FUTURE POLYNOMIAL REGULARIZATION OF ILL-POSED VOLTERRA EQUATIONS*

AARON C. CINZORI† AND PATRICIA K. LAMM‡

Abstract. We examine a new discrete method for regularizing ill-posed Volterra problems. Unlike many classical regularization techniques (such as Tikhonov regularization), this method preserves the Volterra (causal) nature of the problem allowing the regularized solution to be produced sequentially in near real time as data arrives. We analyze the method and, for an important class of Volterra problems, prove that regularized solutions converge to the true solution at the best possible rate with respect to error in the data.

In fact, the future polynomial regularization method discussed here may be applied to quite general operator equations provided that the operator may be discretized by a lower-triangular matrix of Toeplitz type. This enlarges the class of operator equations that may be approximated using the method, but also introduces degenerate situations in which the future polynomial method is no more regularizing than an ordinary discretization method. We characterize these degenerate cases and argue that we are unlikely to see them for the problems of interest here. In particular, such degeneracies cannot occur for the class of Volterra problems for which we are able to prove the future polynomial method converges.

Finally we present numerical evidence that this method works well in the recovery of sharp and discontinuous features in the true solution, features that can be oversmoothed by classical regularization techniques.

Key words. First-kind Volterra equations, inverse problems, regularization, collocation method

AMS subject classifications. 65J20, 65R30, 45D05, 35R30

1. Introduction.

We consider an inverse problem based on a first-kind Volterra integral equation of the form

$$\mathcal{A}u = f \tag{1.1}$$

where $\mathcal{A}$ is a bounded linear operator defined on $L_2(0,1)$ by

$$\mathcal{A}u(t) = \int_0^t k(t-s)u(s)\,ds, \quad \text{a.e. } t \in [0,1]. \tag{1.2}$$

We will assume that the data function $f \in L_2(0,1)$ and the convolution kernel k are such that (1.1) has a unique solution $\bar{u} \in L_2(0,1)$ [9]. The usual situation in applications is that the range of $\mathcal{A}$ is not closed; in this case, the inverse problem defined by (1.1) is ill-posed due to lack of continuous dependence of solutions on data.

A classic example of a problem with this sort of instability is the Inverse Heat Conduction Problem (IHCP). Consider an insulated semi-infinite bar coincident with the non-negative x-axis. An unknown, time-varying heat source $u(t)$ is applied to the end of the bar at $x = 0$, and the temperature of the bar at position $x = 1$ is recorded by the time-varying function $f(t)$. The unknown heat source $u(t)$ is then the solution of the first-kind equation (1.1), where the kernel k in $\mathcal{A}$ is given by

$$k(t) = \frac{1}{2\sqrt{\pi}t^{3/2}}e^{-1/4t}.$$

*This work was supported in part by the National Science Foundation under contract number NSF DMS 9704899 (P.K.L.)

†Department of Mathematics, Allegheny College, Meadville, PA 16335-3902 (acinzori@alleg.edu).

‡Department of Mathematics, Michigan State University, East Lansing, MI 48824-1027 (lamm@math.msu.edu).

The IHCP is an example of an extremely unstable ill-posed problem. In general, the level of instability of a first-kind Volterra problem like (1.1) is characterized by the following property of the kernel k. The kernel k is called ν-smoothing if there exists an integer $\nu > 0$ such that $k \in H^{\nu}(0,1)$ with $k(0) = k'(0) = \cdots = k^{(\nu-2)}(0) = 0$ and $k^{(\nu-1)}(0) \neq 0$. As ν increases, the level of instability of the problem increases as well. The extreme instability of IHCP is illustrated by the fact that $k^{(\nu)}(0) = 0$ for all $\nu = 0, 1, 2, \ldots$. However, as is well known in the literature (and as will be seen in the numerical examples), even 1-smoothing problems can be sufficiently unstable as to require some kind of regularization technique in order to obtain useful solutions.

A classical regularization technique such as Tikhonov regularization may be used to stabilize problems such as (1.1), but such techniques tend to destroy the causal nature of the Volterra problem. By causal we mean that the original problem (1.1) has the property that, given $t_0 \in (0,1)$, the solution u on the interval $[0, t_0)$ is determined only from the values of f on that same interval; i.e., the solution only depends on information from the past. In the course of applying a classical method such as Tikhonov regularization, the problem is transformed in such a way that all values of f, past and future, are required to find the solution at a given t.

A second drawback of classical methods such as Tikhonov regularization is that such methods tend to oversmooth solutions. An illustration of the problem may be seen in Figure 1.1. Here the true solution $\bar{u}$ is given by the dashed line and the regularized solution is given by the solid line. The lack of stability in the problem is illustrated by the highly oscillatory behavior of the solution provided by simple collocation. As Tikhonov's α increases, providing more stability, the Tikhonov solution loses the peaks in the true solution. Recently other regularization techniques such as the idea of bounded variation regularization [1, 4, 7, 10, 11, 24] and that of regularization for curve representations [22, 23], have been used to effectively circumvent the problem of oversmoothing. In addition, better results may be obtained with standard Tikhonov regularization if the basis elements are chosen to better suit the particular problem of interest (instead of the somewhat generic piecewise-constant function approximations used in Figure 1.1). However, none of these methods preserve the causal nature of Volterra problems and, additionally, several can require a reformulation of the original linear problem (or quadratic problem, if viewed in a least-squares sense) into a nondifferentiable or nonquadratic optimization problem.

We present here a regularization method which both preserves the causal nature of the original problem (1.1) and avoids oversmoothing of the regularized solution. In addition, the regularization is performed in least-squares setting, avoiding a reformulation into a nondifferentiable or nonquadratic optimization problem.

2. Discrete regularization methods. For simplicity in what follows, we will assume henceforth that the convolution kernel k in (1.1) is real-valued and continuous, and that $k(t) \neq 0$ for all $t > 0$ sufficiently small.

We let $N = 1, 2, \ldots$ be fixed and define the discretization stepsize $h = 1/N$, with $t_i = ih$ for $i = 0, 1, \ldots, N$. For $i = 2, 3, \ldots, N$, let $\chi_i(t)$ be the indicator function on the interval $(t_{i-1}, t_i]$ (i.e., $\chi_i(t) = 1$, $t \in (t_{i-1}, t_i]$, and $\chi_i(t) = 0$, otherwise), and let χ_1 be the indicator function on $[t_0, t_1]$. Let $\mathcal{S}^N = \operatorname{span}\{\chi_i\}$ denote an approximation space of piecewise constant functions on $[0,1]$.

To motivate our ideas, we first consider a standard discretization of the original problem (1.1). A simple collocation of (1.1) involves finding $q \in \mathcal{S}^N$, i.e., q of the

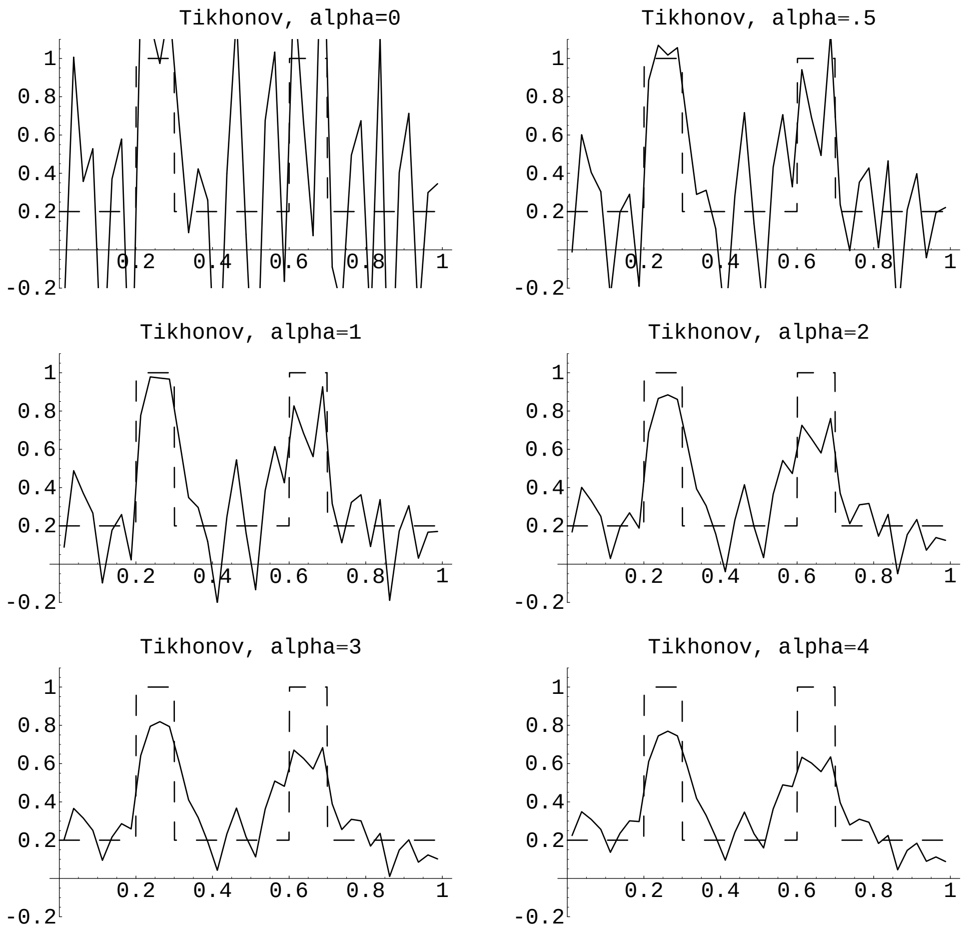

FIG. 1.1. *Tikhonov regularization*

form

$$q(t) = \sum_{j=1}^{N} c_j \chi_j(t), \quad t \in [0,1], \tag{2.1}$$

where the constants $c_j \in \mathbb{R}$, $j = 1, \ldots, N$, are determined by requiring that q solve the collocation equations

