JOURNAL OF INTEGRAL EQUATIONS
AND APPLICATIONS
Volume 24, Number 4, Winter 2012

MULTILEVEL AUGMENTATION METHODS WITH MATRIX COMPRESSION FOR SOLVING REFORMULATED HAMMERSTEIN EQUATIONS

XIANGLING CHEN, ZHONGYING CHEN AND BIN WU

Communicated by Kendall Atkinson

ABSTRACT. In this paper we supplement matrix truncation strategies with the multilevel augmentation methods for solving the reformulated Hammerstein equations. The resulting numerical solutions have nearly optimal convergence order with linear order computational complexity up to a logarithmic factor with respect to the dimension of the discretization subspace. Numerical experiments on one and two dimensional equations illustrate that our algorithm gains remarkably high efficiency without losing accuracy.

1. Introduction. The Hammerstein equation is an important kind of nonlinear integral equation. It serves as a mathematical model for many applications, such as astrophysics, fluid dynamics, cell kinetics, mathematical economies and so on. Moreover, it is used as a model equation to test numerical methods for solving nonlinear integral equations. There are various numerical methods for solving the Hammerstein equation, which include projection methods, Nyström methods, homotopy analysis, domain decomposition, etc. In [**1**], several popular numerical schemes are reviewed. For more recent developments on the numerical methods, see [**4, 5, 18–20, 22, 28, 29**]. All of these methods involve evaluating the Jacobian matrix. Thanks to the

2010 AMS *Mathematics subject classification.* Primary 65M38, 45L05.
Keywords and phrases. Hammerstein equations, collocation methods, multilevel augmentation methods, matrix compression.
The first author was supported in part by the Fundamental Research Funds for the Central Universities from Sun Yat-Sen University and China Postdoctoral Science Foundation under grant 2012M521912. The second author was supported in part by the Natural Science Foundation of China under grants 10771224, 11071264, and by the Science and Technology Section of SINOPEC. The third author was supported in part by the Natural Science Foundation of China under grants 10771224 and 10801138, by the Fundamental Research Funds for the Central Universities from Sun Yat-Sen University.
The third author is the corresponding author.
Received by the editors on February 27, 2011, and in revised form on November 14, 2011.

DOI:10.1216/JIE-2012-24-4-513

structure of the nonlinear operator emerging in the equation, it requires evaluating integrals while we update the Jacobian matrix, which brings high computational complexity to numerical algorithms. The multigrid method effectively utilizes the information on coarse grids, so that the difficulty coming from the nonlinearity can be overcome to some extent. However, the method still has an operation count of $\mathcal{O}(N^2)$, where N is the dimension of the discretization subspace.

In [**27**], Kumar and Sloan gave a reformulation of the Hammerstein equation, the solution of which is equivalent to the original Hammerstein equation. It was observed that the discretization equation resulting from the collocation methods for solving the reformulated equation can be solved much faster since, when we update the Jacobian matrix, we make use of the representation matrix of the integral operator emerging in the equation and need not evaluate the associated integrals. Due to this advantage, the efficiency of the numerical algorithm is comparable with solving the corresponding linear integral equations of the second kind. See also [**25, 26**] for more discussion of the algorithm.

The multilevel augmentation method is introduced in [**11**] in abstract form for solving operator equations. Then it is used to solve integral equations in [**14**], differential equations in [**12**] and ill-posed problems in [**16**]. It is applied for solving Hammerstein equations in [**15**], which is then extended to the fast solution of nonlinear integral equations resulting from boundary value problems with nonlinear boundary conditions (cf. [**6, 7**]). Recently, it has been used to solve the reformulated Hammerstein equations in [**8**]. The key idea of the multilevel augmentation methods is to invert the (linear or nonlinear) operator in a small subspace rather than in the whole discretization space, which avoids establishing Jacobian matrices and solving linear systems in large scale. This has been proved helpful in improving computational efficiency by theoretical results and numerical experiments (cf. [**11, 14, 15**]). However, we still need to generate the representation matrix of the integral operator, which is a full matrix in general. Therefore, the computational complexity of the algorithm in [**8**], which is not equipped with truncation process, is obviously of quadratic order.

In this paper we embed truncation techniques for representation matrices of integral operators into the algorithm from [**8**]. For a representation matrix associated with multiscale bases, truncation strategies are established for weakly singular operators in [**9**] and for smooth op-

erators in [**16**]. In both cases, the matrices can be truncated into sparse matrices, the number of the nonzero entries of which is linear up to a logarithmic factor with respect to the dimension of the matrices. Our algorithm is very efficient because of three reasons. Firstly, the Hammerstein equation is reformulated and then discretized by collocation methods, which avoids evaluation of integrals during the update of Jacobian matrices. Secondly, the multilevel augmentation methods based on multiscale discretization are applied to avoid matrix inversion in the whole discretization subspace. Finally, matrix truncation remarkably reduces the number of the integrals to be evaluated in the algorithm.

However, we are faced with two problems in integrating the truncation techniques with the multilevel augmentation methods. Firstly, truncation is basically a method of approximation and will introduce error to the algorithm. Thus, we have to analyze the error induced from truncation to the resulting numerical solutions. Secondly, when the multiscale bases are used to establish collocation schemes for solving integral equations, the associated collocation functionals are not single point evaluations, but linear combinations of point evaluations. Correspondingly, the values of entries of the representation matrix of the integral operator are linear combinations of function values. However, because of the existence of the nonlinear operator, the creation of the Jacobian matrix needs function values on single points. This needs special tricks to close the gap. In this paper we will address the above two issues.

We organize the paper in four sections. In Section 2 we establish the multilevel augmentation methods for solving the reformulated Hammerstein equation with truncation, where the truncation is characterized by a condition on the error between the original and truncated integral operators. The error estimate is then given for the algorithm. In Section 3 the concrete truncation strategies for two types of integral operators are described and proved to satisfy the condition in Section 2. Then the discrete form of the multilevel augmentation methods is proposed, and the computational efforts are estimated according to the computing steps. In Section 4, four numerical experiments are carried out to compare our algorithm and that from [**8**]. We also compare our algorithm with fast multilevel augmentation methods for solving Hammerstein equations from [**6**].

2. The multilevel augmentation methods for the reformulated Hammerstein equations. In this section we first review the collocation scheme for solving the reformulated Hammerstein equations, as well as the corresponding multilevel augmentation scheme proposed in [**8**]. Then we state strategies for truncating the representation matrix of the integral operator to establish a fast algorithm. The accuracy of the resulting numerical solutions is estimated.

2.1. The collocation scheme and multilevel augmentation methods without matrix truncation. For $d \geq 1$, let $E \subseteq \mathbf{R}^d$ be a compact domain. We consider solving the Hammerstein equation

$$u(s) - \int_E K(s,t)\psi(t,u(t))\,dt = f(s), \quad s \in E, \tag{2.1}$$

where K, f and ψ are given functions and u is the unknown to be determined. We assume that $f \in C(E)$, the space of all continuous functions on E, and for any $s \in E$, we denote $K_s(t) := K(s,t)$. The following assumptions are usually imposed when we consider the numerical solution of equation (2.1).

(H1) $\lim_{s\to t} \|K_s - K_t\|_1 = 0$ for any $t \in E$, and $\sup_{s\in E} \int_E |K(s,t)|\,dt < \infty$;

(H2) $\psi(t,u)$ is continuous in $t \in E$ and Lipschitz continuous in $u \in \mathbf{R}$, the partial derivative $D_u\psi$ of ψ, with respect to the variable u, exists and is Lipschitz continuous, and for any $u \in C(E)$, $\psi(\cdot,u(\cdot)), D_u\psi(\cdot,u(\cdot)) \in C(E)$.

Let $\mathbf{X} := L^\infty(E)$, the space of essentially bounded functions on E. We define the linear integral operator $\mathcal{K} : \mathbf{X} \to \mathbf{X}$ by

$$(\mathcal{K}u)(s) := \int_E K(s,t)u(t)\,dt, \quad s \in E,$$

and the nonlinear operator $\Psi : \mathbf{X} \to \mathbf{X}$ by

$$(\Psi u)(s) := \psi(s,u(s)), \quad s \in E.$$

With the above notations, (2.1) is written in the operator form

$$u - \mathcal{K}\Psi u = f. \tag{2.2}$$

To obtain the reformulation of (2.2), we define

$$z := \Psi(u). \tag{2.3}$$

Then z satisfies the nonlinear equation

$$z - \Psi(f + \mathcal{K}z) = 0. \tag{2.4}$$

It is proved in [**27**] that, if u^* is an isolated solution of (2.2), then $z^* = \Psi(u^*)$ is an isolated solution of (2.4). Conversely, if z^* is an isolated solution of (2.4), then $u^* = f + \mathcal{K}z^*$ is an isolated solution of (2.2). Note that, in (2.4), the unknown function z is first applied by the linear operator $\mathcal{K}$ and then by the nonlinear operator Ψ, which brings convenience to the numerical treatment (cf. [**26, 27**]). Specifically, if we discretize the equations by projection methods and then apply Newton iteration method to solving the resulting nonlinear system, the computational efforts for updating the Jacobian matrix associated with (2.4) are significantly lower than that of (2.2). Moreover, when we have an approximate solution z_n of (2.4), the corresponding approximate solution u_n of (2.2) can be obtained through

$$u_n = f + \mathcal{K}z_n.$$

It is stated in [**26**] that, under suitable conditions, u_n has a higher convergence rate than z_n, which is called a *superconvergence* property.

Now we establish the collocation scheme for solving (2.4). To this end, we choose a sequence of finite-dimensional subspaces $\mathbf{X}_n$, $n \in \mathbf{N}_0 := \{0, 1, 2, \dots\}$ of $\mathbf{X}$ with the property

$$C(E) \subset \overline{\bigcup_{n \in \mathbf{N}_0} \mathbf{X}_n} \subset \mathbf{X}.$$

For each $n \in \mathbf{N}_0$, let $\mathcal{P}_n : \mathbf{X} \to \mathbf{X}_n$ be the interpolatory projection operator onto $\mathbf{X}_n$. We remark that, although the point evaluation functional is not well defined on $\mathbf{X}$, there is a norm preserving extension of the point evaluation functionals on $C(E)$ to $\mathbf{X}$. See [**3**] for the details. Throughout this paper, we assume that

(H3) the sequence $\mathcal{P}_n$, $n \in \mathbf{N}_0$ converges pointwise to the identity operator in $C(E)$, i.e., for any $x \in C(E)$, there holds

$$\lim_{n \to \infty} \|\mathcal{P}_n x - x\| = 0.$$

The collocation method for solving (2.4) is to find $z_n \in \mathbf{X}_n$ such that

$$z_n - \mathcal{P}_n \Psi(f + \mathcal{K} z_n) = 0. \tag{2.5}$$

The following theorem was established in [**27**] for the existence and convergence property of z_n.

