Almost optimal solution of initial-value problems by randomized and quantum algorithms[☆]

Bolesław Kacewicz[*]

Department of Applied Mathematics, AGH University of Science and Technology, Al. Mickiewicza 30, paw. A3/A4, III p., pok. 301, 30-059 Cracow, Poland

Abstract

We establish essentially optimal bounds on the complexity of initial-value problems in the randomized and quantum settings. For this purpose we define a sequence of new algorithms whose error/cost properties improve from step to step. These algorithms yield new upper complexity bounds, which differ from known lower bounds by only an arbitrarily small positive parameter in the exponent, and a logarithmic factor. In both the randomized and quantum settings, initial-value problems turn out to be essentially as difficult as scalar integration.
© 2006 Elsevier Inc. All rights reserved.

Keywords: Initial-value problems; Randomized algorithms; Quantum computation; Optimality; Complexity

1. Introduction

The complexity of initial-value problems has been well understood for many years in the deterministic settings, see [4] for a survey of the worst-case and asymptotic results. Much less is known about the complexity of these problems if randomized methods or quantum computation are allowed. Other problems related to initial-value problems, such as summation or integration, have already been well studied in both randomized and quantum settings. A survey of complexity results for summation and integration can be found in [3].

Randomized and quantum complexity of initial-value problems was first studied in [5]. This paper showed that a nontrivial speed-up can be achieved by switching from the worst-case deterministic setting to the randomized or quantum computation. Improved complexity bounds, the

best known up to now, were next shown in [6]. In the Hölder class of right-hand side functions with r continuous bounded partial derivatives, with the rth derivative being a Hölder function with exponent ρ, the ε-complexity was shown to be $O\left((1/\varepsilon)^{1/(r+\rho+1/3)}\right)$ in the randomized setting, and $O\left((1/\varepsilon)^{1/(r+\rho+1/2)}\right)$ on a quantum computer (logarithmic factors are here neglected).

Unfortunately, these upper bounds do not match known lower bounds on the ε-complexity. It can be shown [5] that the ε-complexity is $\Omega\left((1/\varepsilon)^{1/(r+\rho+1/2)}\right)$ in the randomized case, and $\Omega\left((1/\varepsilon)^{1/(r+\rho+1)}\right)$ on a quantum computer. Hence, aside from logarithmic factors, there is a significant difference between the bounds in the exponent. The gap was essentially filled for a restricted class of scalar autonomous problems in [6]. For systems of equations, the discrepancy between upper and lower bounds remained an open problem. These results are briefly recalled in Section 2.

In this paper, we show almost optimal upper complexity bounds in the randomized and quantum settings for a general initial-value problem. The main results are contained in Theorems 1 and 2. We recursively define a sequence $A_1, A_2, \ldots, A_k$ of randomized or quantum algorithms (which are new for $k \geqslant 3$) for systems of initial-value problems. The performance of A_k improves as k grows, which is a crucial component of the complexity analysis. The error and cost bounds for A_k are shown in Theorem 1.

Properties of A_k, with k properly chosen, lead to new upper complexity bounds given in Theorem 2. These differ from the lower bounds by only an arbitrarily small positive parameter γ in the exponent, and a logarithmic factor. The ε-complexity turns out to be $O\left((1/\varepsilon)^{1/(r+\rho+1/2-\gamma)}\right)$ in the randomized setting, and $O\left((1/\varepsilon)^{1/(r+\rho+1-\gamma)}\right)$ on a quantum computer, up to a logarithmic factor. Hence, in both the randomized and quantum settings, the complexity of initial-value problems is essentially established. Roughly speaking, the problem is as difficult as scalar integration [3].

2. Problem definition and known complexity bounds

In this section we define the problem to be solved, and briefly introduce basic notions on randomized and quantum settings used in the rest of the paper. We also recall, for later comparison, known results on the complexity of initial-value problems in both settings.

We consider the solution of a system of ordinary differential equations with initial conditions

$$z'(t) = f(z(t)), \quad t \in [a, b], \quad z(a) = \eta, \tag{1}$$

where $f : \mathbf{R}^d \to \mathbf{R}^d$, $z : [a, b] \to \mathbf{R}^d$ and $\eta \in \mathbf{R}^d$ $(f(\eta) \neq 0)$. Nonautonomous systems $z'(t) = f(t, z(t))$ with $f : \mathbf{R}^{d+1} \to \mathbf{R}^d$ can also be written in the form (1), by adding one scalar equation

$$\begin{bmatrix} u'(t) \\ z'(t) \end{bmatrix} = \begin{bmatrix} 1 \\ f(u(t), z(t)) \end{bmatrix},$$

with an additional initial condition $u(a) = a$.

We shall study the complexity of problem (1) in the randomized and quantum settings. We assume that the right-hand side function $f = [f^1, \ldots, f^d]^{\mathrm{T}}$ belongs to the Hölder class $F^{r,\rho}$ given as follows. For an integer $r \geqslant 0$, $\rho \in (0, 1]$, and positive numbers $D_0, D_1, \ldots, D_r, H$, we define

$$F^{r,\rho} = \{ f : \mathbf{R}^d \to \mathbf{R}^d \mid f \in C^r(\mathbf{R}^d), \quad |\partial^i f^j(y)| \leqslant D_i, \ |i| \leqslant r,$$

$$|\partial^r f^j(y) - \partial^r f^j(z)| \leqslant H \, \|y - z\|^\rho, \ y, z \in \mathbf{R}^d, \quad j = 1, 2, \ldots, d \}, \tag{2}$$

where $\partial^i f^j$ represents any partial derivative of order $|i|$ of the jth component of f. Here, and in the rest of this paper, $\|\cdot\|$ denotes the maximum norm in $\mathbf{R}^d$.

Let $q = r + \rho$ denote the regularity parameter for the class $F^{r,\rho}$. We assume that $q \geqslant 1$, which assures that f is always a Lipschitz function.

We now briefly recall basic notions related to the randomized and quantum settings. For a complete discussion, in particular of the quantum setting, the reader is referred to [2].