$$\mathcal{A}q(t_i) = f(t_i), \quad i = 1, 2, \ldots, N. \tag{2.2}$$

(See [3], for example.) Because $\mathcal{A}$ is Volterra this collocation process leads to a lower-triangular Toeplitz matrix equation

$$\boldsymbol{A}\boldsymbol{c} = \boldsymbol{f}, \tag{2.3}$$

where $\boldsymbol{A} \in \mathbb{R}^{N\times N}$ and $\boldsymbol{f} \in \mathbb{R}^N$ are given by

$$\boldsymbol{A} = \begin{pmatrix} \Delta_1 & 0 & \cdots & 0 \\ \Delta_2 & \Delta_1 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ \Delta_N & \Delta_{N-1} & \cdots & \Delta_1 \end{pmatrix}, \quad \boldsymbol{f} = \begin{pmatrix} f_1 \\ f_2 \\ \vdots \\ f_N \end{pmatrix}, \tag{2.4}$$

and where, for $i = 1, \dots, N$,

$$\Delta_i = \int_0^{t_1} k(t_i - s)\,ds, \tag{2.5}$$

$$f_i = f(t_i). \tag{2.6}$$

In the above we have used the fact that, for $1 \le j \le i \le N$,

$$\begin{aligned} \mathcal{A}\chi_j(t_i) &= \int_{t_{j-1}}^{t_j} k(t_i - s)\,ds \\ &= \int_{t_0}^{t_1} k(t_{i-j+1} - s)\,ds \\ &= \Delta_{i-j+1}. \end{aligned} \tag{2.7}$$

Under the stated conditions on k, it is clear that $\Delta_1 \neq 0$ for all h sufficiently small, and thus that there is a unique solution $(c_1, \dots, c_N)^T \in \mathbb{R}^N$ of (2.3).

In order to motivate what is to follow, we note that the algorithm described by (2.2) (equivalently (2.3)) may alternatively be formulated in a sequential least-squares setting as follows.

Sequential Collocation Algorithm:
Assuming $c_1, \dots, c_{i-1}$ have already been determined, find $c_i \in \mathbb{R}$ so that

$$c_i = \arg\min_{c\in\mathbb{R}} J_i(c), \tag{2.8}$$

where

$$J_i(c) = \left(\mathcal{A}\left(\sum_{j=1}^{i-1} c_j \chi_j + c\chi_i \right)(t_i) - f(t_i) \right)^2. \tag{2.9}$$

Although the finite-dimensional procedure (2.8)–(2.9) is now a well-posed problem, it tends to be poorly-conditioned as can be seen clearly if Δ_1 is small relative to the off-diagonal entries in the leading matrix in the equivalent matrix equation (2.3); this ill-conditioning leads to poor approximations unless N is small.

In the 1960's, J. V. Beck (see, for example, the survey in [2]) proposed a regularizing variation of this procedure for the solution of the IHCP. The idea behind this method (generalized here to include all Volterra problems of the form (1.1)) is based on the selection of a regularization parameter r, where $(r-1)h$ is the length of a future regularization interval. This algorithm is given as follows:

Sequential "Future Constant Regularization" Algorithm:
Let $r \ge 1$ be a fixed integer. Assuming $c_1, \dots, c_{i-1}$ have already been determined, find $c_i \in \mathbb{R}$ such that

$$c_i = \arg\min_{c\in\mathbb{R}} J_{i,r}(c), \tag{2.10}$$

where

$$J_{i,r}(c) = \sum_{l=0}^{r-1} \left(\mathcal{A}\left(\sum_{j=1}^{i-1} c_j \chi_j + c \sum_{m=0}^{l} \chi_{i+m} \right)(t_{i+l}) - f(t_{i+l}) \right)^2. \tag{2.11}$$

For $r = 1$, this algorithm reduces to the sequential collocation algorithm (2.8)–(2.9), where one finds c_i to be the constant-valued solution on the interval $[t_{i-1}, t_i]$ which best matches model to data at the single point t_i. For $r > 1$, the procedure in (2.10)–(2.11) adds regularization to the i^{th} step by defining c_i to be the best constant-valued solution on the (present, plus future) interval $[t_{i-1}, t_{i-1+r}]$, where now "best" is in the sense of a least-squares fit of model to data at r discrete points in that extended interval. However, c_i is only retained as the value of the solution on the interval $[t_{i-1}, t_i]$, as the remainder of the rigid solution on $[t_i, t_{i-1+r}]$ is dropped as soon as one starts the least-squares fitting procedure all over again with the $(i+1)^{st}$ step.

Although the future constant algorithm (2.10)–(2.11) is easily implemented and has been found to be an effective sequential regularization method for the approximate solution of (1.1) (see [2] and the theoretical studies in [12, 13, 14]), the method tends to oversmooth regularized solutions. (Numerical examples illustrating this behavior may be found in §6.) Indeed this is not surprising since the constant c_i at the i^{th} step is determined by temporarily forcing the discretized solution to remain constant on an extended future interval. We present here a method which is a direct generalization of (2.10)–(2.11) but which does not seem to suffer from the same drawback. This generalization uses a second regularization parameter $d \geq 0$ where d is an integer (in addition to the regularization parameter r). We will use the notation $\mathcal{P}^d$ to denote the space of real-valued polynomials of degree d or less.

Sequential "Future Polynomial Regularization" Algorithm:
Let $d \geq 0$ and $r \geq 1$ be integers. Assuming $c_1, \ldots, c_{i-1}$ have already been determined, find $c_i \in \mathbb{R}$ such that

$$c_i = p_{i,r,d}(t_i) \tag{2.12}$$

where the polynomial $p_{i,r,d} \in \mathcal{P}^d$ is determined by

$$p_{i,r,d} = \arg\min_{p \in \mathcal{P}^d} J_{i,r,d}(p), \tag{2.13}$$

where, for any $p \in \mathcal{P}^d$,

$$\begin{aligned} J_{i,r,d}(p) \\ &= \sum_{l=0}^{r-1} \left(\mathcal{A}\left(\sum_{j=1}^{i-1} c_j \chi_j + \sum_{m=0}^{l} p(t_{i+m}) \chi_{i+m} \right)(t_{i+l}) - f(t_{i+l}) \right)^2 . \end{aligned} \tag{2.14}$$

Thus while the future constant algorithm determines c_i at the i^{th} step by temporarily forcing a constant-valued solution on an interval of length rh, the future polynomial method enforces less rigidity by seeking a d^{th}-degree polynomial solution on the same interval. In both cases, the rigid solution is only used temporarily (as the value on the future interval is dropped as soon as one goes to the $(i+1)^{st}$ step); nevertheless one can see how oversmoothing might be prevented by taking $d > 0$. We note that for $d = 0$ the future polynomial method reduces to the future constant method. We should also remark that, for the special case of $d = 1$, a variation of the above algorithm was proposed in [2], where there the solution q was sought in the space of continuous linear spline functions. As noted in [19], however, imposing continuity on q may not be an advantage in the solution of first-kind Volterra problems because the resulting approximating equations tend to be less stable.

For the future polynomial method, both r and d function as regularization parameters. In general we will take $r \geq d+2$ since forcing fewer than $d+2$ points to lie on a d^{th} degree polynomial is no restriction at all and thus provides no regularization. In fact, when $r \leq d+1$ the future polynomial method reduces to the simple collocation algorithm (2.8)–(2.9). Thus we will generally restrict consideration to $r \geq d+2$ unless we wish to make a comparison with standard collocation (which we will represent using $r = d+1$). We also note that increasing r provides more regularization, while increasing d provides less regularization. In addition, since the case of $r = d+2$ is the first case where regularization occurs, we will focus at times in what follows on this case, viewing it as a one-parameter family (in the parameter r) of regularization techniques.

Because the future polynomial regularization method is based on holding the solution temporarily rigid on a small future interval, it is clear that we either need to extend the original problem (1.1) a little past $[0,1]$, or else settle for approximating u on $[0, 1-\epsilon]$, where $\epsilon > 0$ is small. We will take the former approach here and assume that there is $T > 1$ such that equation (1.1) holds on the interval $[0,T]$, where k and f are both defined on this extended interval.