Theorem 2.1. *Let z^* be an isolated solution of* (2.4). *Suppose that the assumptions* (H1)–(H3) *hold and* 1 *is not an eigenvalue of the linear operator $\Psi'(f + \mathcal{K} z^*)\mathcal{K}$. Then there exists a neighborhood of z^* and a positive integer N such that, for all $n > N$,* (2.5) *has a unique solution z_n in the neighborhood. Moreover, there exist two positive constants c_1 and c_2 such that*

$$c_1 \|z^* - \mathcal{P}_n z^*\|_\infty \le \|z^* - z_n\|_\infty \le c_2 \|z^* - \mathcal{P}_n z^*\|_\infty.$$

In the sense of the above theorem, the collocation solution z_n is *quasi-optimal* to approximate z^* in $\mathbf{X}_n$.

For any positive integer n, we define the index set $\mathbf{Z}_n := \{0, 1, \dots, n-1\}$. The classical way to obtain the discretization form of (2.5) is by choosing a basis $\{w_j : j \in \mathbf{Z}_{d_n}\}$ of the subspace $\mathbf{X}_n$ and a set of points $\{t_i : i \in \mathbf{Z}_{d_n}\}$, where $d_n := \dim(\mathbf{X}_n)$, and assuming $z_n = \sum_{j \in \mathbf{Z}_{d_n}} a_j w_j$, such that

$$\sum_{j \in \mathbf{Z}_{d_n}} a_j w_j(t_i) - \Psi\left(f(t_i) + \sum_{j \in \mathbf{Z}_{d_n}} a_j \mathcal{K} w_j(t_i) \right) = 0, \quad i \in \mathbf{Z}_{d_n}. \tag{2.6}$$

For $i, j \in \mathbf{Z}_{d_n}$, let

$$\widehat{J}_{ij} := w_j(t_i) - \Psi'\left(f(t_i) + \sum_{j \in \mathbf{Z}_{d_n}} a_j \mathcal{K} w_j(t_i) \right) \mathcal{K} w_j(t_i).$$

Then the matrix $\widehat{\mathbf{J}}_n := [\widehat{J}_{ij} : i, j \in \mathbf{Z}_{d_n}]$ is the Jacobian matrix of the Newton iteration method for solving (2.6). Let

$$\widehat{K}_{ij} := \mathcal{K} w_j(t_i), \quad i, j \in \mathbf{Z}_{d_n}.$$

We call $\widehat{\mathbf{K}}_n := [\widehat{K}_{ij} : i, j \in \mathbf{Z}_{d_n}]$ the *representation matrix* of the operator $\mathcal{K}$ with respect to $\{w_j : j \in \mathbf{Z}_{d_n}\}$ and $\{t_i : i \in \mathbf{Z}_{d_n}\}$. We remark that, although $\widehat{\mathbf{K}}_n$ is not the representation matrix of $\mathcal{K}$ in the classical sense, it is the matrix which corresponds to $\mathcal{K}$ with respect to $\{w_j : j \in \mathbf{Z}_{d_n}\}$ and $\{t_i : i \in \mathbf{Z}_{d_n}\}$. Therefore, in this paper, we abuse the terminology to cite it. Note that

$$\widehat{J}_{ij} = w_j(t_i) - \Psi'\bigg(f(t_i) + \sum_{j \in \mathbf{Z}_{d_n}} a_j \widehat{K}_{ij}\bigg)\widehat{K}_{ij}.$$

Therefore, once the representation matrix $\widehat{\mathbf{K}}_n$ is established, the update of the Jacobian matrix does not involve evaluating integrals, which results in a significant reduction of computational efforts of the Newton iteration method.

There are two time-consuming parts in the above numerical process. One is creating the matrix $\widehat{\mathbf{K}}_n$, especially for the case of a singular kernel. The other is inverting the current Jacobian matrix. Thanks to the global property of integral operators, the representation matrix $\widehat{\mathbf{K}}_n$ is in general a full matrix, so is $\widehat{\mathbf{J}}_n$. When the dimension of $\mathbf{X}_n$ is large, solving a linear system with the coefficient matrix $\widehat{\mathbf{J}}_n$ is usually difficult. In [**8**], a multiscale method for solving (2.5) is proposed, which borrows the idea from [**15**] to avoid inverting matrices in the whole discretization subspace. We briefly describe the algorithm below for reference. Given a fixed $k \in \mathbf{N}_0$, for any integer $n > k$, let $m := n - k$. Then we obtain an approximate solution z_{km} as follows. Let $z_{k0} := z_k$ and, for $l = 1, 2, \ldots, m$, we solve z_{kl} from

$$z_{kl} - \mathcal{P}_k \Psi(\mathcal{K}_{k+l} z_{kl} + f) = (\mathcal{P}_{k+l} - \mathcal{P}_k)\Psi(\mathcal{K}_{k+l} z_{k,l-1} + f). \tag{2.7}$$

Note that the subspace sequence $\{\mathbf{X}_n : n \in \mathbf{N}_0\}$ is nested, which allows us to make a multiscale decomposition. Specifically, for any $n \in \mathbf{N}_0$, we decompose $\mathbf{X}_{n+1}$ into the direct sum of $\mathbf{X}_n$ and its orthogonal complement $\mathbf{W}_{n+1}$ in $\mathbf{X}_{n+1}$, i.e., $\mathbf{X}_{n+1} = \mathbf{X}_n \oplus^{\perp} \mathbf{W}_{n+1}$. For any $k, m \in \mathbf{N}_0$, we define

$$\mathbf{W}_{km} := \mathbf{W}_{k+1} \oplus^{\perp} \mathbf{W}_{k+2} \oplus^{\perp} \cdots \oplus^{\perp} \mathbf{W}_{k+m}.$$

It is shown that (cf. [**15**]) equation (2.7) is equivalent to solving $z_{kl}^H \in \mathbf{W}_{kl}$ from

$$z_{kl}^H = (\mathcal{P}_{k+l} - \mathcal{P}_k)\Psi(\mathcal{K}_{k+l} z_{k,l-1} + f), \tag{2.8}$$

then solving $z_{kl}^L \in \mathbf{X}_k$ from

$$\mathcal{P}_k(z_{kl}^L + z_{kl}^H) = \mathcal{P}_k\Psi(\mathcal{K}_{k+l}(z_{kl}^L + z_{kl}^H) + f), \tag{2.9}$$

and letting $z_{kl} = z_{kl}^L + z_{kl}^H$. We observe that (2.8) is a *linear equation*, while (2.9) is nonlinear. Therefore, the algorithm does not invert the nonlinear operator in $\mathbf{X}_n$, but only in its subspace $\mathbf{X}_k$. If $\mathbf{X}_k$ is *much smaller* than $\mathbf{X}_n$, the computational cost will be remarkably cut down. It is proved in [**8**] that the output z_{km} of the algorithm has the same accuracy with the solution z_{k+m} of (2.5).

2.2. Multilevel augmentation methods equipped with matrix truncation. Although the algorithm from [**8**] avoids the computational efforts of inverting matrices in the whole discretization subspace, we still need to create the full matrix $\widehat{\mathbf{K}}_n$, which requires much computing time because of the evaluation of integrals. There are various ways to reduce the computational efforts of creating $\widehat{\mathbf{K}}_n$, one of which is the use of multiscale bases (cf. [**9, 13, 14, 17**]). Multiscale bases with special properties can lead to a *numerically sparse* representation matrix of integral operators, i.e., a majority of the entries of the matrix are *quite small*. By setting these entries to zero, numerically sparse matrices can be *truncated* into sparse matrices. Since truncation leads to numerical errors, truncation strategies are needed to determine the positions of entries to be truncated, such that the truncation errors are *controllable*.

In order to describe the truncation, we require the discretization subspaces to have the following properties. Firstly, the sequence $\{\mathbf{X}_n : n \in \mathbf{N}_0\}$ is nested. Moreover, let $W^{r,\infty}(E)$ be the set of all functions u on E, such that, for all $\alpha := (\alpha_0, \alpha_1, \ldots, \alpha_{d-1})$ with $|\alpha| := \sum_{i\in\mathbf{Z}_d} \alpha_i \le r$, $D^\alpha u \in L^\infty(E)$, where

$$D^\alpha u(x) := \frac{\partial^{|\alpha|} u(x)}{\partial^{\alpha_0} x_0 \cdots \partial^{\alpha_{d-1}} x_{d-1}}$$

for any $x := (x_0, x_1, \ldots, x_{d-1}) \in \mathbf{R}^d$. With the norm

$$\|u\|_{r,\infty} := \max\{\|D^\alpha u\|_\infty : |\alpha| \le r\},$$

$W^{r,\infty}(E)$ is a Banach space. We require that there exists a positive integer $\mu > 1$, such that for any $u \in W^{r,\infty}(E)$,

$$\operatorname{dist}(u, \mathbf{X}_n) \le c\mu^{-rn/d}\|u\|_{r,\infty}, \tag{2.10}$$

where dist (A, B) denotes the distance between two point sets A and B. Throughout the paper, we use c as a generic positive constant whose value may change with the context. In practice, $\mathbf{X}_n$ are chosen as the spaces of piecewise polynomials. The parameter r is then associated with the order of the polynomials as well as the regularity of u. In [**9**] it is shown how to construct the nested subspaces which meet the above requirements.

For different types of integral kernels, the concrete truncation methods are different. For weakly singular integral operators, truncation strategies based on discretization by collocation scheme with multiscale bases have been discussed systematically in [**9**], and those for smooth integral operators are given in [**16**]. However, we do not intend to give the concrete truncation strategies in this section because of two considerations. The first is that describing the strategies requires introducing lots of symbols, such as the bases. The second is that we hope to establish a uniform framework for the error estimate for various truncation strategies. Therefore, we would like to postpone the description of truncation strategies to the next section but only identify the errors between the *original* integral operator and the *truncated* integral operator. Specifically, let $\mathcal{K}_n := \mathcal{P}_n\mathcal{K}|_{\mathbf{X}_n}$ be the approximate operator of $\mathcal{K}$ on $\mathbf{X}_n$, and denote by $\widetilde{\mathcal{K}}_n$ the operator resulting from truncating $\mathcal{K}_n$. Note that the operators and their representation matrices are in a one-to-one correspondence. When we truncate a representation matrix, we are also truncating the corresponding operator. We assume the following condition on the error $\mathcal{K}_n - \widetilde{\mathcal{K}}_n$.

(H4) Let $\{\beta_n : n \in \mathbf{N}_0\}$ be a sequence of positive numbers with

$$\beta_n \longrightarrow 0, \quad n \to \infty.$$

For any $n \in \mathbf{N}_0$ and $v \in \mathbf{X}_n$,

$$\|(\mathcal{K}_n - \widetilde{\mathcal{K}}_n)v\|_\infty \leq \beta_n \|v\|_\infty.$$

Moreover, if $z^* \in W^{r,\infty}(E)$ and there is a positive constant c' such that $\|v - z^*\|_\infty \leq c'\mu^{-rn/d}\|z^*\|_{r,\infty}$, then

$$\|(\mathcal{K}_n - \widetilde{\mathcal{K}}_n)v\|_\infty \leq c(n+1)\mu^{-rn/d}\|z^*\|_{r,\infty}. \tag{2.11}$$

In the next section, we will describe truncation strategies for two types of integral kernels and verify that they both satisfy the above hypothesis.