We wish to compute a bounded function l on $[a, b]$ approximating the solution z. Information available about the right-hand side f in the construction of l is provided by a subroutine that computes values of a component of f or one of its partial derivatives. In the randomized setting, the points at which the values are computed can be selected randomly. On a quantum computer, by subroutine calls we mean either applications of a quantum query operator for (a component of) f, or evaluations of components of f or its partial derivatives on a classical computer. A detailed discussion of a quantum query operation can be found in [2]. A mapping A that computes l based on the available information is called an *algorithm*. In both randomized and quantum settings, the approximating function produced by an algorithm is random.

We now define the error of an algorithm in the two settings. Let $(\Omega, \Sigma, \mathbf{P})$ be a probability space. Let $\{x_i\}$ define the uniform partition of $[a, b]$, so that $x_i = a + ih$, $i = 0, 1, \ldots, n$ with $h = (b - a)/n$. The approximation $l = l(t)$ to $z = z(t)$ in $[a, b]$ is constructed in the algorithm A based on random approximations $y_i = y_i^\omega$ to $z(x_i)$, $i = 0, 1, \ldots, n$, $\omega \in \Omega$. (We assume that the mappings $\omega \mapsto y_i^\omega$ are random variables for each $f \in F^{r,\rho}$.) More specifically, $l = l^\omega$ is given by

$$l^\omega(t) = \psi(y_0^\omega, y_1^\omega, \ldots, y_n^\omega)(t), \tag{3}$$

for $t \in [a, b]$, where ψ is a certain mapping defining an approximation over $[a, b]$ from discrete approximations y_i^ω.

The error of A at f is defined by

$$e^\omega(A, f) = \sup_{t \in [a,b]} \|z(t) - l^\omega(t)\|. \tag{4}$$

We assume that the mapping $\omega \mapsto e^\omega(A, f)$ is a random variable for each $f \in F^{r,\rho}$.

In the *randomized setting*, the *error* of A in the class $F^{r,\rho}$ is given by the maximal dispersion of $e^\omega(A, f)$, namely,

$$e^{\mathrm{rand}}(A, F^{r,\rho}) = \sup_{f \in F^{r,\rho}} \left(\mathbf{E}e^\omega(A, f)^2\right)^{1/2}, \tag{5}$$

where $\mathbf{E}$ is the expectation. (The maximal expected value of $e^\omega(A, f)$ can be considered as well. This only changes constants in the results.)

In the *quantum setting*, the *error* of A in the class $F^{r,\rho}$ is given probabilistically by

$$e^{\mathrm{quant}}(A, F^{r,\rho}) = e^{\mathrm{quant}}(A, F^{r,\rho}, \delta) = \sup_{f \in F^{r,\rho}} \inf \{\, \alpha |\ \mathbf{P}\{e^\omega(A, f) > \alpha\} \leqslant \delta \,\}, \tag{6}$$

for a given number δ, $0 < \delta < \frac{1}{2}$.

The *cost* of an algorithm A, cost(A), in the randomized or quantum settings is measured by the maximal number (with respect to $f \in F^{r,\rho}$) of subroutine calls (with a proper meaning of a subroutine call in the respective setting) needed to compute an approximation. For a given $\varepsilon > 0$, the ε-*complexity* of the problem in the respective setting, $\mathrm{comp}^{\mathrm{rand(quant)}}(F^{r,\rho}, \varepsilon)$, is defined to be the minimal cost of an algorithm A taken among all A such that $e^{\mathrm{rand(quant)}}(A, F^{r,\rho}) \leqslant \varepsilon$.

In the quantum setting, to display the dependence on δ, we shall denote the complexity by $\text{comp}^{\text{quant}}(F^{r,\rho}, \varepsilon, \delta)$.

We now briefly recall known results on the subject. For scalar autonomous problems (1) (with $d = 1$), matching complexity bounds, up to logarithmic factors, were established in [6]. It was shown, in a slightly restricted class $\hat{F}^{r,\rho}$, and for a problem restricted to computing $z(b)$ only, that

$$\text{comp}^{\text{rand}}(\hat{F}^{r,\rho}, \varepsilon) = O\left(\left(\frac{1}{\varepsilon}\right)^{1/(q+1/2)} \left(\log \frac{1}{\varepsilon}\right)^2\right) \tag{7}$$

and

$$\text{comp}^{\text{rand}}(\hat{F}^{r,\rho}, \varepsilon) = \Omega\left(\left(\frac{1}{\varepsilon}\right)^{1/(q+1/2)}\right). \tag{8}$$

In the quantum setting

$$\text{comp}^{\text{quant}}(\hat{F}^{r,\rho}, \varepsilon, \delta) = O\left(\left(\frac{1}{\varepsilon}\right)^{1/(q+1)} \left(\log \log \frac{1}{\varepsilon} + \log \frac{1}{\delta}\right) \log \frac{1}{\varepsilon}\right) \tag{9}$$

and, for $0 < \delta \leqslant \frac{1}{4}$,

$$\text{comp}^{\text{quant}}(\hat{F}^{r,\rho}, \varepsilon, \delta) = \Omega\left(\left(\frac{1}{\varepsilon}\right)^{1/(q+1)}\right). \tag{10}$$

The proof of these results was based on switching to an equivalent scalar nonlinear equation. Such a method cannot be used for systems of equations. The best known complexity bounds for a general problem are given in [6]. It has been shown that

$$\text{comp}^{\text{rand}}(F^{r,\rho}, \varepsilon) = O\left(\left(\frac{1}{\varepsilon}\right)^{1/(q+1/3)} \log \frac{1}{\varepsilon}\right) \tag{11}$$

and

$$\text{comp}^{\text{quant}}(F^{r,\rho}, \varepsilon, \delta) = O\left(\left(\frac{1}{\varepsilon}\right)^{1/(q+1/2)} \left(\log \frac{1}{\varepsilon} + \log \frac{1}{\delta}\right)\right). \tag{12}$$

Similarly to the case of scalar equations, lower bounds are given by (8) and (10). Hence, there is a significant gap in known estimates of the complexity for systems of equations.

