requires \( cm \) information evaluations, where \( c \) is an integer depending on \( r \) and \( d \). Furthermore, for \( x_0 \in K(\alpha,R) \) and sufficiently large \( m \), the sequence \( \{x_s\} \) is well defined, it remains in \( K(\alpha,R) \), and
\[
e_{s+1} \leq \frac{1}{2} S e_s^2 + M_1 m^{-(r+\eta-1)} e_s + M_2 m^{-(r+\eta)}, \quad s = 0, 1, \ldots, \quad (53)
\]
where \( S, M_1 \) and \( M_2 \) are some positive constants, see [1]. Hence, \( e_{s+1} \leq \frac{1}{2} S e_s^2 + \frac{1}{2} r_m \), with \( r_m = \Theta(m^{-(r+\eta-1)}) \). To establish an upper bound on the \( \varepsilon \)-complexity, we use Lemma 3.3 of [1], where it is shown that for a given (sufficiently large) \( m \), it suffices to compute \( \Theta(m \log \log m) \) functionals to get Newton’s approximation to \( \alpha \) with error \( O(m^{-(r+\eta)}) \). A crucial point that allows us to improve the upper bound on the \( \varepsilon \)-complexity by removing the logarithmic factor \( \log \log 1/\varepsilon \) is that the error \( O(m^{-(r+\eta)}) \) can be achieved by computing \( \Theta(m) \) functional evaluations only. To see this, let us properly define the cost function \( c(\delta) \). Since we are interested in the number of functionals involved in computations, we set \( c(\delta) = \min \{ cm : r_m \leq \delta \} \). That is, we deal with the cost function of the form \( c(\delta) = C(1/\delta)^{1/(r+\eta-1)} \) for some positive constant \( C \), see Example 1.

Let now \( e_1 = m^{-(r+\eta+1)} \). We use the strategy described in Section 3 to get an approximation \( x_k \) such that \( e_k \leq e_1 \) with total cost \( \text{TC}(e_1) = O(c(e_1)) = O(m) \). Performing one more step, and using (53), we have
\[
e_{k+1} \leq \frac{1}{2} S e_1^2 + M_1 m^{-(r+\eta-1)} e_1 + M_2 m^{-(r+\eta)}
\]
\[
= O\left( m^{-2(r+\eta-1)} + m^{-(r+\eta)} \right) = O\left( m^{-(r+\eta)} \right).
\]

The computation of \( x_{k+1} \), followed by an approximation of the solution \( z \) with error \( O(m^{-(r+\eta)}) \), requires all together \( n = \Theta(m) \) functionals.

For problem (47) we use the same ideas and proceed similarly. These considerations lead to the following improved result about the complexity \( \text{comp}(\varepsilon) \) for problems (46) and (47), which holds under the same assumptions as in [1]. The upper bound below holds true for \( r \geq 2 \), and for all functions \( p \) in (46), while the lower bound holds for \( r \geq 1 \), and for some function \( p \).

**Theorem 4.** Under the assumptions of Theorems 3.2 and 4.4 of [1] we have
\[
\text{comp}(\varepsilon) = \Theta\left( (1/\varepsilon)^{1/(r+\eta)} \right) \quad \text{as} \quad \varepsilon \to 0, \quad (54)
\]

where \( \eta = 0 \) for both (46) and (47) if standard information is only allowed, \( \eta = 1 \) for (46) and \( \eta = 2 \) for (47) if arbitrary linear information can be used.

**References**