Replacing the operator $\mathcal{K}$ in (2.5) by $\widetilde{\mathcal{K}}_n$, we obtain the following *perturbed* equation

$$\widetilde{z}_n - \mathcal{P}_n\Psi(f + \widetilde{\mathcal{K}}_n\widetilde{z}_n) = 0. \tag{2.12}$$

The following lemma establishes the unique solvability and approximate property of the numerical solution of (2.12).

Lemma 2.2. *Let z^* be an isolated solution of* (2.4). *Suppose that the assumptions* (H1)–(H4) *hold and* 1 *is not an eigenvalue of the linear operator $\Psi'(f + \mathcal{K}z^*)\mathcal{K}$. Then there exists a neighborhood of z^* and a positive integer N such that, for all $n > N$,* (2.12) *has a unique solution $\widetilde{z}_n$ in the neighborhood. Moreover, there exists a positive constant c such that*

$$\|z^* - \widetilde{z}_n\|_\infty \le c(n+1)\mu^{-rn/d}\|z^*\|_{r,\infty}.$$

Proof. Note that $\widetilde{\mathcal{K}}_n$ is an approximate operator of $\mathcal{K}_n$. With a similar idea to proving the unique solvability of (2.5), we can prove that (2.12) has a unique solution $\widetilde{z}_n$ in the neighborhood of z^*. For the estimate of the error $z^* - \widetilde{z}_n$, we subtract (2.12) from (2.5) to obtain

$$z_n - \widetilde{z}_n - \mathcal{P}_n[\Psi(f + \mathcal{K}_n z_n) - \Psi(f + \widetilde{\mathcal{K}}_n\widetilde{z}_n)] = 0.$$

Define

$$\mathcal{R}(z_n, \widetilde{z}_n) := \Psi(f + \mathcal{K}_n z_n) - \Psi(f + \widetilde{\mathcal{K}}_n\widetilde{z}_n) - \Psi'(\mathcal{K}_n z_n - \widetilde{\mathcal{K}}_n\widetilde{z}_n).$$

Since

$$\mathcal{K}_n z_n - \widetilde{\mathcal{K}}_n\widetilde{z}_n = (\mathcal{K}_n - \widetilde{\mathcal{K}}_n)z_n + \widetilde{\mathcal{K}}_n(z_n - \widetilde{z}_n),$$

we have

$$(\mathcal{I} - \mathcal{P}_n\Psi'\widetilde{\mathcal{K}}_n)(z_n - \widetilde{z}_n) = \mathcal{P}_n[\Psi'(\mathcal{K}_n - \widetilde{\mathcal{K}}_n)z_n + \mathcal{R}(z_n, \widetilde{z}_n)].$$

Note that $(\mathcal{I} - \mathcal{P}_n\Psi'\widetilde{\mathcal{K}}_n)^{-1}$ exist and are uniformly bounded for sufficiently large n, and

$$\|\mathcal{R}(z_n, \widetilde{z}_n)\|_\infty \le c'\left(\|(\mathcal{K}_n - \widetilde{\mathcal{K}}_n)z_n\|_\infty^2 + \|\widetilde{\mathcal{K}}_n(z_n - \widetilde{z}_n)\|_\infty^2\right)$$

for some positive constant c'. Thus,

$$\|z_n - \widetilde{z}_n\|_\infty \le c\|(\mathcal{K}_n - \widetilde{\mathcal{K}}_n)z_n\|_\infty.$$

The lemma is then concluded with hypothesis (H4) and the estimate from Theorem 2.1. □

Equation (2.12) has a similar form to (2.5). Therefore, the solution of (2.12) still encounters the difficulty of inverting the nonlinear operator in the whole discretization subspace. In the following algorithm, we utilize the idea of the algorithm from [**8**] to obtain an approximation $\widetilde{z}_{km}$ of the solution $\widetilde{z}_{k+m}$ of (2.12).

Algorithm 1: Multilevel augmentation method equipped with truncation strategy: the operator form. Let k be a fixed positive integer. Given $m \in \mathbf{N}_0$, the algorithm outputs $\widetilde{z}_{km}$, which is an approximation of the solution $\widetilde{z}_{k+m}$ of (2.12).

Step 1: Find the solution $\widetilde{z}_k$ of (2.12) with $n := k$. Set $\widetilde{z}_{k0} := \widetilde{z}_k$ and $l := 1$.

Step 2: Compute $\widetilde{z}_{kl}^H \in \mathbf{W}_{kl}$ by

$$\widetilde{z}_{kl}^H = (\mathcal{P}_{k+l} - \mathcal{P}_k)\Psi(\widetilde{\mathcal{K}}_{k+l}\widetilde{z}_{k,l-1} + f). \tag{2.13}$$

Step 3: Solve $\widetilde{z}_{kl}^L \in \mathbf{X}_k$ from the nonlinear equation

$$\mathcal{P}_k(\widetilde{z}_{kl}^L + \widetilde{z}_{kl}^H) = \mathcal{P}_k\Psi(\widetilde{\mathcal{K}}_{k+l}(\widetilde{z}_{kl}^L + \widetilde{z}_{kl}^H) + f). \tag{2.14}$$

Step 4: Let $\widetilde{z}_{kl} = \widetilde{z}_{kl}^L + \widetilde{z}_{kl}^H$. Set $l \leftarrow l+1$ and go back to Step 2 until $l = m$.

Similar to [**8**], we can estimate the distance between $\widetilde{z}_{km}$ and $\widetilde{z}_{k+m}$. For the convenience of the reader, we provide a sketch of the proof for the error estimate in the following theorem.

Theorem 2.3. *Let z^* be an isolated solution of* (2.4). *Suppose that the assumptions* (H1)–(H4) *hold and* 1 *is not an eigenvalue of the linear*

operator $\Psi'(f+\mathcal{K}z^*)\mathcal{K}$. *Then there exist a positive integer* N *and a positive constant* c *such that, when* $k > N$,

$$\|\widetilde{z}_{km} - \widetilde{z}_{k+m}\|_\infty \le c(k+m)\mu^{-r(k+m)/d}\|z^*\|_{r,\infty}.$$

Proof. Let $\mathcal{L} := \Psi'(f+\mathcal{K}z^*)\mathcal{K}$, and denote

$$\mathcal{R}(u,v) := \Psi(f+\mathcal{K}u) - \Psi(f+\mathcal{K}v) - \mathcal{L}(u-v).$$

Then, for $m \in \mathbf{N}_0$, we have

$$\begin{aligned}(\mathcal{I} - \mathcal{P}_k\mathcal{L})(\widetilde{z}_{km} - \widetilde{z}_{k+m}) &= \mathcal{P}_k\mathcal{R}(\widetilde{z}_{km}, \widetilde{z}_{k+m}) \\ &\quad + (\mathcal{P}_{k+m} - \mathcal{P}_k)[\Psi(\widetilde{\mathcal{K}}_{k+m}\widetilde{z}_{k,m-1} + f) \\ &\quad - \Psi(\widetilde{\mathcal{K}}_{k+m}\widetilde{z}_{k+m} + f)].\end{aligned}$$

Similar to the proof of Lemma 3.2 of [**15**], there exists an $\alpha_{k,m} > 0$ such that

$$\|\widetilde{z}_{km} - \widetilde{z}_{k+m}\|_\infty \le \alpha_{k,m}\|\widetilde{z}_{k,m-1} - \widetilde{z}_{k+m}\|_\infty.$$

Moreover, we have $\alpha_{k,m} \to 0$, $k \to \infty$ uniformly for $m \in \mathbf{N}_0$. Therefore, when k is sufficiently large, we can take $\alpha_{k,m} \le \mu^{-r/d}$ for all $m \in \mathbf{N}_0$. Thus,

$$\begin{aligned}&\|\widetilde{z}_{km} - \widetilde{z}_{k+m}\|_\infty \\ &\le \mu^{-r/d}\left(\|\widetilde{z}_{k,m-1} - \widetilde{z}_{k+m-1}\|_\infty + \|\widetilde{z}_{k+m-1} - z^*\|_\infty + \|z^* - \widetilde{z}_{k+m}\|_\infty\right).\end{aligned} \tag{2.15}$$

Making use of (2.15), the theorem can be proved by induction with the help of Lemma 2.2. □

Combining Lemma 2.2 and Theorem 2.3, a simple application of triangle inequality leads to the following error estimate for $\widetilde{z}_{km}$ to the true solution z^*.

Theorem 2.4. *Let* z^* *be an isolated solution of* (2.4). *Suppose that the assumptions* (H1)–(H4) *hold and* 1 *is not an eigenvalue of the linear*

operator $\Psi'(f + \mathcal{K}z^*)\mathcal{K}$. *Then there exist a positive integer* N *and a positive constant* c *such that, when* $k > N$,

$$\|\widetilde{z}_{km} - z^*\|_\infty \le c(k+m+1)\mu^{-r(k+m)/d}\|z^*\|_{r,\infty}.$$

It is shown by Theorem 2.4 that the output $\widetilde{z}_{km}$ of Algorithm 1 has nearly the same accuracy as $\widetilde{z}_{k+m}$.

3. Truncation strategies and estimation of computational complexity. In this section we propose truncation strategies for the representation matrix of the operator $\mathcal{K}$. We prove that the truncated operators resulting from the strategies satisfy Hypothesis (H4). Moreover, we give the discretization form of the algorithm and estimate the total computational efforts involved in it.

3.1. Two types of truncation strategies. The shape of the representation matrix of an operator is associated closely not only with the properties of the operator, but also with those of the bases to discretize the operator. Therefore, before stating the truncation strategies, we have to describe the bases to discretize equation (2.12). Specifically, we make use of the multiscale bases and collocation functionals which are introduced in [**9**] to establish fast collocation methods for solving linear integral equations of the second kind (see also [**12, 14**] for more information on the bases). We don't intend to describe the concrete construction of the multiscale bases and collocation functionals but only give the notations here. In the next section for numerical experiments, the concrete expressions will be given. For $n \in \mathbf{N}_0$, let $\mathbf{L}_n$ be the space of collocation functionals which corresponds to $\mathbf{X}_n$. We require that the sequence $\{\mathbf{L}_n : n \in \mathbf{N}_0\}$ also be nested; thus, we can make the decomposition $\mathbf{L}_{n+1} = \mathbf{L}_n \oplus \mathbf{V}_{n+1}$ for any $n \in \mathbf{N}_0$. Let $w(0) := \dim(\mathbf{X}_0)$ and $w(i) := \dim(\mathbf{W}_i)$ for $i > 0$. Suppose that

$$\mathbf{X}_0 = \operatorname{span}\{w_{0j} : j \in \mathbf{Z}_{w(0)}\}, \qquad \mathbf{L}_0 = \operatorname{span}\{\ell_{0j} : j \in \mathbf{Z}_{w(0)}\},$$

and, for $i > 0$,

$$\mathbf{W}_i = \operatorname{span}\{w_{ij} : j \in \mathbf{Z}_{w(i)}\}, \qquad \mathbf{V}_i = \operatorname{span}\{\ell_{ij} : j \in \mathbf{Z}_{w(i)}\}.$$

By introducing the index set $J_n := \{(i,j) : i \in \mathbf{Z}_{n+1}, j \in \mathbf{Z}_{w(i)}\}$, we have

$$\mathbf{X}_n = \operatorname{span}\{w_{ij} : (i,j) \in J_n\}, \quad \mathbf{L}_n = \operatorname{span}\{\ell_{ij} : (i,j) \in J_n\},\ n \in \mathbf{N}_0.$$

Define the matrix

$$\mathbf{E}_n := [E_{i'j',ij} : (i',j'),(i,j) \in J_n] \quad \text{with} \quad E_{i'j',ij} := \langle \ell_{i'j'}, w_{ij} \rangle ,$$

and

$$\mathbf{K}_n := [K_{i'j',ij} : (i',j'),(i,j) \in J_n] \quad \text{with} \quad K_{i'j',ij} := \langle \ell_{i'j'}, \mathcal{K} w_{ij} \rangle .$$

Matrices $\mathbf{E}_n$ and $\mathbf{K}_n$ are *representation matrices* with respect to the above multiscale bases and collocation functionals of the operators $\mathcal{I}$ and $\mathcal{K}$, respectively. Moreover, we make use of the index set $J_{kl} := J_{k+l} \backslash J_k$ to define the matrix $\mathbf{E}_{kl} := [E_{i'j',ij} : (i',j'),(i,j) \in J_{kl}]$.