3. Future polynomial regularization. As seen in §2, equation (2.3) is the matrix equation in the unknown vector $\boldsymbol{c} = (c_1, \ldots, c_N)^T$ associated with the sequential collocation algorithm (2.8)–(2.9). In what follows we derive an analogous equation in $\boldsymbol{c}$ which is equivalent to the algorithm (2.12)–(2.14) of future polynomial regularization. Throughout this section we will assume that $r \geq d+1$, with $r \geq d+2$ corresponding to the regularized cases. We will also assume that N is sufficiently large so that $r \leq N$.

To make precise the selection of the i^{th} polynomial $p_{i,r,d} \in \mathcal{P}^d$ defined in (2.13), we let $b_{i,0}, b_{i,1}, \ldots, b_{i,d}$ be real scalars and make the definition

$$p_{i,r,d}(t) = \sum_{j=0}^{d} \frac{1}{h^j} b_{i,j} (t - t_{i-1})^j. \tag{3.1}$$

Then $p_{i,r,d}(t_{i+m}) = \sum_{j=0}^{d} b_{i,j}(m+1)^j$, and $J_{i,r,d}$ defined in (2.14) may be expressed as a function of $b_{i,j}$, $j = 0, 1, \ldots, d$. To this end, we use (2.6) and (2.7) to write $J_{i,r,d}$ as

$$\begin{aligned}
&J_{i,r,d}(b_{i,0}, \ldots, b_{i,d}) \\
&= \sum_{l=0}^{r-1} \left(\left(\sum_{j=1}^{i-1} c_j \Delta_{i+l-j+1} \right) + \sum_{m=0}^{l} \left(\sum_{j=0}^{d} b_{i,j}(m+1)^j \right) \Delta_{l-m+1} - f_{i+l} \right)^2 \\
&= \sum_{l=0}^{r-1} \left(\left(\sum_{j=1}^{i-1} c_j \Delta_{i+l-j+1} \right) + \sum_{j=0}^{d} b_{i,j} \left(\sum_{m=0}^{l} (m+1)^j \Delta_{l-m+1} \right) - f_{i+l} \right)^2 \\
&= \sum_{l=0}^{r-1} \left(\left(\sum_{j=1}^{i-1} c_j \Delta_{i+l-j+1} \right) + \sum_{j=0}^{d} b_{i,j} K_{l,j} - f_{i+l} \right)^2,
\end{aligned}$$

where

$$K_{l,j} = \sum_{m=0}^{l} (m+1)^j \Delta_{l-m+1}, \quad l = 0, 1, \ldots, r-1, \;\; j = 0, 1, \ldots, d. \tag{3.2}$$

Necessary conditions for the minimization of $J_{i,r,d}$ with respect to $b_{i,k}$, for $k = 0, \dots, d$, are then

$$\sum_{j=0}^{d} b_{i,j} \left(\sum_{l=0}^{r-1} K_{l,j} K_{l,k} \right) = \sum_{l=0}^{r-1} \left(f_{i+l} - \sum_{j=1}^{i-1} c_j \Delta_{i+l-j+1} \right) K_{l,k} \tag{3.3}$$

for $k = 0, 1, \dots, d$, a system in the $d+1$ unknowns, $b_{i,0}, b_{i,1}, \dots, b_{i,d}$.

Let $\boldsymbol{b}_i \in \mathbb{R}^{d+1}$ and $\boldsymbol{v}_{l,r}$, $\boldsymbol{f}_{l,r} \in \mathbb{R}^r$, $l = 1, \dots, N$ be defined by

$$\boldsymbol{b}_i = (b_{i,0}, b_{i,1}, \dots, b_{i,d})^T \tag{3.4}$$

$$\boldsymbol{v}_{l,r} = (\Delta_l, \Delta_{l+1}, \dots, \Delta_{l+r-1})^T \tag{3.5}$$

$$\boldsymbol{f}_{l,r} = (f_l, f_{l+1}, \dots, f_{l+r-1})^T, \tag{3.6}$$

and define the $r \times (d+1)$ matrix $\boldsymbol{K}$ by $(\boldsymbol{K})_{i,j} = K_{i,j}$, for $i = 0, \dots, r-1,\ j = 0, \dots d$. That is,

$$\boldsymbol{K} = \begin{pmatrix} \Delta_1 & \Delta_1 & \cdots & \Delta_1 \\ \Delta_1 + \Delta_2 & 2\Delta_1 + \Delta_2 & \cdots & 2^d\Delta_1 + \Delta_2 \\ \Delta_1 + \Delta_2 + \Delta_3 & 3\Delta_1 + 2\Delta_2 + \Delta_3 & \cdots & 3^d\Delta_1 + 2^d\Delta_2 + \Delta_3 \\ \vdots & \vdots & & \vdots \\ \Delta_1 + \cdots + \Delta_r & r\Delta_1 + \cdots + \Delta_r & \cdots & r^d\Delta_1 + \cdots + \Delta_r \end{pmatrix}.$$

It is easy to see that $\boldsymbol{K} = \boldsymbol{L}\boldsymbol{V}$, where $\boldsymbol{L} \in \mathbb{R}^{r \times r}$ is the leading $r \times r$ block of $\boldsymbol{A}$ in (2.3), namely,

$$\boldsymbol{L} = \begin{pmatrix} \Delta_1 & 0 & \cdots & 0 \\ \Delta_2 & \Delta_1 & \ddots & \vdots \\ \vdots & \vdots & \ddots & 0 \\ \Delta_r & \Delta_{r-1} & \cdots & \Delta_1 \end{pmatrix}, \tag{3.7}$$

and $\boldsymbol{V} \in \mathbb{R}^{r \times (d+1)}$ is defined by

$$\boldsymbol{V} = \begin{pmatrix} 1 & 1 & 1 & \cdots & 1 \\ 1 & 2 & 4 & \cdots & 2^d \\ \vdots & \vdots & \vdots & & \vdots \\ 1 & r & r^2 & \cdots & r^d \end{pmatrix}. \tag{3.8}$$

Note that $\boldsymbol{V}$ is a Vandermonde matrix and so is of full rank $(d+1)$ and that $\boldsymbol{L}$ is invertible (since $\Delta_1 \neq 0$ from the assumptions on the kernel k). Thus $\boldsymbol{K}$ is of full rank $(d+1)$.

Using these definitions we may write system (3.3) as a matrix equation in the unknown vector $\boldsymbol{b}_i$,

$$\boldsymbol{K}^T\boldsymbol{K}\boldsymbol{b}_i = \boldsymbol{K}^T(\boldsymbol{f}_{i,r} - \sum_{j=1}^{i-1} c_j \boldsymbol{v}_{i+1-j,r}), \tag{3.9}$$

or, since $\boldsymbol{K}^T\boldsymbol{K}$ is invertible, we may use $\boldsymbol{K}^+ = (\boldsymbol{K}^T\boldsymbol{K})^{-1}\boldsymbol{K}^T$ to denote the (Moore-Penrose) pseudoinverse of $\boldsymbol{K}$ (see, for example, [5]) and write

$$\boldsymbol{b}_i = \boldsymbol{K}^+ \left(\boldsymbol{f}_{i,r} - \sum_{j=1}^{i-1} c_j \boldsymbol{v}_{i+1-j,r} \right). \tag{3.10}$$

From the above arguments we see that $p_{i,r,d}$ defined in (3.1) (using the components of $\boldsymbol{b}_i$ in (3.10) as coefficients) is the unique minimizer in $\mathcal{P}^d$ of $J_{i,r,d}$ in (2.14). Now the remaining work in the i^{th} step of the future polynomial algorithm (2.12)–(2.14) is to set $c_i = p_{i,r,d}(t_i)$. That is,

$$\begin{aligned} c_i &= \sum_{j=0}^{d} \frac{1}{h^j} b_{i,j}(t_i - t_{i-1})^j \\ &= \boldsymbol{b}_i^T (1, \ldots, 1)^T \\ &= \left(\boldsymbol{f}_{i,r} - \sum_{j=1}^{i-1} c_j \boldsymbol{v}_{i+1-j,r} \right)^T \left(\boldsymbol{K}^T \right)^+ (1, \ldots, 1)^T \end{aligned} \tag{3.11}$$

where we have used the fact that $(\boldsymbol{K}^+)^T = (\boldsymbol{K}^T)^+$ [5] and where $(1, \ldots, 1)^T$ is a vector in $\mathbb{R}^{d+1}$.