In the next part of this paper we show that the existing gap can essentially be filled up.

3. Main results

In this section we present main results of this paper. The first result, Theorem 1, assures the existence of a class of algorithms for solving problem (1) in the randomized (RAND) and quantum (QUANT) settings, which possess certain error and cost properties. These properties will allow us to derive the desired complexity bounds. The algorithms are defined constructively in the next section. The next result, Theorem 2, is an immediate consequence of Theorem 1. It contains new, and almost optimal, complexity bounds for problem (1) in the randomized and quantum settings.

In what follows, L denotes the Lipschitz constant for f.

Theorem 1. *For any $k \in \mathbf{N}$ and $s = 1, 2, \ldots, k$, there exists an algorithm A_s for solving (1) with the right-hand side $f \in F^{r,\rho}$ that computes for each $n \in \mathbf{N}$ and $\delta \in (0, \frac{1}{2})$ an approximation l^s to the solution z, and has the following properties.*

(i) There exist constants $\bar{C}_1^s$ and C_1^s, depending only on the parameters of the class $F^{r,\rho}$, a and b (and independent of η), such that

$$
\sup_{t \in [a,b]} \|z(t) - l^s(t)\| \leqslant
\begin{cases}
\bar{C}_1^s n^{-\alpha_s} & \text{if } L(b-a) \text{ is arbitrary} \\
& \quad \text{and } n \geqslant L(b-a)/\ln 2, \\
C_1^s (b-a)^{q+1} n^{-\alpha_s} & \text{if } L(b-a) \leqslant \ln 2 \text{ and } n \in \mathbf{N},
\end{cases}
\tag{13}
$$

where

$$
\alpha_s =
\begin{cases}
q(2^s - 1) + 2^{s-1} - 1 & \text{in RAND}, \\
qs + s - 1 & \text{in QUANT},
\end{cases}
\tag{14}
$$

and $\sup\{C_1^s : b - a \leqslant M\} < \infty$ for any $M > 0$.
This holds (for $n \geqslant 5$) with probability at least $1 - \delta$.

(ii) There exists a constant C_2^s, depending only on the parameters of the class $F^{r,\rho}$, a and b (and independent of η), such that for $n \in \mathbf{N}$ the cost of A_s is bounded by

$$
\text{cost}(A_s) \leqslant C_2^s n^{\beta_s} \left(\beta_k \log n + \log \frac{1}{\delta} \right),
\tag{15}
$$

where

$$
\beta_s =
\begin{cases}
2^s - 1 & \text{in RAND}, \\
s & \text{in QUANT}.
\end{cases}
\tag{16}
$$

and $\sup\{C_2^s : b - a \leqslant M\} < \infty$ for any $M > 0$.

Proof. See Section 5. $\square$

Let us comment on the conditions in Theorem 1 satisfied by the constants C_1^s and C_2^s. The algorithm A_s will be defined by recursive application of $A_{s-1}, \ldots, A_1$ on intervals of length going to 0. The properties of C_1^s and C_2^s assure that constants appearing in the error and cost estimates do not grow to infinity as the length of the interval tends to 0.

The error and cost estimates for algorithm A_k in the randomized and quantum versions, with a suitably chosen parameter k, play the main role in establishing the complexity of problem (1). The crucial point is the convergence from above, as $k \to \infty$, of the sequence $\{\beta_k/\alpha_k\}$ to the optimal exponent,

$$
\lim_{k \to \infty} \frac{\beta_k}{\alpha_k} =
\begin{cases}
\dfrac{1}{q + (1/2)} & \text{in RAND}, \\[2mm]
\dfrac{1}{q + 1} & \text{in QUANT}.
\end{cases}
$$

Note that the results for $k = 2$ agree with those from [6], since β_2/α_2 equals to $1/(q + \frac{1}{3})$ in RAND, and $1/(q + \frac{1}{2})$ in QUANT.

The following theorem gives essentially optimal upper complexity bounds for problem (1).

Theorem 2. *For any $\gamma \in (0, 1)$, there exist positive constants $C_1(\gamma)$, $C_2(\gamma)$ and $\varepsilon_0(\gamma)$ (depending only on γ, the parameters of the class $F^{r,\rho}$, a and b) such that for all $\varepsilon \in (0, \varepsilon_0(\gamma))$ and $\delta \in (0, \frac{1}{2})$ the ε-complexity in the randomized and quantum settings satisfies*

$$\text{comp}^{\text{rand}}(F^{r,\rho}, \varepsilon) \leqslant C_1(\gamma) \left(\frac{1}{\varepsilon}\right)^{1/(q+1/2-\gamma)} \tag{17}$$

and

$$\text{comp}^{\text{quant}}(F^{r,\rho}, \varepsilon, \delta) \leqslant C_2(\gamma) \left(\frac{1}{\varepsilon}\right)^{1/(q+1-\gamma)} \log \frac{1}{\delta}. \tag{18}$$

Proof. Define

$$k = \begin{cases} \lceil \log(1/\gamma + 1) \rceil & \text{in RAND,} \\ \lceil 2/\gamma \rceil & \text{in QUANT.} \end{cases} \tag{19}$$

Consider the randomized setting. We pass from the probabilistic error to the error given by (5) in a usual way, by selecting a suitable δ. Let K denote a positive (deterministic) upper bound on $e^{\omega}(A_k, f)$, depending only on the parameters of the class $F^{r,\rho}$, a and b. (One can see from the proof of Theorem 1 that such a bound exists.) Then the randomized error (5) of A_k is bounded by

$$e^{\text{rand}}(A_k, F^{r,\rho})^2 \leqslant K^2 \delta + (\bar{C}_1^k n^{-\alpha_k})^2$$