In [**9**] the properties of $\{\ell_{ij} : (i,j) \in J_n\}$ and $\{w_{ij} : (i,j) \in J_n\}$ are described. We list below some of them for possible reference in the proof of the theoretical analyses.

(I) For any $(i,j) \in \mathbf{U} := \{(i,j) : i \in \mathbf{N}_0, j \in \mathbf{Z}_{w(i)}\}$, there are at most $(\mu - 1)r - 1$ number of $w_{ij'}$, $j' \in \mathbf{Z}_{w(i)}$, such that $\operatorname{meas}(w_{ij} \cap w_{ij'}) \neq 0$.

(II) For any $i', i \in \mathbf{N}_0$ with $i \leq i'$, there holds

$$\langle \ell_{i'j'}, w_{ij} \rangle = \delta_{i'i} \delta_{j'j}, \quad j' \in \mathbf{Z}_{w(i')}\ j \in \mathbf{Z}_{w(i)},$$

where $\delta_{i'i}$ is the Kronecker delta.

(III) For any polynomial p of order r and $i \geq 1$,

$$\langle \ell_{ij}, p \rangle = 0, \qquad (w_{ij}, p) = 0, \quad (i,j) \in \mathbf{U},$$

where $(\cdot,\cdot)$ denotes the inner product in $L^2(E)$.

(IV) There exists a positive constant θ_0 such that $\|\ell_{ij}\| \leq \theta_0$ and $\|w_{ij}\|_\infty \leq \theta_0$ for all $(i,j) \in \mathbf{U}$.

(V) There exist positive constants c_- and c_+ such that, for any $i \in \mathbf{N}_0$,

$$c_- \mu^i \leq w(i) \leq c_+ \mu^i, \qquad c_- \mu^{-i} \leq \max_{j \in \mathbf{Z}_{w(i)}} |\operatorname{supp}(w_{ij})| \leq c_+ \mu^{-i}.$$

(VI) For any $v \in \mathbf{X}_n$, we have the unique expansion

$$v = \sum_{(i,j) \in J_n} v_{ij} w_{ij}.$$

There exist positive constants θ_1 and θ_2 such that

$$\theta_1\|\mathbf{v}\|_\infty \le \|v\|_\infty \le \theta_2(n+1)\|\mathbf{E}_n\mathbf{v}\|_\infty,$$

in which $\mathbf{E}_n := [\langle \ell_{i'j'}, w_{ij}\rangle : (i',j'),(i,j) \in J_n]$.

The truncated operator $\widetilde{\mathcal{K}}_n$ is defined by its representation matrix, which is denoted by $\widetilde{\mathbf{K}}_n$, and $\widetilde{\mathbf{K}}_n$ is obtained by truncating the representation matrix $\mathbf{K}_n$ of the operator $\mathcal{K}_n$. In the following we describe the truncation strategies for two cases, in which the kernel K is smooth and weakly singular, respectively.

1. K **is smooth.** There exist a positive constant Λ and a sufficiently large positive integer r, such that

$$|D_s^\alpha D_t^\beta K(s,t)| \le \Lambda \tag{3.1}$$

for any $s,t \in E$ and $|\alpha| \le r$, $|\beta| \le r$. The truncation strategy is as follows.

(TM) For $(i',j'),(i,j) \in J_n$, define

$$\widetilde{K}_{i'j',ij} = \begin{cases} K_{i'j',ij} & i'+i \le n, \\ 0 & \text{otherwise.} \end{cases}$$

Then the truncated matrix is defined by

$$\widetilde{\mathbf{K}}_n := [\widetilde{K}_{i'j',ij} : (i',j'),(i,j) \in J_n].$$

The above strategy has been proposed in [**17**] and used in [**6**] for one-dimensional equations. The following lemma provides estimates on $\mathcal{K}_n - \widetilde{\mathcal{K}}_n$. Since the one-dimensional case was proven in [**6**], we don't intend to provide all details in the proof of the lemma.

Lemma 3.1. *Suppose that the kernel K is smooth in the sense of* (3.1) *and the representation matrix of $\mathcal{K}$ is truncated with strategy* (TM). *For a matrix A, we denote by $\mathcal{N}(A)$ the number of nonzero entries of A. Then $\mathcal{N}(\widetilde{\mathbf{K}}_n) = \mathcal{O}(n\mu^n)$. For any $v \in \mathbf{X}_n$, the following holds*

$$\|(\mathcal{K}_n - \widetilde{\mathcal{K}}_n)v\|_\infty \le c(n+1)^2\mu^{-rn/d}\|v\|_\infty \tag{3.2}$$

for all $n \in \mathbf{N}_0$. *Moreover, if* $z^* \in W^{r,\infty}(E)$ *and* v *satisfies* $\|v - z^*\|_\infty \leq cn\mu^{-r(n-1)/d}\|z^*\|_{r,\infty}$, *then*

$$\|(\mathcal{K}_n - \widetilde{\mathcal{K}}_n)v\|_\infty \leq c(n+1)\mu^{-rn/d}\|z^*\|_{r,\infty}. \tag{3.3}$$

Proof. The estimate for $\mathcal{N}(\widetilde{\mathbf{K}}_n)$ has been given in [**17**]. For $i', i \in \mathbf{Z}_{n+1}$, we define

$$\mathbf{K}_{i'i} := [K_{i'j',ij} : j' \in \mathbf{Z}_{w(i')}, j \in \mathbf{Z}_{w(i)}]$$

and

$$\widetilde{\mathbf{K}}_{i'i} := [\widetilde{K}_{i'j',ij} : j' \in \mathbf{Z}_{w(i'j)} \in \mathbf{Z}_{w(i)}.$$

According to the truncation strategy, when $i' + i \leq n$, $\mathbf{K}_{i'i} = \widetilde{\mathbf{K}}_{i'i}$, and when $i' + i > n$, $\mathbf{K}_{i'i} - \widetilde{\mathbf{K}}_{i'i} = \mathbf{K}_{i'i}$. It follows from property (III) and (3.1) that, for $j' \in \mathbf{Z}_{w(i')}$ and $j \in \mathbf{Z}_{w(i)}$,

$$|K_{i'j',ij}| \leq c\mu^{-r(i'+i)/d}\Lambda|S_{ij}|.$$

We then make use of property (I) to obtain

$$\sum_{j \in \mathbf{Z}_{w(i)}} |K_{i'j',ij}| \leq c\mu^{-r(i'+i)}\Lambda \leq c\mu^{-rn/d}\Lambda,$$

when $i' + i > n$. Therefore, for $i', i \in \mathbf{Z}_n$, $\|\mathbf{K}_{i'i} - \widetilde{\mathbf{K}}_{i'i}\|_\infty \leq c\mu^{-rn/d}\Lambda$.

Let $\mathbf{h} := \mathbf{E}_n^{-1}(\mathbf{K}_n - \widetilde{\mathbf{K}}_n)\mathbf{v}$. We expand $(\mathcal{K}_n - \widetilde{\mathcal{K}}_n)v$ into

$$(\mathcal{K}_n - \widetilde{\mathcal{K}}_n)v = \sum_{(i,j) \in J_n} h_{ij}w_{ij}.$$

Then property (VI) gives

$$\|(\mathcal{K}_n - \widetilde{\mathcal{K}}_n)v\|_\infty \leq \theta_2(n+1)\|(\mathbf{K}_n - \widetilde{\mathbf{K}}_n)\mathbf{v}\|_\infty. \tag{3.4}$$

It is left to estimate $\|(\mathbf{K}_n - \widetilde{\mathbf{K}}_n)\mathbf{v}\|_\infty$. For the first case, note that property (VI) leads to $\|\mathbf{v}\|_\infty \leq \theta_1^{-1}\|v\|_\infty$. On the other hand, there exists a positive constant c such that, for all $n \in \mathbf{N}_0$,

$$\|\mathbf{K}_n - \widetilde{\mathbf{K}}_n\|_\infty \leq \max_{i' \in \mathbf{Z}_{n+1}} \sum_{i \in \mathbf{Z}_{n+1}} \|\mathbf{K}_{i'i} - \widetilde{\mathbf{K}}_{i'i}\|_\infty \leq c(n+1)\mu^{-rn/d}.$$

For the second case, the proof is very similar to Lemma 4.2 of [**6**], the idea of which originates from Lemma 4.2 in [**9**]. □

2. K **is weakly singular.** For $s, t \in E$ with $s \neq t$, the kernel has continuous partial derivatives $D_s^\alpha D_t^\beta K(s,t)$ for $|\alpha| \leq r$ and $|\beta| \leq r$, and there exist positive constants θ and $\sigma < d$ such that, for $|\alpha| = |\beta| = r$,

$$|D_s^\alpha D_t^\beta K(s,t)| \leq \theta |s-t|^{-(\sigma+2r)}.$$

For $(i,j) \in J_n$, let $S_{ij} := \operatorname{supp}(w_{ij})$, and define $d_i := \max\{\operatorname{diam}(S_{ij}) : j \in \mathbf{Z}_{w(i)}\}$ for $i \in \mathbf{N}_0$. The truncation strategy for this case is

(TS) For $(i',j'), (i,j) \in J_n$, define

$$\widetilde{K}_{i'j',ij} = \begin{cases} K_{i'j',ij} & \operatorname{dist}(S_{i'j'}, S_{ij}) \leq \varepsilon_{i'i}^n, \\ 0 & \text{otherwise}, \end{cases}$$

in which the *truncation parameters* $\varepsilon_{i'i}^n$ are identified by

$$\varepsilon_{i'i}^n := \max\{a\mu^{[-n+b(n-i)+b'(n-i')]/d}, \rho(d_i + d_{i'})\} \tag{3.5}$$

for some positive constants b', b, $a > 0$ and $\rho > 1$. Then the truncated matrix is defined by

$$\widetilde{\mathbf{K}}_n := [\widetilde{K}_{i'j',ij} : (i',j'), (i,j) \in J_n].$$