Using the above construction we may write a precise expression which allows for the determination of the c_i's via a single matrix equation. To this end, we note from (3.11) that c_i satisfies, for $i = 1, \ldots, N$,

$$\sum_{j=1}^{i-1} c_j \left(\boldsymbol{v}_{i+1-j,r}^T (\boldsymbol{K}^T)^+ (1, \ldots, 1)^T \right) + c_i = \boldsymbol{f}_{i,r}^T (\boldsymbol{K}^T)^+ (1, \ldots, 1)^T.$$

That is,

$$(\tilde{\boldsymbol{A}} + \tilde{\alpha} \boldsymbol{I}) \boldsymbol{c} = \tilde{\boldsymbol{f}}, \tag{3.12}$$

where $\boldsymbol{I}$ denotes the $N \times N$ identity matrix, $\boldsymbol{c} = (c_1, \ldots, c_N)^T$,

$$\tilde{\boldsymbol{A}} = \begin{pmatrix} \tilde{\Delta}_1 & 0 & \cdots & 0 \\ \tilde{\Delta}_2 & \tilde{\Delta}_1 & \ddots & \vdots \\ \vdots & \vdots & \ddots & 0 \\ \tilde{\Delta}_N & \tilde{\Delta}_{N-1} & \cdots & \tilde{\Delta}_1 \end{pmatrix}, \quad \tilde{\boldsymbol{f}} = \begin{pmatrix} \tilde{f}_1 \\ \tilde{f}_2, \\ \vdots \\ \tilde{f}_N \end{pmatrix}, \tag{3.13}$$

and

$$\tilde{\alpha} = 1 - \tilde{\Delta}_1, \tag{3.14}$$

with $\tilde{\Delta}_i = \boldsymbol{v}_{i,r}^T (\boldsymbol{K}^T)^+ (1, \ldots, 1)^T$, $\tilde{f}_i = \boldsymbol{f}_{i,r}^T (\boldsymbol{K}^T)^+ (1, \ldots, 1)^T$, for $i = 1, \ldots, N$. That is,

$$\tilde{\Delta}_i = \sum_{l=1}^{r} \tau_l \Delta_{i+l-1}, \tag{3.15}$$

$$\tilde{f}_i = \sum_{l=1}^{r} \tau_l f_{i+l-1}, \tag{3.16}$$

for $i = 1, \ldots, N$, where $\boldsymbol{\tau} = (\tau_1, \ldots, \tau_r)^T \in \mathbb{R}^r$ is given by

$$\boldsymbol{\tau} = \left(\boldsymbol{K}^T \right)^+ (1, \ldots, 1)^T, \tag{3.17}$$

for $(1,\ldots,1)^T \in \mathbb{R}^{d+1}$. We note that the dependence of many of the above quantities on r, d, and N has been suppressed.

So it is clear that the constants c_i determined by the future polynomial algorithm (2.12)–(2.14) are equivalently found as the solution of (3.12), a matrix equation which is always consistent and which retains the lower triangular (causal), Toeplitz structure associated with approximations of the original unregularized problem (1.1).

An interesting result is that equation (3.12) also arises in another setting, namely, that of a collocation-based discretization of a second-kind Volterra equation, one that is a variation of the original Volterra problem (1.1). This is the content of the theorem which follows, a result which places the future polynomial regularization method in (a generalization of) the class of regularization methods developed in [13], a class which has already been shown to include the future constant method that Beck applied to the IHCP. In addition, because we know that second-kind Volterra equations are well-posed, this result gives us hope that this particular method offers improved regularization over simple collocation alone.

THEOREM 3.1. *Let $d \geq 0$ and $r \geq d+1$ be given integers, and let $h = 1/N$ for fixed $N = 1, 2, \ldots$. Define $\Delta r = (r-1)h$. Then the solution $q(\cdot) \in \mathcal{S}^N$ given by (2.1) of the future polynomial regularization algorithm (2.12)–(2.14) satisfies the following Volterra equation*

$$\int_0^t \tilde{k}(t-s;\Delta r)u(s)\,ds + \alpha(\Delta r)u(t) = \tilde{f}(t;\Delta r), \quad t \in [0,1], \tag{3.18}$$

exactly at collocation points $t_i = ih$, $i = 1,\ldots,N$, where

$$\tilde{k}(t;\Delta r) = \int_0^{\Delta r} k(t+\rho)\,d\eta_{\Delta r}(\rho), \tag{3.19}$$

$$\tilde{f}(t;\Delta r) = \int_0^{\Delta r} f(t+\rho)\,d\eta_{\Delta r}(\rho), \tag{3.20}$$

$$\alpha(\Delta r) = \int_0^{\Delta r}\int_0^{\rho} k(\rho-s)\,ds\,d\eta_{\Delta r}(\rho), \tag{3.21}$$

and where $\eta_{\Delta r}$ is a discrete signed measure defined for bounded g on $[0,\Delta r]$ by

$$\int_0^{\Delta r} g(\rho)\,d\eta_{\Delta r}(\rho) = \sum_{l=1}^{r} \tau_l g(t_{l-1}), \tag{3.22}$$

with $\boldsymbol{\tau} = (\tau_1,\ldots,\tau_r)^T$ defined in (3.17). If $\alpha(\Delta r) \neq 0$, then (3.18) is a Volterra integral equation of second kind, with solutions depending continuously on data.

Proof. We show that the collocation-based discretization of (3.18) leads precisely to equation (3.12). Defining the Δr-dependent operator $\tilde{\mathcal{A}}$ by

$$\tilde{\mathcal{A}}u(t) = \int_0^t \tilde{k}(t-s;\Delta r)u(s)\,ds, \quad t \in [0,1],$$

for $u \in L_2(0,1)$, we note that for $1 \leq j \leq i \leq N$,

$$\tilde{\mathcal{A}}\chi_j(t_i) = \int_{t_{j-1}}^{t_j}\int_0^{\Delta r} k(t_i+\rho-s)\,d\eta_{\Delta r}(\rho)\,ds$$

$$
\begin{aligned}
&= \sum_{l=1}^{r} \tau_l \int_{t_{j-1}}^{t_j} k(t_i + t_{l-1} - s)\, ds \\
&= \sum_{l=1}^{r} \tau_l \int_{0}^{t_1} k(t_{i+l-j} - s)\, ds \\
&= \sum_{l=1}^{r} \tau_l \Delta_{i+l-j} \\
&= \tilde{\Delta}_{i-j+1},
\end{aligned}
$$

where $\tilde{\Delta}_i$ is defined in (3.15). Thus $\tilde{\mathcal{A}}q(t_i) = \sum_{j=1}^{i} c_j \tilde{\Delta}_{i-j+1}$, for $i = 1, \ldots, N$. In addition, $\tilde{f}(t_i; \Delta r) = \sum_{l=1}^{r} \tau_l f(t_i + t_{l-1}) = \tilde{f}_i$, for $i = 1, \ldots, N$. It remains to show that $\alpha(\Delta r) = \tilde{\alpha} = 1 - \tilde{\Delta}_1$. From (3.21) and (3.22),

$$
\alpha(\Delta r) = \sum_{l=1}^{r} \tau_l \int_0^{t_{l-1}} k(t_{l-1} - s)\, ds, \tag{3.23}
$$

where a change of integration variable gives

$$
\begin{aligned}
\alpha(\Delta r) &= \sum_{l=1}^{r} \tau_l \left[\int_0^{t_l} k(t_l - s)\, ds - \int_0^{t_1} k(t_l - s)\, ds \right] \\
&= \sum_{l=1}^{r} \tau_l \left[(\Delta_1 + \cdots + \Delta_l) - \Delta_l \right] \qquad (3.24)\\
&= (\boldsymbol{K}\boldsymbol{e}_1)^T \boldsymbol{\tau} - \tilde{\Delta}_1,
\end{aligned}
$$

where $\tilde{\Delta}_1$ is defined in (3.15). Thus, using the definitions of $\boldsymbol{\tau}$ in (3.17),

$$
\begin{aligned}
\alpha(\Delta r) &= \left(\boldsymbol{K}^+ \boldsymbol{K}\boldsymbol{e}_1\right)^T (1, \ldots, 1)^T - \tilde{\Delta}_1 \\
&= 1 - \tilde{\Delta}_1 \qquad (3.25)
\end{aligned}
$$

where we have additionally used the fact that $\boldsymbol{K}$ has full column rank and thus $\boldsymbol{K}^+ \boldsymbol{K}\boldsymbol{e}_1 = \boldsymbol{e}_1$. □

Finally, we close the section by summarizing the three equivalent characterizations of the future polynomial regularization method that we have found. The method may be viewed as

- A sequential algorithm (2.12)–(2.14), in which the solution is found by solving a sequence of regularized least-squares problems over small future intervals;
- A second-kind matrix equation (3.12), a perturbation of the original first-kind matrix equation (2.3); and,
- A collocation-based discretization of a second-kind Volterra equation (3.18) related to the original Volterra problem (1.1). In this case, the solution is given by $\sum_{j=1}^{N} c_j \chi_j(\cdot)$, where $(c_1, \ldots, c_N)^T$ satisfies, for $i = 1, \ldots, N$,

$$
c_i \alpha(\Delta r) + \int_0^{t_i} \tilde{k}(t_i - s; \Delta r) \sum_{j=1}^{i} c_j \chi_j(s)\, ds = \tilde{f}(t_i; \Delta r), \tag{3.26}
$$

  with $\Delta r = (1 - r)h$; here $\tilde{k}$, $\tilde{f}$, and $\alpha(\Delta r)$ are defined in (3.19)–(3.21), respectively.