(for n sufficiently large). Selecting $\delta = 3\varepsilon^2/(4K^2)$ and $n = \lceil (2\bar{C}_1^k/\varepsilon)^{1/\alpha_k} \rceil$, we get that $e^{\text{rand}}(A_k, F^{r,\rho}) \leqslant \varepsilon$. The cost of A_k is then bounded by

$$\text{cost}(A_k) = O\left(\left(\frac{1}{\varepsilon}\right)^{\beta_k/\alpha_k} \log \frac{1}{\varepsilon}\right),$$

with the constant in the O-notation depending only on k and the parameters of the class $F^{r,\rho}$, a and b. By the definition of k, we have that

$$\frac{\alpha_k}{\beta_k} \geqslant q + \frac{1}{2} - \frac{\gamma}{2}.$$

Hence,

$$\text{cost}(A_k) \leqslant \hat{C}_1(\gamma) \left(\frac{1}{\varepsilon}\right)^{1/(q+1/2-\gamma/2)} \log \frac{1}{\varepsilon} \leqslant C_1(\gamma) \left(\frac{1}{\varepsilon}\right)^{1/(q+1/2-\gamma)},$$

for suitable constants $\hat{C}_1(\gamma)$, $C_1(\gamma)$, and ε sufficiently small. This proves (17).

We handle the quantum setting similarly. We estimate the error $e^{\text{quant}}(A_k, F^{r,\rho}, \delta)$ using (13) and (14), select n to assure that $e^{\text{quant}}(A_k, F^{r,\rho}, \delta) \leqslant \varepsilon$, and use the cost bound (15). The choice (19) of k leads to (18). $\quad\square$

Comparing upper bounds from Theorem 2 with lower bounds given in (8) and (10), we see that they match up to a small parameter γ in the exponent. In the quantum setting this holds up to a logarithmic factor that depends on δ.

4. Algorithms in the randomized and quantum settings

We define algorithms whose properties are the subject of Theorem 1. The definition is recursive. In both the randomized and quantum settings, we inductively define a sequence of algorithms $A_1, A_2, \ldots, A_s, \ldots, A_k$ for solving problem (1). The algorithm A_k, with a properly chosen index k, will be our final algorithm.

The points $\{x_i\}$ define the uniform partition of $[a, b]$, $x_i = a + ih$, $i = 0, 1, \ldots, n$ with $h = (b - a)/n$. We shall call n a *basic parameter*. To display the dependence on the interval over which the algorithm is applied and on the basic parameter used, we shall use the notation $A_s = A_s([a, b], n)$. An approximation to the solution z computed in $A_s([a, b], n)$ is denoted by $l^s = l^s(t)$, $t \in [a, b]$.

We start the induction with $s = 1$. The algorithm $A_1([a, b], n)$ is defined to be Taylor's algorithm in $[a, b]$ with step size h. We set $y_0 = \eta$. For a given y_i (an approximation to $z(x_i)$), we let z_i be the solution of the local problem

$$z_i'(t) = f(z_i(t)), \quad t \in [x_i, x_{i+1}], \quad z_i(x_i) = y_i, \tag{20}$$

and we define $y_{i+1} = l_i^0(x_{i+1})$. Here, $l_i^0(t)$ is the truncated Taylor's expansion of z_i for $t \in [x_i, x_{i+1}]$, given by

$$l_i^0(t) = \sum_{j=0}^{r+1} (1/j!) z_i^{(j)}(x_i)(t - x_i)^j.$$

The approximation to z in $A_1([a, b], n)$ is defined by

$$l^1(t) = l_i^0(t) \quad \text{for } t \in [x_i, x_{i+1}]. \tag{21}$$

Suppose that the algorithm $A_s([a, b], n)$ is defined for any $[a, b]$ and n. We now describe how to get $A_{s+1}([a, b], n)$ from $A_s([a, b], n)$.

We first inductively define a sequence of approximations $\{y_i\}$ in $A_{s+1}([a, b], n)$, where $y_i \approx z(x_i)$, $i = 0, 1, \ldots, n$. We set $y_0 = \eta$. For a given y_i, we apply the algorithm $A_s([x_i, x_{i+1}], m)$ to the problem (20), with a certain basic parameter m to be chosen later on. The approximating function obtained in this way is denoted by $l_i^s = l_i^s(t)$, $t \in [x_i, x_{i+1}]$.

The crucial point, which we now describe, is how we obtain y_{i+1} from y_i and l_i^s. We divide the interval $[x_i, x_{i+1}]$ into ml subintervals of equal length, with end points

$$z_j^i = x_i + j\bar{h},$$

with $j = 0, 1, \ldots, ml$, where $\bar{h} = (x_{i+1} - x_i)/(ml)$. Here l is another parameter that will be chosen in the sequel. (The notation l used for the parameter will not be confused with $l^s = l^s(t)$ denoting the approximating function, since the latter always appears with a superscript. The distinction is clear from the context.)

The solution of (20) satisfies the identity

$$z_i(x_{i+1}) = z_i(x_i) + \sum_{j=0}^{ml-1} \int_{z_j^i}^{z_{j+1}^i} f(z_i(t)) \, dt.$$

For $j = 0, 1, \ldots, ml - 1$ and $i = 0, 1, \ldots, n - 1$, define the polynomial

$$w_{ij}(y) = \sum_{k=0}^{r} \frac{1}{k!} f^{(k)}(l_i^s(z_j^i))(y - l_i^s(z_j^i))^k. \tag{22}$$

Then we can equivalently write

$$z_i(x_{i+1}) = z_i(x_i) + \sum_{j=0}^{ml-1} \int_{z_j^i}^{z_{j+1}^i} w_{ij}(l_i^s(t)) \, dt$$

$$+ \bar{h}^{q+1} ml \frac{1}{ml} \sum_{j=0}^{ml-1} \int_0^1 g_{ij}(u) \, du$$

$$+ \sum_{j=0}^{ml-1} \int_{z_j^i}^{z_{j+1}^i} \left(f(z_i(t)) - f(l_i^s(t)) \right) \, dt, \tag{23}$$

where function g_{ij} is defined for $u \in [0, 1]$ by

$$g_{ij}(u) = \frac{f(l_i^s(z_j^i + u\bar{h})) - w_{ij}(l_i^s(z_j^i + u\bar{h}))}{\bar{h}^q}. \tag{24}$$

(The value $l_i^s(z_{j+1}^i)$ is meant here as the limit $\lim l_i^s(t)$ as $t \to z_{j+1}^i$ from the left.)