The strategy is established and analyzed in [**9**] and used in [**6, 7, 10, 11, 14**]. According to the analysis in [**9**], the following holds:

$$\|\mathbf{K}_{i'i} - \widetilde{\mathbf{K}}_{i'i}\|_\infty \leq c(\varepsilon_{i'i}^n)^{-(2r-\sigma')}\mu^{-r(i'+i)/d}, \quad i', i \in \mathbf{Z}_{n+1}. \tag{3.6}$$

Following a similar theme as the proof of Lemma 3.1, we conclude

Lemma 3.2. *Let* $\sigma' \in (0,d)$, $\eta := 2r - \sigma'$, *and set* $b = 1$, $b' \in (r/\eta, 1)$ *in* (3.5). *Then* $\mathcal{N}(\widetilde{\mathbf{K}}_n) = \mathcal{O}(n\mu^n)$. *For any* $v \in \mathbf{X}_n$ *and* $n \in \mathbf{N}_0$,

$$\|(\mathcal{K}_n - \widetilde{\mathcal{K}}_n)v\|_\infty \leq c(n+1)\mu^{-\sigma' n/d}\|v\|_\infty. \tag{3.7}$$

Moreover, if $z^* \in W^{r,\infty}(E)$ *and* v *satisfies* $\|v - z^*\|_\infty \leq cn\mu^{-r(n-1)/d}\|z^*\|_{r,\infty}$, *then*

$$\|(\mathcal{K}_n - \widetilde{\mathcal{K}}_n)v\|_\infty \leq c(n+1)\mu^{-rn/d}\|z^*\|_{r,\infty}. \tag{3.8}$$

For the meaning and ranges of the parameters in Lemma 3.2, please refer to [**6, 9**].

The results of Lemmas 3.1 and 3.2 confirm that Hypothesis (H4) is satisfied if we use strategy (TM) to truncate the representation matrices of smooth kernels and strategy (TS) to truncate those of weakly singular kernels.

With the truncated matrix $\widetilde{\mathbf{K}}_n$, the truncated operator $\widetilde{\mathcal{K}}_n$ is defined as the linear operator from $\mathbf{X}_n$ onto $\mathbf{X}_n$ satisfying

$$\widetilde{K}_{i'j',ij} = \left\langle \ell_{i'j'}, \widetilde{\mathcal{K}}_n w_{ij} \right\rangle, \quad (i',j'),(i,j) \in J_n.$$

It is easily seen that the operator is uniquely determined by the above conditions.

3.2. Discrete form of multilevel augmentation methods equipped with matrix truncation. We have shown above how to establish the representation matrix $\widetilde{\mathbf{K}}_n$ of the truncated integral operator $\widetilde{\mathcal{K}}_n$. However, $\widetilde{\mathbf{K}}_n$ cannot be used in the algorithm. To see this, we note that, with respect to the multiscale bases and collocation functionals, (2.13) is equivalent to the nonlinear system

$$\langle \ell_{i'j'}, \widetilde{z}_{kl}^H \rangle = \left\langle \ell_{i'j'}, \Psi(\widetilde{\mathcal{K}}_{k+l}\widetilde{z}_{k,l-1} + f) \right\rangle, \quad (i',j') \in J_{kl}. \tag{3.9}$$

Suppose that

$$\widetilde{z}_{k,l-1} = \sum_{(i,j)\in J_{k+l-1}} (\widetilde{\mathbf{z}}_{k,l-1})_{ij} w_{ij},$$

and define the *representation vector* $\widetilde{\mathbf{z}}_{k,l-1} := [(\widetilde{\mathbf{z}}_{k,l-1})_{ij} : (i,j) \in J_{k+l-1}]^T$ and the matrix $\widetilde{\mathbf{K}}_{k,l-1}^H := [\widetilde{K}_{i'j',ij} : (i',j') \in J_{kl}, (i,j) \in J_{k+l-1}]$. Then

$$\widetilde{\mathbf{K}}_{k,l-1}^H \widetilde{\mathbf{z}}_{k,l-1} = \left[\left\langle \ell_{i'j'}, \widetilde{\mathcal{K}}_{k+l}\widetilde{z}_{k,l-1} \right\rangle : (i',j') \in J_{kl} \right]^T,$$

i.e., making use of the matrix $\widetilde{\mathbf{K}}_n$, we have the values of the functionals $\ell_{i'j'}$ applied on $\widetilde{\mathcal{K}}_{k+l}\widetilde{z}_{k,l-1}$. Note that, in general, $\ell_{i'j'}$ is not a single point evaluation, but a finite combination of point evaluations (see [**9**] for the details). On the other hand, what we need in the right

hand side of (3.9) is not the combination of point evaluations, but the value of single point evaluation. To observe this, let ℓ be a functional, which is a combination of M point evaluations. We express it by $\ell := \sum_{j\in\mathbf{Z}_M} \alpha_j \delta_{t_j}$. For $j \in \mathbf{Z}_M$, let

$$g_j := (\widetilde{\mathcal{K}}_{k+l}\widetilde{z}_{k,l-1})(t_j) + f(t_j).$$

Then

$$\left\langle \ell, \Psi(\widetilde{\mathcal{K}}_{k+l}\widetilde{z}_{k,l-1} + f)\right\rangle = \sum_{j\in\mathbf{Z}_M} \alpha_j \psi(t_j, g_j).$$

Based on the above observations, we use the following computing trick. For any functional $\ell = \sum_{j\in\mathbf{Z}_M} \alpha_j \delta_{t_j}$, we denote by $\boldsymbol{\alpha}(\ell) := [\alpha_j : j \in \mathbf{Z}_M]$ the vector of combination coefficients and by $\mathbf{t}(\ell) := [t_j : j \in \mathbf{Z}_M]$ the vector of the points emerging in ℓ. For any $(i', j'), (i, j) \in J_n$, define the vector

$$\overline{K}_{i'j',ij} := \begin{cases} (\mathcal{K}w_{ij})(\mathbf{t}(\ell_{i'j'})) & \widetilde{K}_{i'j',ij} \neq 0, \\ \mathbf{0} & \widetilde{K}_{i'j',ij} = 0. \end{cases}$$

We remark that each $\overline{K}_{i'j',ij}$ is a vector since $\mathbf{t}(\ell_{i'j'})$ is a vector. For $\mathbf{s} := [s_j : j \in \mathbf{Z}_M]$ and $\mathbf{t} := [t_j : j \in \mathbf{Z}_M]$, we use the notation $\psi(\mathbf{s}, \mathbf{t})$ to denote the vector $[\psi(s_j, t_j) : j \in \mathbf{Z}_M]$. Compute the vector

$$\mathbf{g} := \sum_{(i,j)\in J_{k+l-1}} \overline{K}_{i'j',ij}(\widetilde{\mathbf{z}}_{k,l-1})_{ij} + f(\mathbf{t}(\ell_{i'j'})).$$

Then

$$\left\langle \ell_{i'j'}, \Psi(\widetilde{\mathcal{K}}_{k+l}\widetilde{z}_{k,l-1} + f)\right\rangle = \left(\alpha(\ell_{i'j'}), \psi(\mathbf{t}(\ell_{i'j'}), \mathbf{g})\right),$$

in which $(\cdot, \cdot)$ denotes the ℓ^2 inner product. Therefore, we have the following discrete form of multilevel augmentation method.

Algorithm 2: Multilevel augmentation method equipped with truncation strategy: the discrete form. Let k be a fixed positive integer. Given $m \in \mathbf{N}_0$, the algorithm gives the representation vector $\widetilde{\mathbf{z}}_{km}$ of $\widetilde{z}_{km}$. Before carrying out the following steps, we assume that the values of $\overline{K}_{i'j',ij},\ (i', j'), (i, j) \in J_{k+m}$ have been obtained.

Step 1: For $(i', j') \in J_k$, define the vector

$$\overline{f}_{i'j'}(\widetilde{\mathbf{z}}_k) := \sum_{(i,j)\in J_k} \overline{K}_{i'j',ij}(\widetilde{\mathbf{z}}_k)_{ij} + f(\mathbf{t}(\ell_{i'j'})).$$

Then let

$$\mathbf{f}_k(\widetilde{\mathbf{z}}_k) := \left[\left(\boldsymbol{\alpha}(\ell_{i'j'}), \psi(\mathbf{t}(\ell_{i'j'}), \overline{f}_{i'j'}(\widetilde{\mathbf{z}}_k))\right) : (i', j') \in J_k\right]^T.$$

Solve $\widetilde{\mathbf{z}}_k := [(\widetilde{\mathbf{z}}_k)_{ij} : (i, j) \in J_k]$ from the nonlinear system

$$\mathbf{E}_k\widetilde{\mathbf{z}}_k - \mathbf{f}_k(\widetilde{\mathbf{z}}_k) = 0.$$

Let $\widetilde{\mathbf{z}}_{k0} := \widetilde{\mathbf{z}}_k$, and set $l := 1$.

Step 2: For $(i', j') \in J_{kl}$, compute

$$\overline{f}_{i'j'} := \sum_{(i,j)\in J_{k+l-1}} \overline{K}_{i'j',ij}(\widetilde{\mathbf{z}}_{k,l-1})_{ij} + f(\mathbf{t}(\ell_{i'j'})).$$

Then we define

$$\mathbf{f}_{kl} := \left[\left(\boldsymbol{\alpha}(\ell_{i'j'}), \psi(\mathbf{t}(\ell_{i'j'}), \overline{f}_{i'j'})\right) : (i', j') \in J_{kl}\right]^T.$$

Solve the linear system

$$\mathbf{E}_{kl}\widetilde{\mathbf{z}}^H_{kl} = \mathbf{f}_{kl} \tag{3.10}$$

to obtain the vector $\widetilde{\mathbf{z}}^H_{kl} := [(\widetilde{\mathbf{z}}_{kl})_{ij} : (i, j) \in J_{kl}]^T$.

Step 3: For $(i', j') \in J_k$, compute

$$\widehat{f}_{i'j'}(\widetilde{\mathbf{z}}_{kl}) := \sum_{(i,j)\in J_{k+l}} \overline{K}_{i'j',ij}(\widetilde{\mathbf{z}}_{kl})_{ij} + f(\mathbf{t}(\ell_{i'j'})).$$

Then we define

$$\widehat{\mathbf{f}}_k(\widetilde{\mathbf{z}}_{kl}) := \left[\left(\boldsymbol{\alpha}(\ell_{i'j'}), \psi(\mathbf{t}(\ell_{i'j'}), \widehat{f}_{i'j'}(\widetilde{\mathbf{z}}_{kl}))\right) : (i', j') \in J_k\right]^T.$$

Let $\mathbf{E}^{LH}_{kl} := [E_{i'j',ij} : (i', j') \in J_k, (i, j) \in J_{kl}]$. Solve the nonlinear system

$$\mathbf{E}_k\widetilde{\mathbf{z}}^L_{kl} + \mathbf{E}^{LH}_{kl}\widetilde{\mathbf{z}}^H_{kl} = \widehat{\mathbf{f}}_k(\widetilde{\mathbf{z}}_{kl}) \tag{3.11}$$

to obtain $\widetilde{\mathbf{z}}_{kl}^L := [(\widetilde{\mathbf{z}}_{kl})_{ij} : (i,j) \in J_k]^T$. Note that, in the above nonlinear system, $(\widetilde{\mathbf{z}}_{kl})_{ij},\ (i,j) \in J_{kl}$ are known values.