The third approach gives a useful characterization when it comes to proving convergence in the next section.

4. Convergence of the Future Polynomial Regularization Method. In this section we prove convergence of the solution provided by the future polynomial method to the true solution $\bar{u}$ of (1.1) in the case of $r = d+2$, for $d = 0, 1, \ldots$ fixed. Throughout the section, all function norms will be the L_∞-norm and all vector and matrix norms will be the 2-norm unless otherwise noted.

We will also make the following standing hypotheses throughout this section. Assume that $\bar{u}, k \in C^1[0,T]$ with $k(0) \neq 0$. Without loss of generality we will let $k(0) = 1$, so that $\Delta_1 > 0$ for all h sufficiently small. In addition we consider a perturbation f^δ of the data function f, where $f^\delta(t) = f(t) + e(t)$, $t \in [0,T]$, with e piecewise continuous on $[0,T]$ and such that $\|e\| \leq \delta$ for some fixed $\delta > 0$. As is typical of classical theory for discrete regularization methods, convergence occurs provided the stepsize h is selected in accordance with the amount δ of error in the data.

THEOREM 4.1. *Let $r = d+2$ for fixed $d = 0, 1, \ldots$ and let $h = 1/N$. Let $u_h(\cdot; r, f^\delta) = \sum_{j=1}^N c_j \chi_j$ denote the solution given by the sequential future polynomial algorithm (2.12)–(2.14) for given h, r, and where f^δ replaces f in (2.14). Then, if $h \sim c\sqrt{\delta}$ for any constant $c > 0$, convergence of $u_h(t; r, f^\delta)$ to $\bar{u}(t)$ occurs at the collocation points as $\delta \to 0$, at the best possible rate with respect to the level of error in the data. That is,*

$$|u_h(t_j; r, f^\delta) - \bar{u}(t_j)| \leq \delta^{1/2} C(r) + \mathcal{O}(\delta), \quad j = 1, \ldots, N(\delta), \tag{4.1}$$

as $\delta \to 0$ where $C(r)$ is a positive constant and $N(\delta) = 1/h(\delta)$.

As in [12], the theory is developed by viewing the regularization method as a discretization of a second-kind Volterra equation of the form (3.18) with a specified measure $\eta_{\Delta r}$. We depart from [12], however, in that our measure $\eta_{\Delta r}$ need not be positive and the quantity $\alpha(\Delta r)$ in (3.21) may actually be zero.

Before turning to the proof we will need a few preliminary technical results regarding the size as $h \to 0$ of $\alpha(\Delta r)$ and $\boldsymbol{\tau}$ (given in (3.21) and (3.17), respectively, with $\Delta r = (r-1)h$).

LEMMA 4.2. *Let $r = d+2$ for fixed $d = 0, 1, \ldots$. Let $h > 0$ be small and $\Delta r = (r-1)h$. If the matrix $\boldsymbol{L}$ defined by (3.7) is such that $\left[(\boldsymbol{L}^T)^{-1}\boldsymbol{n}_1\right]_1 > 1/\Delta_1$, for $\boldsymbol{n}_1 \in \mathbb{R}^{d+2}$ given by*

$$\boldsymbol{n}_1 = \left(\binom{d+1}{0}, -\binom{d+1}{1}, \ldots, (-1)^{d+1}\binom{d+1}{d+1} \right)^T, \tag{4.2}$$

then $0 \leq \alpha(\Delta r) < 1$.

Proof. Recall that $\boldsymbol{\tau}$ as defined by (3.17) is the minimum norm solution of

$$\boldsymbol{K}^T \boldsymbol{x} = (1, \ldots, 1)^T,$$

where $(1, \ldots, 1)^T \in \mathbb{R}^{d+1}$. But $(1/\Delta_1)\boldsymbol{K}^T \boldsymbol{e}_1 = (1, \ldots, 1)^T$, so it follows that

$$\boldsymbol{\tau} = \frac{1}{\Delta_1}\boldsymbol{e}_1 + \boldsymbol{z} \tag{4.3}$$

for some $\boldsymbol{z} \in \mathcal{N}(\boldsymbol{K}^T)$, the nullspace of $\boldsymbol{K}^T$. Now for $r = d+2$, the nullspace of $\boldsymbol{V}^T$ is 1-dimensional, and using a combinatorial result (see, for example, [6]) one can show that $\mathcal{N}(\boldsymbol{V}^T) = \operatorname{span}\{\boldsymbol{n}_1\}$. So, since $z \in \mathcal{N}(\boldsymbol{K}^T) = \mathcal{N}(\boldsymbol{V}^T\boldsymbol{L}^T)$, it follows that $\boldsymbol{z}$ satisfies $\boldsymbol{L}^T\boldsymbol{z} = a\boldsymbol{n}_1$ for an appropriate scalar a. That is,

$$\boldsymbol{\tau} = \frac{1}{\Delta_1}\boldsymbol{e}_1 + a(\boldsymbol{L}^T)^{-1}\boldsymbol{n}_1$$

where a is chosen minimizing $\|\boldsymbol{\tau}\|^2$,

$$\|\boldsymbol{\tau}\|^2 = \frac{1}{\Delta_1}\|\boldsymbol{e}_1\|^2 + \frac{2a}{\Delta_1}\left\langle \boldsymbol{e}_1, (\boldsymbol{L}^T)^{-1}\boldsymbol{n}_1 \right\rangle + a^2\|(\boldsymbol{L}^T)^{-1}\boldsymbol{n}_1\|^2,$$

where $\langle\cdot,\cdot\rangle$ denotes the usual Euclidean inner product on $\mathbb{R}^r$. Thus a satisfies

$$\frac{2}{\Delta_1}\left\langle \boldsymbol{e}_1, (\boldsymbol{L}^T)^{-1}\boldsymbol{n}_1 \right\rangle + 2a\|(\boldsymbol{L}^T)^{-1}\boldsymbol{n}_1\|^2 = 0,$$

or

$$\begin{aligned} |a| &= \frac{|\langle \boldsymbol{e}_1, (\boldsymbol{L}^T)^{-1}\boldsymbol{n}_1\rangle|}{\Delta_1\|(\boldsymbol{L}^T)^{-1}\boldsymbol{n}_1\|^2} \\ &\le \frac{1}{\Delta_1\|(\boldsymbol{L}^T)^{-1}\boldsymbol{n}_1\|}. \end{aligned}$$

Since $1/\Delta_1 < \left[(\boldsymbol{L}^T)^{-1}\boldsymbol{n}_1\right]_1 \le \|(\boldsymbol{L}^T)^{-1}\boldsymbol{n}_1\|$, we have that $|a| < 1$.

In addition, it follows from (4.3) and the definition of $\boldsymbol{\tau}$ that $\|\boldsymbol{\tau}\| \le 1/\Delta_1$, and thus $|\tau_1| \le 1/\Delta_1$. But $\tau_1 = 1/\Delta_1 + a\left[(\boldsymbol{L}^T)^{-1}\boldsymbol{n}_1\right]_1$ where $\left[(\boldsymbol{L}^T)^{-1}\boldsymbol{n}_1\right]_1 > 0$, so it must be that $-1 < a \le 0$.

Finally, we show that $\alpha(\Delta r) = -a$. From (3.24),

$$\begin{aligned} \alpha(\Delta r) &= \sum_{l=2}^{r} \tau_l(\Delta_1 + \ldots + \Delta_{l-1}) \\ &= \left\langle \boldsymbol{\tau}, \boldsymbol{L}(0,1,\ldots,1)^T \right\rangle \\ &= \left\langle \frac{1}{\Delta_1}\boldsymbol{e}_1 + a(\boldsymbol{L}^T)^{-1}\boldsymbol{n}_1, \boldsymbol{L}(0,1,\ldots,1)^T \right\rangle \\ &= \left\langle \boldsymbol{e}_1 + a\boldsymbol{n}_1, (0,1,\ldots,1)^T \right\rangle \\ &= a\left\langle \boldsymbol{n}_1, (0,1,\ldots,1)^T \right\rangle. \end{aligned} \tag{4.4}$$

A simple combinatorial result gives $\langle \boldsymbol{n}_1, (1,\ldots,1)^T\rangle = 0$, from which it follows that

$$\begin{aligned} \alpha(\Delta r) &= -a\binom{d+1}{0} \\ &= -a, \end{aligned}$$

and the proof is complete. □

LEMMA 4.3. *Let $r = d+2$ for fixed $d = 0,1,\ldots$. Then there are constants $0 \le w(r) < 1$ and $M(r) > 0$ independent of h such that*

(i) $0 \le \alpha(\Delta r) \le w(r) < 1$, *and*
(ii) $\|h\boldsymbol{\tau}\|_1 \le M(r)$,
for all $h > 0$ sufficiently small. Here $\|\cdot\|_1$ denotes the usual vector 1-norm on $\mathbb{R}^r$.