To get the formula for y_{i+1}, we neglect the last right-hand side term in (23), and we approximate the penultimate term. We arrive at the formula

$$y_{i+1} = y_i + \sum_{j=0}^{ml-1} \int_{z_j^i}^{z_{j+1}^i} w_{ij}(l_i^s(t)) \, dt + \bar{h}^{q+1} ml \, \mathrm{AP}_i(f). \tag{25}$$

The vector $\mathrm{AP}_i(f)$ is constructed to approximate the mean of ml integrals

$$\mathrm{AP}_i(f) \approx \frac{1}{ml} \sum_{j=0}^{ml-1} \int_0^1 g_{ij}(u) \, du \tag{26}$$

as follows.

Approximate the integral of g_{ij} by the mid-point rule with N knots u_k, i.e.,

$$\int_0^1 g_{ij}(u) \, du \approx \frac{1}{N} \sum_{k=0}^{N-1} g_{ij}(u_k). \tag{27}$$

Hence,

$$\frac{1}{ml} \sum_{j=0}^{ml-1} \int_0^1 g_{ij}(u) \, du \approx \frac{1}{mlN} \sum_{j=0}^{ml-1} \sum_{k=0}^{N-1} g_{ij}(u_k). \tag{28}$$

The (random) vector $\mathrm{AP}_i(f)$ is computed, in the appropriate setting, by applying the optimal randomized or quantum algorithm (with repetitions) to approximate each component of the mean

of mlN vectors in the right-hand side of (28). For a discussion of algorithms for computing the mean, see [2,3]. The approximation error is required to be bounded by ε_1,

$$\left\| \mathrm{AP}_i(f) - \frac{1}{mlN} \sum_{j=0}^{ml-1} \sum_{k=0}^{N-1} g_{ij}(u_k) \right\| \leqslant \varepsilon_1, \tag{29}$$

with probability at least $1 - \delta_1$. The parameters N, ε_1 and δ_1 are to be chosen, and will be defined soon.

An approximation l^{s+1} to z over $[a, b]$ in the algorithm $A_{s+1}([a, b], n)$ is defined by $l^{s+1}(t) = l_i^s(t)$ for $t \in [x_i, x_{i+1})$, and $l^{s+1}(b) = l_{n-1}^s(b)$. This completes the definition of A_{s+1} and, by induction, of the sequence of algorithms $\{A_s([a, b], n)\}_{s \geqslant 1}$.

It remains to define the parameters m, l, N, ε_1 and δ_1. They are selected to assure the best performance of the algorithms, and result from solving a number of auxiliary optimization problems. We do not present here the rather technical analysis leading to the proper selection. We give instead the resulting values of the parameters, and prove in the next sections the correctness of our choice. To accomplish the definition of algorithm A_{s+1} we make the following choice of parameters:

$$(m, l, N) = \begin{cases} (n^2, n^{2^{s+1}-4}, n^{2^s-1}) & \text{in RAND}, \\ (n, n^{s-1}, n^s) & \text{in QUANT}, \end{cases} \tag{30}$$

and we set $\varepsilon_1 = 1/N$. The parameter δ_1 is chosen as a function of δ, the basic parameter n and k to be

$$\delta_1 = \begin{cases} 1 - (1 - \delta)^{1/n^{2^k-1}} & \text{in RAND}, \\ 1 - (1 - \delta)^{1/n^k} & \text{in QUANT}. \end{cases} \tag{31}$$

It is useful to illustrate with an example the choice of the basic parameter in the recursive procedure described above. Consider the randomized case with $s = 3$. We have that $m = n^2$, i.e., at each step a new basic parameter is obtained by squaring the current one. The algorithm $A_3([a, b], n)$ is thus defined by n applications of $A_2([x_i, x_{i+1}], n^2)$ for $0 \leqslant i \leqslant n - 1$. Computing $A_2([x_i, x_{i+1}], n^2)$ requires in turn applications of A_1 on subintervals of length $(x_{i+1} - x_i)/n^2 = (b - a)/n^3$, with the basic parameter $m = (n^2)^2 = n^4$. At the final step, the function l^3 is given by Taylor's approximations (polynomials) on subintervals of length $(b - a)/(n^3m) = (b - a)/n^7$. (For an arbitrary s, intervals in which l^s is a polynomial will be established in the next section.)

Let us remark that the algorithms A_1 and A_2 defined above are the well known Taylor algorithm and the algorithm analyzed in [6], respectively. For $k \geqslant 3$, algorithms A_k are new.

5. Performance analysis

This section is devoted to the proof of Theorem 1. We first show error bounds for the algorithm A_s, neglecting all probabilistic considerations related to the bounds. Such issues are considered in the next subsection, as well as the cost of the algorithm.

5.1. Error bounds

We shall prove (13). To derive error bounds, we first discuss the regularity of the approximation l^s computed in $A_s([a, b], n)$, and the regularity of the function g_{ij} given by (24). Note that the

function l^s is in general not continuous. It is a polynomial on each subinterval $[c, p)$ of a uniform partition of $[a, b]$ (the last subinterval ending with b is a closed one), with length

$$(b - a)/n^{2^s-1} \text{ in RAND} \quad \text{and} \quad (b - a)/n^s \text{ in QUANT.} \tag{32}$$

To prove this, note that the length of each subinterval in which the approximation l^1 in $A_1([a, b], n)$ is a polynomial equals to $(b-a)/n$. Let the length of each subinterval in which l^s in $A_s([a, b], n)$ is a polynomial be equal to $(b - a)/n^{\phi_s}$. From the definition, the function l_i^s is a polynomial on each subinterval of length

$$(b - a)/(nm^{\phi_s}) = \begin{cases} (b - a)/n^{1+2\phi_s} & \text{in RAND,} \\ (b - a)/n^{1+\phi_s} & \text{in QUANT.} \end{cases} \tag{33}$$

Since functions l_i^s define l^{s+1}, we get the recurrence relation

$$\phi_{s+1} = \begin{cases} 1 + 2\phi_s & \text{in RAND,} \\ 1 + \phi_s & \text{in QUANT,} \end{cases} \tag{34}$$

with $\phi_1 = 1$. This gives the desired statement (32).