Step 4: Concatenate $\widetilde{\mathbf{z}}_{kl}^L$ and $\widetilde{\mathbf{z}}_{kl}^H$ to form the vector $\widetilde{\mathbf{z}}_{kl}$. Set $l \leftarrow l+1$ and go back to Step 2 until $l = m$.

According to the description of the algorithm given above, we can estimate the computational efforts. Since we do not state how to numerically evaluate the integrals involved, we adopt the assumption that each integral is approximately computed with a constant number of multiplications and functional evaluations.

Theorem 3.3. *The total numbers of multiplications and functional evaluations in implementing Algorithm* 2 *are both of* $\mathcal{O}((k+m)\mu^{k+m})$.

Proof. Since the initial level is fixed and Step 1 is only carried out once, the numbers of multiplications and functional evaluations are both constants. For $l \le m$ and $j = 2, 3$, let $N_{l,j,1}$ and $N_{l,j,2}$ denote the number of multiplications and functional evaluations in step j, respectively. Throughout the proof, we use the fact that the numbers of points involved in the collocation functionals $\ell_{ij},\ (i,j) \in J_{k+m}$ are uniformly bounded.

In Step 2, the number of multiplications for computing $\overline{f}_{i'j'},\ (i',j') \in J_{kl}$ is $\mathcal{O}(\mathcal{N}(\widetilde{\mathbf{K}}_{k,l-1}^H))$, and that of functional evaluations is $\mathcal{O}(|J_{kl}|)$, where $|J_{kl}|$ is the cardinality of the index set J_{kl}. The number of multiplications and functional evaluations for generating $\mathbf{f}_{kl}$ are both $\mathcal{O}(|J_{kl}|)$. The number of multiplications for solving the linear system (3.10) is $\mathcal{N}(\mathbf{E}_{kl})$. Note that $\mathcal{N}(\widetilde{\mathbf{K}}_{k,l-1}^H) \le \mathcal{N}(\widetilde{\mathbf{K}}_{k+l-1}) = \mathcal{O}((k+l-1)\mu^{k+l-1})$, $|J_{kl}| < \mathcal{O}(\mu^{k+l})$, and $\mathcal{N}(\mathbf{E}_{kl}) < \mathcal{N}(\mathbf{E}_{k+l}) = \mathcal{O}((k+l)\mu^{k+l})$. See [**9**] for the estimate of the number of nonzero entries of the matrix $\mathbf{E}_n$. Therefore, we have

$$N_{l,2,1} = \mathcal{O}((k+l+\mu-1)\mu^{k+l-1}), \quad N_{l,2,2} = \mathcal{O}(\mu^{k+l}). \tag{3.12}$$

In Step 3 we use Newton iteration method to solve the nonlinear system (3.11). Given $\widetilde{\mathbf{z}}_{kl}$, computing $\widehat{f}_{i'j'}(\widetilde{\mathbf{z}}_{kl}),\ (i',j') \in J_k$ and then forming the vector $\widehat{\mathbf{f}}_k(\widetilde{\mathbf{z}}_{kl})$ requires $\mathcal{O}((k+l)\mu^{k+l})$ number of multiplications and $\mathcal{O}(\mu^k)$ number of functional evaluations. Since

$\mathcal{N}(\mathbf{E}_k) + \mathcal{N}(\mathbf{E}_{kl}^{LH}) \le \mathcal{N}(\mathbf{E}_{k+l}) = \mathcal{O}((k+l)\mu^{k+l})$, computing

$$\mathbf{E}_k \widetilde{\mathbf{z}}_{kl}^{L} + \mathbf{E}_{kl}^{LH} \widetilde{\mathbf{z}}_{kl}^{H} - \widehat{\mathbf{f}}_k(\widetilde{\mathbf{z}}_{kl})$$

requires $\mathcal{O}((k+l)\mu^{k+l})$ number of multiplications and $\mathcal{O}(\mu^k)$ number of functional evaluations. On the other hand, establishing the Jacobian matrix also requires $\mathcal{O}((k+l)\mu^{k+l})$ number of multiplications and $\mathcal{O}(\mu^k)$ number of functional evaluations. The computational effort for one iteration can be estimated by adding them up. Assuming the number of iterations is uniformly bounded, we have

$$N_{l,3,1} = \mathcal{O}((k+l)\mu^{k+l}), \qquad N_{l,3,2} = \mathcal{O}(\mu^k). \tag{3.13}$$

A direct calculation shows that

$$\sum_{l=1}^{m} N_{l,2,1} = \mathcal{O}((k+m)\mu^{k+m}), \qquad \sum_{l=1}^{m} N_{l,3,1} = \mathcal{O}(\mu^{k+m}).$$

Moreover, before running the algorithm, we have to evaluate $\overline{K}_{i'j',ij}$, $(i',j'),(i,j) \in J_{k+m}$, which requires $\mathcal{O}((k+m)\mu^{k+m})$ number of multiplications and $\mathcal{O}((k+m)\mu^{k+m})$ number of functional evaluations. The result of the theorem is concluded by summing up the above estimates. □

4. Numerical experiments. In this section we use four experiments to compare the efficiency and accuracy of the fast algorithm proposed in the paper with that from [**8**], the multilevel augmentation methods without matrix compression. The four numerical examples of equation (2.4) include two one-dimensional equations and two two-dimensional equations. We also compare our algorithm with that in [**6**], which applies multilevel augmentation methods for solving the Hammerstein equation, but not its reformulated form. The computer to run the programs is equipped with a 2.66GHz CPU and 4G memory.

4.1. One-dimensional equations. Let $E := [0,1]$. In the following two experiments, we set $\mu = 2$ and let $\mathbf{X}_n$ be the space of piecewise linear polynomials with the knots at $j/2^n$, $j = 1, 2, \ldots, 2^n - 1$.

Obviously $\dim(\mathbf{X}_n) = 2^{n+1}$. The basis functions of $\mathbf{X}_0$ and $\mathbf{W}_1$ are defined by

$$w_{00}(t) := -3t + 2, \qquad w_{01}(t) := 3t - 1, \quad t \in [0, 1]$$

and

$$w_{10}(t) := \begin{cases} 1 - (9/2)t & t \in [0, 1/2), \\ (3/2) - 1 & t \in [1/2, 1], \end{cases}$$
$$w_{11}(t) := \begin{cases} (1/2) - (3/2)t & t \in [0, 1/2), \\ (9/2) - (7/2) & t \in [1/2, 1]. \end{cases}$$

The corresponding collocation functionals are

$$\ell_{00} = \delta_{1/3}, \qquad \ell_{01} = \delta_{2/3},$$
$$\ell_{10} = \delta_{1/6} - \frac{3}{2}\delta_{1/3} + \frac{1}{2}\delta_{2/3},$$
$$\ell_{11} = \frac{1}{2}\delta_{1/3} - \frac{3}{2}\delta_{2/3,} + \delta_{5/6}.$$

The basis functions and collocation functionals for $i > 1$ are constructed recursively from those for $i = 1$ through the family of contractive mappings $\Phi := \{\phi_0, \phi_1\}$ with

$$\phi_0(t) := \frac{t}{2}, \qquad \phi_1(t) := \frac{t+1}{2}, \quad t \in [0, 1].$$

For example, the basis functions $w_{2j},\ j \in \mathbf{Z}_4$ can be obtained by

$$w_{20}(t) := w_{10}(\phi_0^{-1}(t)), \qquad w_{21}(t) := w_{11}(\phi_0^{-1}(t)),$$
$$w_{22}(t) := w_{10}(\phi_1^{-1}(t)), \qquad w_{23}(t) := w_{11}(\phi_1^{-1}(t)).$$

It is easily seen that the supports of w_{20} and w_{21} are contained in $[0, 1/2]$, while those of w_{22} and w_{23} are contained in $[1/2, 1]$. The functions $w_{3j},\ j \in \mathbf{Z}_8$ can be constructed from $w_{2j},\ j \in \mathbf{Z}_4$ in the above way. The collocation functionals can also be constructed in the same way. The general construction with the recursive process is established in [**9**], and the above basis functions and collocation functionals are described in [**13**] and have been used in numerical experiments in [**6, 11, 15, 16**]. To implement the fast multilevel augmentation methods, we set the initial level at $k = 4$.

1. Smooth kernel. Let the kernel $K(s,t) := \sin(\pi(s+t))$, and define $\psi(s,u) := u^2$. Let

$$f(t) := \sin(\pi t) - \frac{4}{3\pi}\cos(\pi t),$$

so that $z(t) = \sin^2(\pi t)$ is an isolated solution of (2.4).

Since the kernel K is smooth, we apply strategy (TM). The experiment data are listed in Table 1. In the table the second column lists the dimensions of the discretization subspaces, and the third and fourth columns give the errors of numerical solutions resulting from fast multilevel augmentation methods without and with truncation, respectively. It is seen that for the same m, $\|z^* - \widetilde{z}_{4,m}\|_\infty$ is only a little bit bigger than $\|z^* - z_{4,m}\|_\infty$, i.e., truncation hardly brings any new error to numerical solutions, and $z_{4,m}$ and $\widetilde{z}_{4,m}$ converge at nearly the same rate. The column with title, "Comp. Rate," lists the compression rate of the truncation, the value which is defined by the ratio $\mathcal{N}(\widetilde{\mathbf{K}}_{4+m})/\mathcal{N}(\mathbf{K}_{4+m})$. According to Lemma 3.1,

$$\frac{\mathcal{N}(\widetilde{\mathbf{K}}_{4+m})}{\mathcal{N}(\mathbf{K}_{4+m})} = \mathcal{O}((4+m)\mu^{-(4+m)}).$$

Therefore, when m increases by 1, the above ratio decreases to about half since $\mu = 2$. In the last two columns, "T_M" and "$\widetilde{T}_M$" are total running time of the multilevel augmentation methods without and with truncation, respectively. It is seen that T_M grows much faster than $\widetilde{T}_M$. For $m = 9$, T_M is about 50 times that of $\widetilde{T}_M$. Correspondingly, the compression rate of $\widetilde{\mathbf{K}}_{13}$ is 0.0017, i.e., less than 1 percent of the matrix entries needed to evaluate, which in part explains why the running time is shortened so much. We also observe that, when the discretization dimension doubles, $\widetilde{T}_M$ is a little bit more than doubled. This is consistent with the result of Theorem 3.3.