Proof. We recall from (4.4) and (3.17) that

$$\begin{aligned} \alpha(\Delta r) &= (0,1,\ldots,1)\boldsymbol{L}^T\boldsymbol{\tau} \\ &= (0,1,\ldots,1)(\boldsymbol{K}^+\boldsymbol{L})^T(1,\ldots,1)^T \end{aligned} \tag{4.5}$$

where $(0,1,\ldots,1)^T \in \mathbb{R}^r$, $(1,\ldots,1)^T \in \mathbb{R}^{d+1}$, and where

$$\begin{aligned} \boldsymbol{K}^+\boldsymbol{L} &= (\boldsymbol{K}^T\boldsymbol{K})^{-1}\boldsymbol{K}^T\boldsymbol{L} \\ &= (\boldsymbol{V}^T\boldsymbol{L}^T\boldsymbol{L}\boldsymbol{V})^{-1}\boldsymbol{V}^T\boldsymbol{L}^T\boldsymbol{L}. \end{aligned} \tag{4.6}$$

The proof of the theorem will involve viewing the problem for k satisfying the assumptions of this section as a perturbation of the problem associated with $k(t) \equiv 1$. We first consider (i) and (ii) in the statement of the lemma for the special case of $k(t) \equiv 1$. We will denote the usual $r \times (d+1)$ $\boldsymbol{K}$-matrix associated with this particular kernel by $\boldsymbol{K}_1$. Similarly, we will use $\alpha_1(\Delta r)$, $\boldsymbol{\tau}_1$, $\boldsymbol{L}_1$, and $\boldsymbol{V}_1$ to designate $\alpha(\Delta r)$, $\boldsymbol{\tau}$, $\boldsymbol{L}$, and $\boldsymbol{V}$ for this kernel (where all of these quantities depend on h). In particular, we have for $\boldsymbol{L}_1 \in \mathbb{R}^{r\times r}$,

$$\boldsymbol{L}_1 = h\begin{pmatrix} 1 & & 0 \\ \vdots & \ddots & \\ 1 & \cdots & 1 \end{pmatrix}, \quad \boldsymbol{L}_1^{-1} = \frac{1}{h}\begin{pmatrix} 1 & & & 0 \\ -1 & 1 & & \\ & \ddots & \ddots & \\ 0 & & -1 & 1 \end{pmatrix}. \tag{4.7}$$

Let $\hat{\boldsymbol{L}}_1 = \frac{1}{h}\boldsymbol{L}_1$. Then from (4.6), we know that

$$\boldsymbol{K}_1^+\boldsymbol{L}_1 = (\boldsymbol{V}_1^T\hat{\boldsymbol{L}}_1^T\hat{\boldsymbol{L}}_1\boldsymbol{V}_1)^{-1}\boldsymbol{V}_1^T\hat{\boldsymbol{L}}_1^T\hat{\boldsymbol{L}}_1 \tag{4.8}$$

where $\hat{\boldsymbol{L}}_1$ and $\boldsymbol{V}_1$ are independent of h. It then follows from (4.5) that $\alpha_1(\Delta r) = C_1(r)$ for some constant $C_1(r)$ independent of $h > 0$. In addition, for the kernel $k(t) \equiv 1$ and any fixed $h > 0$, we have for $\boldsymbol{n}_1$ defined in (4.2),

$$\begin{aligned} \left[(\boldsymbol{L}_1^T)^{-1}\boldsymbol{n}_1\right]_1 &= (1/h)\left(\binom{d+1}{0} + \binom{d+1}{1}\right) \\ &\ge 2/h \\ &> 1/h \end{aligned} \tag{4.9}$$

so that the conditions of Lemma 4.2 hold. From this it follows that $0 \le C_1(r) < 1$.

To bound $h\boldsymbol{\tau}_1$, we note that

$$\begin{aligned} h\boldsymbol{\tau}_1 &= h(\boldsymbol{K}_1^+)^T(1,\ldots,1)^T \\ &= \hat{\boldsymbol{L}}_1\boldsymbol{V}_1(\boldsymbol{V}_1^T\hat{\boldsymbol{L}}_1^T\hat{\boldsymbol{L}}_1\boldsymbol{V}_1)^{-1}(1,\ldots,1)^T \\ &= \boldsymbol{w}_1(r) \end{aligned} \tag{4.10}$$

where $\boldsymbol{w}_1(r) \in \mathbb{R}^r$ is independent of $h > 0$.

Now we return to the case of a general k satisfying the hypotheses of this section. A Taylor expansion gives $k(t_i - s) = k(0) + g_i(s,h)$ where $|g_i(s,h)| = \mathcal{O}(h)$ for $i = 1, \dots, r$ and a.e. $s \in [0,h]$. Now for each $i = 1, \dots, r$, we have from (2.5) that

$$\begin{aligned} \Delta_i &= \int_0^{t_1} [1 + g_i(s,h)]\, ds \\ &= h(1 + \theta_i(h)), \end{aligned} \tag{4.11}$$

where θ_i is an appropriately chosen $\mathcal{O}(h)$ function for $i = 1, \dots, r$. For all h sufficiently small, $\Delta_i > 0$ for $i = 1, \dots, r$. The $r \times r$ $\boldsymbol{L}$-matrix for this kernel may thus be written $\boldsymbol{L} = \boldsymbol{L}_1 + h\boldsymbol{H}$, where $\boldsymbol{L}_1$ is defined in (4.7) and $\boldsymbol{H} \in \mathbb{R}^{r\times r}$ is given by

$$\boldsymbol{H} = \begin{pmatrix} \theta_1(h) & & 0 \\ \vdots & \ddots & \\ \theta_r(h) & \cdots & \theta_1(h) \end{pmatrix}. \tag{4.12}$$

Clearly $\|\boldsymbol{H}\|$ is $\mathcal{O}(h)$ as $h \to 0$ for fixed r.

Let $\hat{\boldsymbol{L}} = (1/h)\boldsymbol{L}$. Then $\hat{\boldsymbol{L}}^T = \hat{\boldsymbol{L}}_1^T + \boldsymbol{H}^T$, where $\hat{\boldsymbol{L}}_1 = (1/h)\boldsymbol{L}_1$ is independent of h. We may thus write

$$\hat{\boldsymbol{L}}^T = \hat{\boldsymbol{L}}_1^T(\boldsymbol{I} + \boldsymbol{F})$$

where $\boldsymbol{F} = (\hat{\boldsymbol{L}}_1^T)^{-1}\boldsymbol{H}^T$ is such that $\|\boldsymbol{F}\| = \mathcal{O}(h)$ as $h \to 0$. For all $h > 0$ sufficiently small we have $\|\boldsymbol{F}\| < 1$, so, using [25, page 188],

$$\|(\hat{\boldsymbol{L}}^T)^{-1}\boldsymbol{n}_1 - (\hat{\boldsymbol{L}}_1^T)^{-1}\boldsymbol{n}_1\| \le \frac{\|\boldsymbol{F}\|}{1 - \|\boldsymbol{F}\|}\|(\hat{\boldsymbol{L}}_1^T)^{-1}\boldsymbol{n}_1\|,$$

where the right-hand side of this inequality goes to zero with h. In particular,

$$[(\hat{\boldsymbol{L}}^T)^{-1}\boldsymbol{n}_1]_1 \stackrel{h\to 0}{\longrightarrow} [(\hat{\boldsymbol{L}}_1^T)^{-1}\boldsymbol{n}_1]_1$$

where from (4.9), $[(\hat{\boldsymbol{L}}_1^T)^{-1}\boldsymbol{n}_1]_1 \ge 2$. Since $h/\Delta_1 = h/[h(1+\theta_1(h))] = 1/(1+\theta_1(h))$, then for all $h > 0$ sufficiently small we have $[(\hat{\boldsymbol{L}}^T)^{-1}\boldsymbol{n}_1]_1 > h/\Delta_1$ or $[(\boldsymbol{L}^T)^{-1}\boldsymbol{n}_1]_1 > 1/\Delta_1$. Thus for all h sufficiently small we have $0 \le \alpha(\Delta r) < 1$ by Lemma 4.2.