Function l_i^s is thus a polynomial on each interval $[z_j^i, z_{j+1}^i)$, see (23), (24) and (25). We use this property to observe that the following fact concerning g_{ij} holds.

Fact 1. *There exist positive constants M_1 and M_2, depending only on the parameters of the class $F^{r,\rho}$, a and b (and independent of i, j, z_j^i, y_j^i), such that*

$$\|g_{ij}(u)\| \leqslant M_1, \quad u \in [0, 1], \tag{35}$$

$$\|g_{ij}(u_1) - g_{ij}(u_2)\| \leqslant M_2|u_1 - u_2|, \quad u_1, u_2 \in [0, 1], \tag{36}$$

and $\sup\{\max\{M_1, M_2\}: \quad b - a \leqslant M\} < \infty$ *for any $M > 0$.*

Proof. The proof follows from arguments used in the proof of Lemma on p. 828 in [5]. We replace $h := \bar{h}$, $x_i := z_j^i$, $x_{i+1} := z_{j+1}^i$, $l_i^* := l_i^s$, $y_i^* := l_i^s(z_j^i)$ and $w_i^* := w_{ij}$ in that Lemma. We use the fact that function l_i^s is a polynomial in $[z_j^i, z_{j+1}^i)$ of the form

$$l_i^s(t) = \sum_{k=0}^{r+1} (1/k!) z_{ij}^{(k)}(z_j^i)(t - z_j^i)^k,$$

where z_{ij} satisfies the equation $z_{ij}'(t) = f(z_{ij}(t))$, $t \in [z_j^i, z_{j+1}^i)$, with the initial condition $z_{ij}(z_j^i) = c$ resulting from the recursive definition of the algorithms. $\quad\square$

We are ready to prove (13). We proceed by induction. Let $s = 1$. The desired bound in this case is a version of the well-known error estimate for Taylor's method. The error of Taylor's expansion formula satisfies

$$\|z_i(t) - l_i^0(t)\| \leqslant \hat{M}(t - x_i)^{q+1},$$

for $t \in [x_i, x_{i+1}]$, where $\hat{M}$ depends only on the parameters of the class $F^{r,\rho}$, a, b and is bounded for bounded $b - a$. First estimating the error $\|z(x_i) - y_i\|$ in the usual way, and then passing to

the error over $[a, b]$, we arrive at the bounds

$$\sup_{t \in [a,b]} \|z(t) - l^1(t)\| \leqslant \begin{cases} e^{Lh}(e^{L(b-a)} - 1)L^{-1}\hat{M}h^q + \hat{M}h^{q+1} & \text{if } L(b-a) \text{ is arbitrary,} \\ 5\hat{M}(b-a)h^q & \text{if } L(b-a) \leqslant \ln 2. \end{cases}$$

Hence, (13) holds with

$$\bar{C}_1^1 = 2(e^{L(b-a)} - 1)L^{-1}\hat{M}(b-a)^q + \hat{M}(b-a)^{q+1}, \quad C_1^1 = 5\hat{M} \text{ and } \alpha_1 = q.$$

Suppose by induction that (13) holds for l^s, for any $[a, b]$ and n. Let $e_i = z(x_i) - y_i$. By the triangle inequality,

$$\|e_{i+1}\| \leqslant \|z(x_{i+1}) - z_i(x_{i+1})\| + \|z_i(x_{i+1}) - y_{i+1}\|. \tag{37}$$

From the dependence of the solution on initial conditions, we see that the first term is bounded by

$$\|z(x_{i+1}) - z_i(x_{i+1})\| \leqslant e^{Lh}\|e_i\|. \tag{38}$$

To estimate the second term, we subtract (25) from (23) to get

$$z_i(x_{i+1}) - y_{i+1} = \bar{h}^{q+1}ml\left(\frac{1}{ml}\sum_{j=0}^{ml-1}\int_0^1 g_{ij}(u)\,du - \text{AP}_i(f)\right)$$
$$+ \sum_{j=0}^{ml-1}\int_{z_j^i}^{z_{j+1}^i}\left(f(z_i(t)) - f(l_i^s(t))\right)dt. \tag{39}$$

From the definition of $\text{AP}_i(f)$, and the properties of g_{ij} stated in Fact 1, we write the bound

$$\left\|\frac{1}{ml}\sum_{j=0}^{ml-1}\int_0^1 g_{ij}(u)\,du - \text{AP}_i(f)\right\| \leqslant CN^{-1} + \varepsilon_1. \tag{40}$$

Here, the term CN^{-1} is the estimate of the mid-point rule error, and ε_1 comes from the randomized or quantum approximation of the mean of mlN vectors, see (29). (The last bound holds with probability at least $1 - \delta_1$.) The constant C depends only on the parameters of the class $F^{r,\rho}$, a and b, and is bounded for bounded $b - a$.

The second right-hand side term in (39) will be estimated from the definition of $l_i^s(t)$.

Let $n \geqslant L(b-a)/\ln 2$. Then $L(x_{i+1} - x_i) \leqslant \ln 2$, so that the second inequality in (13) holds for $A_s([x_i, x_{i+1}], m)$ (with $m = n^2$ in RAND and $m = n$ in QUANT). We have

$$\|f(z_i(t)) - f(l_i^s(t))\| \leqslant L\|z_i(t) - l_i^s(t)\| \leqslant LC_1^s(x_{i+1} - x_i)^{q+1}m^{-\alpha_s}. \tag{41}$$

(The coefficient C_1^s for $A_s([x_i, x_{i+1}], m)$ depends on x_i, x_{i+1}, but, due to the inductive assumption on C_1^s, it is bounded by a constant depending only on the parameters of the class $F^{r,\rho}$, a and b. We denote the bound by the same symbol C_1^s.)