2. Singular kernel. Let the kernel

$$K(s,t) := \log\left(\frac{1}{16}|\cos(\pi s) - \cos(\pi t)|\right)$$

TABLE 1. Numerical results for one-dimensional equation with smooth kernel.

m	d_{4+m}	$\|z^* - z_{4,m}\|_\infty$	$\|z^* - \widetilde{z}_{4,m}\|_\infty$	Comp. Rate	T_M	$\widetilde{T}_M$
0	32	9.113e-3	9.113e-3	0.3125	0.06	0.03
1	64	2.216e-3	2.297e-3	0.1875	0.13	0.07
2	128	5.642e-4	5.708e-4	0.1094	0.32	0.14
3	256	1.400e-4	1.427e-4	0.0625	0.74	0.26
4	512	3.470e-5	3.578e-5	0.0352	1.96	0.50
5	1024	8.616e-6	8.922e-6	0.0195	5.03	1.00
6	2048	2.102e-6	2.241e-6	0.0107	15.0	2.00
7	4096	5.547e-7	5.601e-7	0.0059	47.2	3.81
8	8192	1.373e-7	1.398e-7	0.0032	187	7.89
9	16384	3.478e-8	3.486e-8	0.0017	749	15.9

TABLE 2. Numerical results for one-dimensional equation with singular kernel.

m	d_{4+m}	$\|z^* - z_{4,m}\|_\infty$	$\|z^* - \widetilde{z}_{4,m}\|_\infty$	Comp. Rate	T_M	$\widetilde{T}_M$
0	32	4.111e-3	4.111e-3	0.8910	0.03	0.03
1	64	1.043e-3	1.043e-3	0.7148	0.07	0.07
2	128	2.595e-4	2.595e-4	0.4863	0.21	0.14
3	256	6.429e-5	6.429e-5	0.3137	0.63	0.36
4	512	1.612e-5	1.612e-5	0.1938	2.16	0.94
5	1024	4.015e-6	4.014e-6	0.1159	7.80	2.46
6	2048	1.008e-6	1.008e-6	0.0676	29.4	6.39
7	4096	2.512e-7	2.510e-7	0.0387	112	18.0
8	8192	6.234e-8	6.564e-8	0.0218	450	39.4
9	16384	1.797e-8	1.838e-8	0.0121	1812	81.5

and $\psi(s,u) := 2u^2 - 1$. Moreover, let

$$f(t) := \cos\left(\frac{\pi}{2}(\frac{1}{2} - t)\right) + \frac{1}{16\pi}[2 - (1 - \cos(\pi t))\log(1 - \cos(\pi t)) - (1 + \cos(\pi t))\log(1 + \cos(\pi t))].$$

Then $z(t) = \cos^2((\pi/2)((1/2) - t)) - 1$ is an isolated solution of (2.4).

Note that, in this example, the kernel is weakly singular. Thus, strategy (TS) is applied. The numerical results are reported in Table 2. The

data in Table 2 give us a similar conclusion with the case of a smooth kernel. The solutions $z_{4,m}$ and $\widetilde{z}_{4,m}$ have nearly the same accuracy. The compression rate decreases to nearly half when m increases by 1. $\widetilde{T}_M$ is a little more than doubled when the discretization dimension is doubled, and much less than T_M for large m. As Lemmas 3.1 and 3.2 claim, $\mathcal{N}(\widetilde{\mathbf{K}}_n)$ has the same growth rate for smooth and singular cases. However, it is worth noting that, for the same discretization dimension, the compression rates for representation matrices of singular kernel are much more than those of smooth kernel. For $m = 9$, the compression rate is larger than 1 percent. Correspondingly, T_M is about 22.5 times that of $\widetilde{T}_M$.

4.2. Two-dimensional equations. In the following two numerical experiments we consider solving two-dimensional equations. To this end, we let the domain $E := \{(x, y) : 0 \le x \le y \le 1\}$, which is a triangle. To discretize (2.4), we set $\mu = 4$ and define the family of contractive mappings $\Phi := \{\phi_e : e \in \mathbf{Z}_4\}$ with

$$\phi_0(x,y) := \left(\frac{x}{2}, \frac{y}{2}\right), \qquad \phi_1(x,y) := \left(\frac{x}{2}, \frac{y+1}{2}\right),$$
$$\phi_2(x,y) := \left(\frac{1-x}{2}, \frac{1-y}{2}\right), \qquad \phi_3(x,y) := \left(\frac{1+x}{2}, \frac{y+1}{2}\right).$$

Denote $\mathbf{Z}_4^n := \mathbf{Z}_4 \times \mathbf{Z}_4 \times \cdots \times \mathbf{Z}_4$ (n times) and, for any $\mathbf{e} := (e_0, e_1, \ldots, e_{n-1}) \in \mathbf{Z}_4^n$, we define the composite mapping $\phi_{\mathbf{e}} := \phi_{e_0} \circ \cdots \circ \phi_{e_{n-1}}$. Then

$$E_n := \{\phi_{\mathbf{e}}(E) : \mathbf{e} \in \mathbf{Z}_4^n\}$$

is a uniform partition of E. Let $\mathbf{X}_n$ be the space of piecewise linear polynomials with respect to the partition E_n. It is not difficult to obtain that $\dim(\mathbf{X}_n) = 3 \times 4^n$. For $e \in \mathbf{Z}_4$, denote $S_e := \phi_e(E)$. The basis functions of $\mathbf{X}_0$ are chosen as

$$w_{00}(x,y) := -3x + 2y,$$
$$w_{01}(x,y) := x - 3y + 2,$$
$$w_{02}(x,y) := 2x + y - 1,$$
$$(x,y) \in E.$$

The corresponding collocation functionals are

$$\ell_{00} = \delta_{((2/7),(3/7))}, \qquad \ell_{01} = \delta_{((1/7),(5/7))}, \qquad \ell_{02} = \delta_{((4/7),(6/7))}.$$

Moreover, the basis functions for $\mathbf{W}_1$ are chosen to be

$$w_{10}(x,y) := \begin{cases} -(11/8) - (15/8)x + (41/8)y & (x,y) \in S_0, \\ (5/8) + (1/8)x - (7/8)y & (x,y) \in S \setminus S_0, \end{cases}$$

$$w_{11}(x,y) := \begin{cases} 1 - (15/4)x - (7/8)y & (x,y) \in S_0, \\ -1 + (1/4)x + (9/8)y, & (x,y) \in S \setminus S_0, \end{cases}$$

$$w_{12}(x,y) := \begin{cases} (9/8) + (15/8)x - (29/8)y, & (x,y) \in S_0, \\ -(15/8) - (1/8)x + (19/8)y, & (x,y) \in S \setminus S_0, \end{cases}$$

$$w_{13}(x,y) := \begin{cases} -(15/8) - (41/8)x + (13/4)y, & (x,y) \in S_1, \\ (1/8) + (7/8)x - (3/4)y, & (x,y) \in S \setminus S_1, \end{cases}$$

$$w_{14}(x,y) := \begin{cases} (29/8) + (7/8)x - (37/8)y, & (x,y) \in S_1, \\ -(3/8) - (9/8)x + (11/8)y, & (x,y) \in S \setminus S_1, \end{cases}$$

$$w_{15}(x,y) := \begin{cases} -(5/8) - (29/8)x + (7/4)y, & (x,y) \in S_1, \\ (3/8) + (19/8)x - (9/4)y, & (x,y) \in S \setminus S_1, \end{cases}$$

$$w_{16}(x,y) := \begin{cases} (15/4) - (13/8)x - (15/8)y, & (x,y) \in S_3, \\ -(1/4) + (3/4)x + (1/8)y, & (x,y) \in S \setminus S_3, \end{cases}$$

$$w_{17}(x,y) := \begin{cases} -(1/8) - (37/8)x + (15/4)y, & (x,y) \in S_3, \\ -(1/8) + (11/8)x - (1/4)y, & (x,y) \in S \setminus S_3, \end{cases}$$

$$w_{18}(x,y) := \begin{cases} -(5/2) + (7/4)x + (15/8)y, & (x,y) \in S_3, \\ (1/2) - (9/4)x - (1/8)y, & (x,y) \in S \setminus S_3, \end{cases}$$

and the corresponding collocation functionals are chosen as

$$\ell_{10} = \delta_{((1/14),(5/14))} - \delta_{((1/7),(3/14))} + \delta_{((2/7),(3/7))} - \delta_{((3/14),(4/7))},$$
$$\ell_{11} = \delta_{((1/14),(5/14))} - \delta_{((2/7)),(3/7))} + \delta_{((3/7),(9/14))} - \delta_{((3/14),(4/7))},$$
$$\ell_{12} = \delta_{((5/14),(11/14))} - \delta_{((3/7),(9/14))} + \delta_{((2/7),(3/7))} - \delta_{((3/14),(4/7))},$$
$$\ell_{13} = \delta_{((1/14),(6/7))} - \delta_{((1/7),(5/7))} + \delta_{((5/14),(11/14))} - \delta_{((2/7),(13/14))},$$
$$\ell_{14} = \delta_{((1/7),(5/7))} - \delta_{((2/7),(13/14))} + \delta_{((5/14),(11/14))} - \delta_{((3/14),(4/7))},$$

$$\ell_{15} = \delta_{((3/7),(9/14))} - \delta_{((5/14),(11/14))} + \delta_{((1/7),(5/7))} - \delta_{((3/14),(4/7))},$$
$$\ell_{16} = \delta_{((4/7),(6/7))} - \delta_{((11/14),(13/14))} + \delta_{((9/14),(5/7))} - \delta_{((3/7),(9/14))},$$
$$\ell_{17} = \delta_{((4/7),(6/7))} - \delta_{((9/14),(5/7))} + \delta_{((3/7),(9/14))} - \delta_{((5/14),(11/14))},$$
$$\ell_{18} = \delta_{((3/14),(4/7))} - \delta_{((3/7),(9/14))} + \delta_{((4/7),(6/7))} - \delta_{((5/14),(11/14))}.$$

The basis functions and collocation functionals for $i > 1$ are constructed recursively from those for $i = 1$ with the help of the contractive mapping family Φ. It is described how to carry out the construction in the general case in [**9**], and the above basis was constructed and used in [**14**]. We set the initial level $k = 3$ to implement the fast multilevel augmentation methods in the following two numerical examples.

1. Smooth kernel. Let the kernel $K(\mathbf{s}, \mathbf{t}) := \sin(s_1 s_2 + t_1 t_2)$ with $\mathbf{s} = (s_1, s_2)$ and $\mathbf{t} = (t_1, t_2)$. The function f is properly chosen so that $z(\mathbf{s}) = (s_1^2 + s_2^2)^2$ is an isolated solution of the equation. We see from Table 3 that $z_{3,m}$ and $\widetilde{z}_{3,m}$ have nearly the same accuracy. For any m, $\widetilde{T}_M(m+1)$ is about 4 times of $\widetilde{T}_M(m)$, which confirms the conclusion of Theorem 3.3. The compression rate decreases like $\mathcal{O}((3+m)4^{-(3+m)})$ according to Lemma 3.1, i.e., when m increases by 1, the rate decreases to a little bigger than quarter, which is confirmed by the data in Table 3.