To obtain the bound in (i) in the statement of the lemma, we use (4.5) to write

$$\begin{aligned} |\alpha(\Delta r) - \alpha_1(\Delta r)| &= |(0,1,\dots,1)(\boldsymbol{K}^+\boldsymbol{L} - \boldsymbol{K}_1^+\boldsymbol{L}_1)^T(1,\dots,1)^T| \\ &\le C\|\boldsymbol{K}^+\boldsymbol{L} - \boldsymbol{K}_1^+\boldsymbol{L}_1\| \end{aligned}$$

for some $C > 0$ independent of h. But

$$\begin{aligned} \|\boldsymbol{K}^+\boldsymbol{L} - \boldsymbol{K}_1^+\boldsymbol{L}_1\| &\le \|\boldsymbol{K}^+\boldsymbol{L} - \boldsymbol{K}^+\boldsymbol{L}_1\| + \|\boldsymbol{K}^+\boldsymbol{L}_1 - \boldsymbol{K}_1^+\boldsymbol{L}_1\| \\ &\le \|\boldsymbol{K}^+\|\,\|\boldsymbol{L} - \boldsymbol{L}_1\| + \|\boldsymbol{K}^+ - \boldsymbol{K}_1^+\|\,\|\boldsymbol{L}_1\| \\ &\le \|\boldsymbol{K}^+ - \boldsymbol{K}_1^+\|\,(\|\boldsymbol{L} - \boldsymbol{L}_1\| + \|\boldsymbol{L}_1\|) + \|\boldsymbol{K}_1^+\|\,\|\boldsymbol{L} - \boldsymbol{L}_1\|. \end{aligned}$$

Let $\boldsymbol{E} = \boldsymbol{K} - \boldsymbol{K}_1 = (\boldsymbol{L} - \boldsymbol{L}_1)\boldsymbol{V} = h\boldsymbol{H}\boldsymbol{V}$, where here we have used the fact that $\boldsymbol{V} = \boldsymbol{V}_1$. Then $\|\boldsymbol{E}\|$ is $\mathcal{O}(h^2)$ and we see from (4.8) that $\|\boldsymbol{K}_1^+\|$ is $\mathcal{O}(1/h)$. So, for

h sufficiently small, we have $\|\boldsymbol{E}\|\,\|\boldsymbol{K}_1^+\| < 1$. With this information, we may use a perturbation bound on the pseudoinverse [18, page 46] to get

$$\|\boldsymbol{K}^+ - \boldsymbol{K}_1^+\| \le \frac{\sqrt{2}\|\boldsymbol{E}\|\,\|\boldsymbol{K}_1^+\|^2}{1 - \|\boldsymbol{E}\|\,\|\boldsymbol{K}_1^+\|}$$

so that $\|\boldsymbol{K}^+ - \boldsymbol{K}_1^+\| \le M_1$ for all h sufficiently small. Thus for h small,

$$\|\boldsymbol{K}^+\boldsymbol{L} - \boldsymbol{K}_1^+\boldsymbol{L}_1\| \le M_1(h\|\boldsymbol{H}\| + \|\boldsymbol{L}_1\|) + h\|\boldsymbol{K}_1^+\|\,\|\boldsymbol{H}\|$$

where $\|\boldsymbol{L}_1\| = \mathcal{O}(h)$. It follows that $\|\boldsymbol{K}^+\boldsymbol{L} - \boldsymbol{K}_1^+\boldsymbol{L}_1\| \to 0$ as $h \to 0$, and that $\alpha(\Delta r) \to \alpha_1(\Delta r) = C_1(r) \in [0,1)$ as $h \to 0$. If we define

$$w(r) = \frac{1}{2}(1 + C_1(r)),$$

it follows that $0 < w(r) < 1$ and $0 \le \alpha(\Delta r) \le w(r)$ for all $h > 0$ sufficiently small. Thus the estimate in part (i) of the lemma is valid.

Turning to (ii) we observe that

$$\begin{aligned}\|\boldsymbol{\tau} - \boldsymbol{\tau}_1\| &= \|((\boldsymbol{K}^+)^T - (\boldsymbol{K}_1^+)^T)(1 \cdots 1)^T\| \\ &\le C_2\|\boldsymbol{K}^+ - \boldsymbol{K}_1^+\|\end{aligned}$$

for some constant $C_2 > 0$ independent of $h > 0$. Since $\|\boldsymbol{K}^+ - \boldsymbol{K}_1^+\| \le M_1$ for all $h > 0$ sufficiently small, we have

$$\|\boldsymbol{\tau}\|_1 \le C_3 M_1 + \|\boldsymbol{\tau}_1\|_1 \tag{4.13}$$

for some $C_3 > 0$ independent of $h > 0$ and so from (4.10),

$$\begin{aligned}h\|\boldsymbol{\tau}\|_1 &\le hC_3M_1 + \|\boldsymbol{w}_1(r)\|_1 \\ &\le M(r),\end{aligned}$$

where $M(r) \equiv 1 + \|\boldsymbol{w}_1(r)\|_1$ is a constant independent of h for all $h > 0$ sufficiently small. It follows that the bound in (ii) holds and the proof of the lemma is complete. □

Finally we return to the proof of the main theorem of this section which concerns the convergence of the future polynomial regularization method.

Proof of Theorem 4.1. Let $\Delta r = (r-1)h$ and consider the future polynomial equation (3.26) in c_i (which is equivalent to (2.12)–(2.14)), now using noisy data f^δ in place of f,

$$c_i\alpha(\Delta r) + \int_0^{t_i} \tilde{k}(t_i - s; \Delta r)\sum_{j=1}^{i} c_j\chi_j(s)\,ds = \tilde{f}^\delta(t_i; \Delta r), \tag{4.14}$$

for $i = 1, \ldots, N$. Subtracting (4.14) from a similar equation evaluated at t_{i+1} yields

$$\begin{aligned}&c_{i+1}\alpha(\Delta r) + \int_{t_i}^{t_{i+1}} \tilde{k}(t_{i+1} - s; \Delta r)c_{i+1}\chi_{i+1}(s)\,ds \\ &= \int_0^{\Delta r} [f^\delta(t_{i+1} + \rho) - f^\delta(t_i + \rho)]\,d\eta_{\Delta r}(\rho) \\ &- \sum_{\mu=1}^{i}\int_{t_{\mu-1}}^{t_\mu} \left[\tilde{k}(t_{i+1} - s; \Delta r) - \tilde{k}(t_i - s; \Delta r)\right] c_\mu\chi_\mu(s)\,ds \\ &+ c_i\alpha(\Delta r),\end{aligned} \tag{4.15}$$

for $i = 1, \ldots, N-1$.

We perform a similar differencing on an equation satisfied by the true solution $\bar{u}$. Since $\bar{u}$ satisfies (1.1) on $[0,T]$, we have that

$$\int_0^{t+\rho} k(t+\rho-s)\bar{u}(s)\,ds = f(t+\rho), \quad t \in (0,1],\ \rho \in [0,\Delta r], \tag{4.16}$$

so $\bar{u}$ also satisfies

$$\int_0^{\Delta r}\int_0^{t+\rho} k(t+\rho-s)\bar{u}(s)\,ds\,d\eta_{\Delta r}(\rho) = \int_0^{\Delta r} f(t+\rho)\,d\eta_{\Delta r}(\rho),$$

or, after splitting a term and changing the order of integration,

$$\begin{aligned} \int_0^t \tilde{k}(t-s;\Delta r)\bar{u}(s)\,ds &+ \int_0^{\Delta r}\int_0^{\rho} k(\rho-s)\bar{u}(t+s)\,ds\,d\eta_{\Delta r}(\rho) \\ &= \int_0^{\Delta r} f(t+\rho)\,d\eta_{\Delta r}(\rho), \quad t \in (0,1]. \end{aligned} \tag{4.17}$$

Evaluating (4.17) at $t = t_i$ and $t = t_{i+1}$ and subtracting the first from the second yields

$$\begin{aligned} &\alpha(\Delta r)\bar{u}(t_{i+1}) + \int_{t_i}^{t_{i+1}} \tilde{k}(t_{i+1}-s;\Delta r)\bar{u}(s)\,ds \\ &= \int_0^{\Delta r} [f(t_{i+1}+\rho) - f(t_i+\rho)]\,d\eta_{\Delta r}(\rho) \\ &\quad - \sum_{\mu=1}^{i}\int_{t_{\mu-1}}^{t_\mu} \left[\tilde{k}(t_{i+1}-s;\Delta r) - \tilde{k}(t_i-s;\Delta r)\right]\bar{u}(s)\,ds + \alpha(\Delta r)\bar{u}(t_i) \\ &\quad - \left\{\int_0^{\Delta r}\int_0^{\rho} k(\rho-s)\,[\bar{u}(t_{i+1}+s) - \bar{u}(t_{i+1})]\,ds\,d\eta_{\Delta r}(\rho)\right. \\ &\quad\quad \left. - \int_0^{\Delta r}\int_0^{\rho} k(\rho-s)\,[\bar{u}(t_i+s) - \bar{u}(t_i)]\,ds\,d\eta_{\Delta r}(\rho)\right\}, \end{aligned} \tag{4.18}$$

for $i = 1, \ldots, N-1$.

Finally, subtracting (4.15) from (4.18),

$$\begin{aligned} &\alpha(\Delta r)[\bar{u}(t_{i+1}) - c_{i+1}] + \int_{t_i}^{t_{i+1}} \tilde{k}(t_{i+1}-s;\Delta r)[\bar{u}(s) - c_{i+1}\chi_{i+1}(s)]\,ds \\ &= -\int_0^{\Delta r} [e(t_{i+1}+\rho) - e(t_i+\rho)]\,d\eta_{\Delta r}(\rho) \\ &\quad - \sum_{\mu=1}^{i}\int_{t_{\mu-1}}^{t_\mu} \left[\tilde{k}(t_{i+1}-s;\Delta r) - \tilde{k}(t_i-s;\Delta r)\right][\bar{u}(s) - c_\mu\chi_\mu(s)]\,ds \\ &\quad + \alpha(\Delta r)[\bar{u}(t_i) - c_i] \\ &\quad - \left\{\int_0^{\Delta r}\int_0^{\rho} k(\rho-s)\,[\bar{u}(t_{i+1}+s) - \bar{u}(t_{i+1})]\,ds\,d\eta_{\Delta r}(\rho)\right. \\ &\quad\quad \left. - \int_0^{\Delta r}\int_0^{\rho} k(\rho-s)\,[\bar{u}(t_i+s) - \bar{u}(t_i)]\,ds\,d\eta_{\Delta r}(\rho)\right\}, \end{aligned} \tag{4.19}$$

for $i = 1, \ldots, N-1$.