Putting together relations (38)–(41), we can use (37) to see that

$$\|e_{i+1}\| \leqslant e^{Lh}\|e_i\| + \bar{h}^{q+1}ml\left(CN^{-1} + \varepsilon_1\right) + LC_1^s(b-a)^{q+2}n^{-(q+2)}m^{-\alpha_s}, \tag{42}$$

for $i = 0, 1, \ldots, n-1$, where $e_0 = 0$. Hence,

$$\|e_i\| \leqslant \frac{e^{Lhn} - 1}{e^{Lh} - 1}\left(\bar{h}^{q+1}ml\left(CN^{-1} + \varepsilon_1\right) + LC_1^s(b-a)^{q+2}n^{-(q+2)}m^{-\alpha_s}\right), \tag{43}$$

for $i = 0, 1, \ldots, n$.

We now pass to the error in $[a, b]$. For $t \in [x_i, x_{i+1}]$, we have

$$\|z(t) - l^{s+1}(t)\| = \|z(t) - l_i^s(t)\| \leqslant \|z(t) - z_i(t)\| + \|z_i(t) - l_i^s(t)\|$$
$$\leqslant e^{Lh}\|e_i\| + C_1^s h^{q+1}m^{-\alpha_s}. \tag{44}$$

We now use (43) in (44). Since $e^{Lh} - 1 \geqslant Lh$, we get for arbitrary $L(b-a)$ and $n \geqslant L(b-a)/\ln 2$ that

$$\sup_{t \in [a,b]} \|z(t) - l^{s+1}(t)\| \leqslant 2\left(e^{L(b-a)} - 1\right)L^{-1}(b-a)^q(mln)^{-q}(CN^{-1} + \varepsilon_1)$$
$$+ \left(2e^{L(b-a)} - 1\right)C_1^s(b-a)^{q+1}n^{-(q+1)}m^{-\alpha_s}. \tag{45}$$

If the interval $[a, b]$ is small enough that $L(b-a) \leqslant \ln 2$, then $\exp(L(b-a)) - 1 \leqslant 2L(b-a)$, and we obtain

$$\sup_{t \in [a,b]} \|z(t) - l^{s+1}(t)\| \leqslant 4(b-a)^{q+1}(mln)^{-q}(CN^{-1} + \varepsilon_1)$$
$$+ (4L(b-a) + 1)C_1^s(b-a)^{q+1}n^{-(q+1)}m^{-\alpha_s}. \tag{46}$$

Bounds (45) and (46) allow us to finish the inductive proof of (13). To close the induction, we continue with two cases, randomized and quantum, introducing proper parameters m, l, N and ε_1 given in (30).

Consider the randomized setting. Introducing the parameters we get

$$\sup_{t \in [a,b]} \|z(t) - l^{s+1}(t)\| \leqslant \begin{cases} \bar{C}_1^{s+1}n^{-\alpha_{s+1}} & \text{if } L(b-a) \text{ is arbitrary} \\ & \text{and } n \geqslant L(b-a)/\ln 2, \\ C_1^{s+1}(b-a)^{q+1}n^{-\alpha_{s+1}} & \text{if } L(b-a) \leqslant \ln 2 \text{ and } n \in \mathbf{N}, \end{cases} \tag{47}$$

where

$$\alpha_{s+1} = \min\{(2^{s+1} - 1)q + 2^s - 1, \ q + 1 + 2\alpha_s\}. \tag{48}$$

Constants $\bar{C}_1^{s+1}$ and C_1^{s+1} are given by

$$\bar{C}_1^{s+1} = (b-a)^q\left(2\left(e^{L(b-a)} - 1\right)L^{-1}(C+1) + \left(2e^{L(b-a)} - 1\right)C_1^s(b-a)\right) \tag{49}$$

and

$$C_1^{s+1} = 4(C_1^s + C + 1). \tag{50}$$

Since α_s is given by (14), the same holds due to (48) for α_{s+1}, and the induction is closed in the randomized setting.

In the quantum setting, after introducing the parameters from (30), we arrive at the bound (47), where only the exponent α_{s+1} differs from that in the randomized setting. It is now given by

$$\alpha_{s+1} = \min\{\, (s+1)q + s\,,\; q + 1 + \alpha_s\,\}. \tag{51}$$

Constants $\bar{C}_1^{s+1}$ and C_1^{s+1} remain the same as in the randomized setting, see (49) and (50).

Since α_{s+1} agrees with (14), the inductive proof of (13) is completed in the quantum setting.

5.2. Probability and cost

In the previous subsection we neglected probability issues related to the bounds obtained. The cost of computations was also not considered. We now show that (13) holds with probability at least $1 - \delta$, and we prove the cost bound (15).

We first show that

$$\text{bound (13) holds with probability at least } (1 - \delta_1)^{\psi(n,s)}, \tag{52}$$

where $\psi(n, 1) = 0$ for all n, and

$$\psi(n, s) = \begin{cases} \displaystyle\sum_{i=1}^{s-1} n^{2^i - 1} & \text{in RAND}, \\[2mm] \displaystyle\frac{n^s - n}{n - 1} & \text{in QUANT}, \end{cases} \tag{53}$$

for $s \geqslant 2$ and $n \geqslant 2$. Indeed, for $s = 1$ the error bound for Taylor's algorithm holds with certainty, that is, $\psi(n, 1) = 0$. For $s = 2$, (13) holds with probability at least $(1 - \delta_1)^n$, so that $\psi(n, 2) = n$, which agrees with (53). Suppose that (52) and (53) hold for some $s \geqslant 2$. The error estimate for l^{s+1} holds true if the error bounds for l_i^s and bounds (29) are satisfied for $i = 0, 1, \ldots, n - 1$. This holds with probability at least $(1 - \delta_1)^{n\psi(m,s)+n}$. Hence, we have that

$$\psi(n, s+1) = n\psi(m, s) + n. \tag{54}$$

This yields that $\psi(n, s + 1)$ satisfies (53), which follows after introducing $m = n^2$ in the randomized case, and $m = n$ in the quantum case.