TABLE 3. Numerical results for two-dimensional equation with smooth kernel.

m	d_{3+m}	$\|z^* - z_{3,m}\|_\infty$	$\|z^* - \widetilde{z}_{3,m}\|_\infty$	Comp. Rate	T_M	$\widetilde{T}_M$
0	48	1.077e-1	1.107e-1	0.4380	6.70	4.4
1	192	2.775e-2	2.780e-2	0.1563	43.07	19
2	768	7.612e-3	7.612e-3	0.0508	274.8	77
3	3072	2.157e-3	2.157e-3	0.0156	1955	311
4	12288		5.710e-4	0.0046		1255

TABLE 4. Numerical results for two-dimensional equation with singular kernel.

m	d_{3+m}	$\|z^* - z_{3,m}\|_\infty$	$\|z^* - \tilde{z}_{3,m}\|_\infty$	Comp. Rate	T_M	$\widetilde{T}_M$
0	48	1.094e-1	1.094e-1	0.789	7.3	4.4
1	192	2.834e-2	2.834e-2	0.499	43	37.6
2	768	7.679e-3	7.679e-3	0.254	278	197
3	3072	2.188e-3	2.188e-3	0.106	1982	963
4	12288		5.850e-4	0.039		4461

TABLE 5. Numerical results for solving one-dimensional equation with smooth kernel with the algorithm from [**6**].

m	d_{4+m}	$\|z^* - \widehat{z}_{4,m}\|_\infty$	Comp. Time
0	32	9.984e-3	2.06
1	64	2.452e-3	2.53
2	128	6.224e-4	3.01
3	256	1.551e-5	3.70
4	512	3.866e-5	4.60
5	1024	9.686e-6	6.30
6	2048	2.326e-6	7.80
7	4096	5.956e-7	11.3
8	8192	1.486e-8	17.0
9	16384	3.675e-8	35.1

2. Singular kernel. Let the kernel

$$K(\mathbf{s},\mathbf{t}) := \frac{1}{|\mathbf{s}-\mathbf{t}|} = \frac{1}{\sqrt{(s_1-t_1)^2+(s_2-t_2)^2}}.$$

It is easily seen that the kernel is weakly singular. Function f is properly chosen so that $z(\mathbf{s}) = (s_1^2+s_2^2)^2$ is an isolated solution of the equation. Table 4 gives data which also support the theoretical results from Lemma 3.2 and Theorem 3.3.

4.3. Comparison with fast multilevel augmentation methods for solving Hammerstein equations. In [**15**], multilevel augmentation methods are established for Hammerstein equations. Recently, [**6**] introduced approximation to modify the algorithm of [**15**], so as to make use of truncation techniques to remarkably improve computational efficiency. We use the first example of this section, which is a one-dimensional equation with smooth kernel, to compare our algorithm with that from [**6**]. Since the experimental data of our algorithm have been given in Table 1, we only need to report those for the algorithm from [**6**], which are listed in Table 5.

In the experiment we use the same multiscale bases and collocation functionals as those used by our algorithm. The initial level is also set to

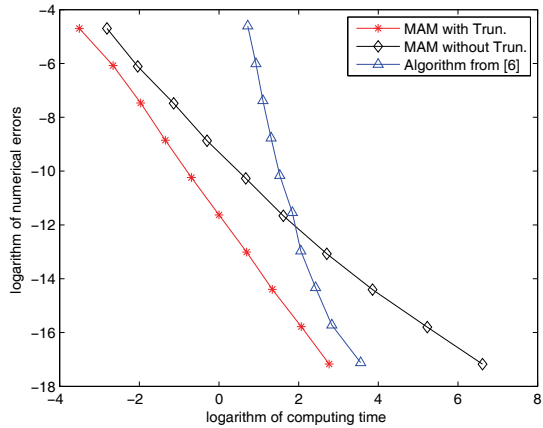

FIGURE 1. Comparison of the three algorithms with numerical errors vs. computing time.

be 4. The output of the algorithm is denoted by $u_{4,m}$ and we compute $\widehat{z}_{4,m} := \Psi(u_{4,m})$. Column 3 of the table gives the errors of the numerical solutions $\widehat{z}_{4,m}$, which can be seen to be nearly the same as the output of our algorithm. For example, $\|z^* - \widetilde{z}_{4,9}\|_\infty = 3.486\text{e-}8$ in Table 1, while $\|z - \widehat{z}_{4,9}\|_\infty = 3.675\text{e-}8$ in Table 5. The last column of Table 5 reports the running time of the program. We observe that, for $m = 9$, the running time of the two algorithms is 15.9 and 35.1, respectively, our algorithm being less than half of the one being compared.

As a conclusion, we would like to compare the computational efficiency of the three algorithms, i.e., the one proposed in this paper, the multilevel augmentation methods without matrix truncation which was described in [**8**], and the fast multilevel augmentation methods from [**6**] for Hammerstein equations. We still use the first example of this section, and collect data from Tables 1 and 5 to plot Figure 1. We intend to use Figure 1 to illustrate the efficiency of the algorithms. To this end, the x-axis of the figure represents the logarithm of computing time, while the y-axis represents the logarithm of numerical errors. We observe from the figure that, to reach the same accuracy, the algorithm proposed in this paper ("MAM with Trun") is always the fastest among the three algorithms. In small scale, the algorithm from [**8**] ("MAM

without Trun") is faster than that from [**6**]. However, since its computing time grows rapidly, it becomes the slowest algorithm when the discretization scale is large. The algorithm from [**6**] also makes use of matrix truncation to improve efficiency, and its computational complexity is also nearly linear order. However, in the same discretization scale, its computing time is at least twice of our algorithm.

REFERENCES

1. K.E. Atkinson, *A survey of numerical methods for solving nonlinear integral equations*, J. Int. Equat. Appl. **4** (1992), 15–46.

2. K.E. Atkinson and G. Chandler, *Boundary integral equation methods for solving Laplace's equation with nonlinear boundary conditions*: *The smooth boundary case*, Math. Comput. **55** (1990), 451–472.

3. K.E. Atkinson, I. Graham and I. Sloan, *Piecewise continuous collocation for integral equations*, SIAM J. Numer. Anal. **20** (1983), 172–186.

4. E. Babolian, F. Fattahzadeh and E. Raboky, *A Chebyshev approximation for solving nonlinear integral equations of Hammerstein type*, Appl. Math. Comput. **189** (2007), 641–646.

5. E. Babolian and A. Shahsavaran, *Numerical solution of nonlinear Fredholm integral equations of the second kind using Haar wavelets*, J. Comput. Appl. Math. **225** (2009), 87–95.

6. X. Chen, Z. Chen, B. Wu and Y. Xu, *Fast multilevel augmentation methods for nonlinear boundary integral equations*, SIAM J. Numer. Anal. **49** (2011), 2231–2255.

7. ———, *Fast multilevel augmentation methods for nonlinear boundary integral equations* II: *Efficient implementation*, J. Integral Equat. Appl., to appear.

8. Z. Chen, J. Li and Y. Zhang, *A fast multiscale solver for modified Hammerstein equations*, Appl. Math. Comput. **218** (2011), 3057–3067.

9. Z. Chen, C.A. Micchelli and Y. Xu, *Fast collocation methods for second kind integral equations*, SIAM J. Numer. Anal. **40** (2002), 344–375.

10. Z. Chen, B. Wu and Y. Xu, *Error control strategies for numerical integrations in fast collocation methods*, Northeast Math. J. **21** (2005), 233–252.

11. Z. Chen, B. Wu and Y. Xu, *Multilevel augmentation methods for solving operator equations*, Numer. Math. J. Chinese Univ. **14** (2005), 31–55.

12. ———, *Multilevel augmentation methods for differential equations*, Adv. Comput. Math. **24** (2006), 213–238.

13. ———, *Fast numerical collocation solutions of integral equations*, Comm. Pure Appl. Anal. **6** (2007), 643–666.

14. ———, *Fast collocation methods for high-dimensional weakly singular integral equations*, J. Integ. Equat. Appl. **20** (2008), 49–92.

15. ———, *Fast multilevel augmentation methods for solving Hammerstein equations*, SIAM. J. Num. Anal. **47** (2009), 2321–2346.

16. Z. Chen, Y. Xu and H. Yang, *A multilevel augmentation method for solving ill-posed operator equations*, Inverse Prob. **22** (2006), 155–174.

17. Z. Chen, Y. Xu and H. Yang, *Fast collocation methods for solving ill-posed integral equations of the first kind*, Inverse Prob. **24** (2008), 065007, 21 pages.

18. G. Han, *Exptrapolation of a discrete collocation–type method of Hammerstein equations*, J. Comput. Appl. Math. **61** (1995), 73–86.

19. G. Han and L. Zhang, *Asymptotic error expansion of a collocation–type method for Hammerstein equations*, Appl. Math. Comput. **72** (1995), 1–19.

20. H. Kaneko, R. Noren and B. Novaprateep, *Wavelet applications to the Petrov-Galerkin method for Hammerstein equations*, Appl. Num. Math. **45** (2003), 255–273.

21. H. Kaneko, R. Noren and P. Padilla, *Superconvergence of the iterated collocation methods for Hammerstein equations*, J. Comput. Appl. Math. **80** (1997), 335–349.

22. H. Kaneko, R. Noren and Y. Xu, *Numerical solutions for weakly singular Hammerstein equations and their superconvergence*, J. Integ. Equat. Appl. **4** (1992), 391–407.

23. H. Kaneko and Y. Xu, *Degenerate kernel method for Hammerstein equations*, Math. Comput. **56** (1991), 141–148.

24. ———, *Superconvergence of the iterated Galerkin methods for Hammerstein equations*, SIAM J. Numer. Anal. **33** (1996), 1048–1064.

25. S. Kumar, *Superconvergence of a collocation-type method for Hammerstein equations*, IMA J. Numer. Anal. **7** (1987), 313–325.

26. ———, *A discrete collocation-type method for Hammerstein equations*, SIAM J. Numer. Anal. **25** (1988), 328–341.

27. S. Kumar and I. Sloan, *A new collocation-type method for Hammerstein integral equations*, Math. Comp. **48** (1987), 585–593.

28. Y. Mahmoud, *Wavelet Galerkin method for numerical solution of nonlinear integral equation*, Appl. Math. Comput. **167** (2005), 1119–1129.

29. K. Maleknejad, K. Nouri and M. Sahlan, *Convergence of approximate solution of nonlinear Fredholm-Hammerstein integral equations*, Comm. Nonlin. Sci. Numer. Simu. **15** (2010), 3293–3298.

30. J. Saranen, *Projection methods for a class of Hammerstein equations*, SIAM J. Numer. Anal. **27** (1990), 1445–1449.

Department of Mathematics and System Science, College of Science, National University of Defense Technology, Changsha 410073, P.R. China
Email address: chenxiangling828@163.com

Department of Scientific Computing and Computer Applications, Sun Yat-Sen University, Guangzhou 510275, P.R. China
Email address: lnsczy@mail.sysu.edu.cn

Department of Scientific Computing and Computer Applications, Sun Yat-Sen University, Guangzhou 510275, P.R. China
Email address: wubin@mail.sysu.edu.cn