Note that for $n \geqslant 5$, we have

$$\psi(n, s) \leqslant \begin{cases} n^{2^s - 1} & \text{in RAND}, \\ n^s & \text{in QUANT}. \end{cases}$$

This, and the choice of δ_1 given in (31), assures that

$$(1 - \delta_1)^{\psi(n,s)} \geqslant 1 - \delta$$

for $s = 1, 2, \ldots, k$, as claimed.

We now prove the cost estimate (15) by induction. For $s = 1$, the cost is bounded by $c_1(r, d)n$ classical evaluations of components of f or its partial derivatives, where $c_1(r, d)$ depends only on r and d. Hence, (15) holds with $\beta_1 = 1$.

Suppose that (15) is satisfied for A_s. The costs of computing successive components that define the approximation l^{s+1} in A_{s+1} are bounded as follows. The cost of computing l_i^s for all i is

bounded by

$$C_2^s nm^{\beta_s}\left(\beta_k \log m + \log \frac{1}{\delta}\right),$$

and the cost of w_{ij} for all i, j by

$$c_2(r, d)nml,$$

where $c_2(r, d)$ depends only on r and d. Here, C_2^s can be chosen to depend only on the parameters of the class $F^{r,\rho}$, a and b, and is bounded for bounded $b - a$.

It remains to take into account the cost of computing $\text{AP}_i(f)$, $i = 0, 1, \ldots, n - 1$. We shall use the results on computation of the mean of real numbers, those of Mathé [3] in the randomized setting, and of Brassard et al. [1] in the quantum setting. Let $\kappa = 2$ in RAND, and $\kappa = 1$ in QUANT. To compute $\text{AP}_i(f)$ for $i = 0, 1, \ldots, n - 1$, each satisfying (29), we need at most

$$Pn \min\left\{mlN, \left(\frac{1}{\varepsilon_1}\right)^{\kappa}\right\} \log \frac{1}{\delta_1}$$

evaluations of f in RAND, or quantum queries on f in QUANT. The constant P depends only on the parameters of the class $F^{r,\rho}$, a and b, and is bounded for bounded $b - a$. The logarithmic factor is related to the number of repetitions needed to increase probability of (29). For the basic algorithm for computing the mean, (29) holds with probability at least $\frac{3}{4}$, and for our purposes it must be at least $1 - \delta_1$. For a discussion of the role of repetitions and computing the median of a number of results, see [2].

Putting together these bounds, we get

$$\text{cost}(A_{s+1}) \leqslant C_2^s nm^{\beta_s}\left(\beta_k \log m + \log \frac{1}{\delta}\right) + c_2(r, d)nml$$

$$+ Pn \min\left\{mlN, \left(\frac{1}{\varepsilon_1}\right)^{\kappa}\right\} \log \frac{1}{\delta_1}. \tag{55}$$

Note that

$$\log \frac{1}{\delta_1} = \log \frac{1}{1 - (1 - \delta)^{1/n^{\beta_k}}} \leqslant c\left(\log \frac{1}{\delta} + \beta_k \log n\right),$$

where c is an absolute constant.

Finally, we introduce in (55) the parameters m, l, N and ε_1 defined by (30).

In the randomized case, we arrive at

$$\text{cost}(A_{s+1}) \leqslant C_2^{s+1} n^{\beta_{s+1}}\left(\beta_k \log n + \log \frac{1}{\delta}\right), \tag{56}$$

where $\beta_{s+1} = \max\{2\beta_s + 1, 2^{s+1} - 1\}$ (which agrees with (16)), and $C_2^{s+1} = 2C_2^s + c_2(r, d) + Pc$.

In the quantum case, (56) holds with $\beta_{s+1} = \max\{\beta_s + 1, s + 1\}$ (which again agrees with (16)) and $C_2^{s+1} = C_2^s + c_2(r, d) + Pc$.

The inductive proof of (15) is thus completed. This also ends the proof of Theorem 1.

6. Final remarks

We have established, up to arbitrarily small γ in the exponent and a logarithmic factor, the randomized and quantum complexity of initial-value problems. The matching upper bound is

obtained by defining a sequence of randomized and quantum algorithms for solving the problem. The main issues here are the following:

- the use of integral identity (23) satisfied by the solution,
- proper application of randomized and quantum algorithms for computing the mean of real numbers, and
- the recursive definition of the algorithms. To compute the approximation at the final level k, we pass a number of times through all the proceeding levels.

Let us summarize the results in various settings. Neglecting γ and the logarithmic factor, we have that the complexity is of order $(1/\varepsilon)^{1/(q+1)}$ in the worst-case deterministic setting with linear information [4], and the same holds in the quantum case. In the worst-case deterministic setting with standard information, it is of order $(1/\varepsilon)^{1/q}$, see [4]. In the randomized setting, we stay between these results with complexity of order $(1/\varepsilon)^{1/(q+1/2)}$.

Acknowledgments

References

[1] G. Brassard, P. Høyer, M. Mosca, A. Tapp, Quantum amplitude amplification and estimation ⟨http://arXiv.org/abs/quant-ph/0005055⟩, 2000.

[2] S. Heinrich, Quantum summation with an application to integration, J. Complexity 18 (2002) 1–50.

[3] S. Heinrich, E. Novak, Optimal summation and integration by deterministic, randomized, and quantum algorithms, in: K.-T. Fang, F. J. Hickernell, H. Niederreiter (Eds.), Monte Carlo and Quasi-Monte Carlo Methods 2000, Springer, Berlin, 2002, pp. 50–62, see also ⟨http://arXiv.org/abs/quant-ph/0105114⟩.

[4] B. Kacewicz, Optimal solution of ordinary differential equations, J. Complexity 3 (1987) 451–465.

[5] B. Kacewicz, Randomized and quantum algorithms yield a speed-up for initial-value problems, J. Complexity 20 (2004) 821–834 see also ⟨http://arXiv.org/abs/quant-ph/0311148⟩.

[6] B. Kacewicz, Improved bounds on the randomized and quantum complexity of initial-value problems, J. Complexity 21 (2005) 740–756 see also ⟨http://arXiv.org/abs/quant-ph/0405018⟩.