A Taylor expansion of $\bar{u}(t)$ at t_{i+1} yields

$$\bar{u}(t) - c_{i+1}\chi_{i+1}(t) = h\left[\frac{\bar{u}(t_{i+1}) - c_{i+1}}{h} + \frac{t - t_{i+1}}{h}\bar{u}'(\zeta_{i+1}(t))\right], \tag{4.20}$$

for $t \in (t_i, t_{i+1}]$, $i = 0, \ldots, N-1$, and $\zeta_{i+1}(t) \in (t, t_{i+1})$, $i = 0, \ldots, N-1$. We thus define

$$\beta_{i+1} = \frac{\bar{u}(t_{i+1}) - c_{i+1}}{h}, \tag{4.21}$$

for $i = 0, \ldots, N-1$, and using (4.20) and (4.21) in (4.19) we find (using arguments similar to those in [12]) that

$$\beta_{i+1} = W(r,h)\beta_i - h\sum_{\mu=1}^{i} Y_{i+1,\mu}(r,h)\beta_\mu - \frac{1}{h^2}E_{i+1}(\delta, r, h) - Z_{i+1}(r,h), \tag{4.22}$$

for $i = 1, \ldots, N-1$, where, for appropriate ψ-variables,

$$W(r,h) = \frac{\displaystyle\int_0^{\Delta r}\int_0^{\rho/h} k(\rho - sh)\, ds\, d\eta_{\Delta r}(\rho)}{d(r,h)},$$

and for $\mu = 1, \ldots, i,\ \ i = 1, \ldots, N-1$,

$$Y_{i+1,\mu}(r,h) = \frac{\displaystyle\int_0^1\int_0^{\Delta r} k'(\psi_{i+1,\mu,h}(s,\rho))\, d\eta_{\Delta r}(\rho)\, ds}{d(r,h)},$$

$$E_{i+1}(\delta, r, h) = \frac{\displaystyle\int_0^{\Delta r} \left[e(t_{i+1} + \rho) - e(t_i + \rho)\right] d\eta_{\Delta r}(\rho)}{d(r,h)},$$

$$Z_{i+1}(r,h) = \frac{z_{i+1}(r,h)}{d(r,h)}.$$

Here

$$d(r,h) = \int_0^{\Delta r}\int_0^{\rho/h} k(\rho - sh)\, ds\, d\eta_{\Delta r}(\rho) + \int_0^1\int_0^{\Delta r} k((1-s)h + \rho)\, d\eta_{\Delta r}(\rho)\, ds,$$

and, for $i = 1, \ldots, N-1$,

$$\begin{aligned}
z_{i+1}(r,h) = &\int_0^{\Delta r}\int_0^{\rho/h} k(\rho - sh)\bar{u}'(\psi_{i+1,h}(s))\, ds\, d\eta_{\Delta r}(\rho) \\
&- \int_0^{\Delta r}\int_0^{\rho/h} k(\rho - sh)\bar{u}'(\psi_{i+1,h})\, ds\, d\eta_{\Delta r}(\rho) \\
&+ h\sum_{\mu=1}^{i}\int_0^1\int_0^{\Delta r} k'(\psi_{i+1,\mu,h}(s,\rho))\, d\eta_{\Delta r}(\rho)(s-1)\bar{u}'(\zeta_\mu((s+\mu-1)h))\, ds \\
&+ \int_0^1\int_0^{\Delta r} k((1-s)h + \rho)\, d\eta_{\Delta r}(\rho)(s-1)\bar{u}'(\zeta_{i+1}((s+i)h))\, ds.
\end{aligned}$$

Equation (4.22) gives an expression for $\beta_2,\ldots,\beta_N$ in terms of $\beta_1,\ldots,\beta_{N-1}$. To determine β_1, we repeat a process similar to the above, evaluating equation (4.14) at $i=1$ and equation (4.17) at $t=t_1$ and subtracting, to obtain

$$\beta_1 = -\frac{1}{h^2}E_1(\delta,r,h) - Z_1(r,h), \tag{4.23}$$

where

$$E_1(\delta,r,h) = \frac{\int_0^{\Delta r} e(t_1+\rho)\,d\eta_{\Delta r}(\rho)}{d(r,h)}, \tag{4.24}$$

$$Z_1(r,h) = \frac{z_1(r,h)}{d(r,h)}, \tag{4.25}$$

$$\begin{aligned} z_1(r,h) = {}& \int_0^1\int_0^{\Delta r} k((1-s)h+\rho)(s-1)\bar{u}'(\zeta_1(sh))\,d\eta_{\Delta r}(\rho)\,ds \\ &+\int_0^{\Delta r}\int_0^{\rho/h} k(\rho-sh)s\bar{u}'(\zeta_1(h+sh))\,ds\,d\eta_{\Delta r}(\rho). \end{aligned} \tag{4.26}$$

If the following bounds can be obtained for h sufficiently small,

$$|W(r,h)| \le w(r), \tag{4.27}$$

$$|Y_{i,\mu}(r,h)| \le y(r),\ \mu=1,\ldots,i;\ i=2,\ldots,N, \tag{4.28}$$

$$|E_i(\delta,r,h)| \le \varepsilon(r)\delta,\ i=1,\ldots,N, \tag{4.29}$$

$$|Z_i(r,h)| \le z(r),\ i=1,\ldots,N, \tag{4.30}$$

where $y(r), z(r), \varepsilon(r) \ge 0$ and $0 \le w(r) < 1$, then the theorem is proved using an induction argument and difference equation techniques (as in [12]).

To this end we first simplify $d(r,h)$. We note that a change of integration variable in (3.21) gives

$$\alpha(\Delta r) = h\int_0^{\Delta r}\int_0^{\rho/h} k(\rho-sh)\,ds\,d\eta_{\Delta r}(\rho), \tag{4.31}$$

so that, using the definitions of $\eta_{\Delta r}$, Δ_i, and $\tilde{\Delta}_1$ in (3.22), (2.5), and (3.15), respectively,

$$\begin{aligned} d(r,h) &= (1/h)\left(\alpha(\Delta r) + \int_0^{\Delta r}\int_0^{t_1} k(t_1-s+\rho)\,ds\,d\eta_{\Delta r}(\rho)\right) \\ &= (1/h)\left(\alpha(\Delta r) + \sum_{i=1}^{r}\tau_i\int_0^{t_1} k(t_1-s+t_{i-1})\,ds\right) \\ &= (1/h)\left(\alpha(\Delta r) + \sum_{i=1}^{r}\tau_i\Delta_i\right) \\ &= (1/h)\left(\alpha(\Delta r) + \tilde{\Delta}_1\right). \end{aligned}$$

Thus we have from (3.25) that

$$d(r,h) = 1/h. \tag{4.32}$$

Similar steps give

$$W(r,h) = \alpha(\Delta r),$$

so that using Lemma 4.3 we have the existence of $w(r)$ with $0 \le w(r) < 1$ and $|W(r,h)| \le w(r)$ for h sufficiently small. Thus we obtain the required bound in (4.27).

To establish the bounds (4.28)–(4.30), we will make use of the constant $M(r)$ defined in Lemma 4.3. We have for $\mu = 1,\dots,i,\ i = 2,\dots,N$,

$$\begin{aligned} Y_{i+1,\mu}(r,h) &= h\int_0^1\int_0^{\Delta r} k'(\psi_{i+1,\mu,h}(s,\rho))\, d\eta_{\Delta r}(\rho)\, ds \\ &= h\sum_{j=1}^{r}\tau_j\int_0^1 k'(\psi_{i+1,\mu,h}(s,t_{j-1}))\, ds, \end{aligned}$$

so that $|Y_{i+1,\mu}(r,h)| \le h\|\boldsymbol{\tau}\|_1\|k'\|_\infty \le \|k'\|_\infty M(r)$ for $\mu = 1,\dots,i,\ i = 2,\dots,N$. We thus have the uniform bound in (4.28).

We obtain the bound in (4.29) in a similar manner, using the fact that

$$\begin{aligned} E_{i+1}(\delta,r,h) &= h\int_0^{\Delta r}[e(t_{i+1}+\rho) - e(t_i+\rho)]\, d\eta_{\Delta r}(\rho) \\ &= h\sum_{j=1}^{r}\tau_j[e(t_{i+1}+t_{j-1}) - e(t_i+t_{j-1})] \end{aligned}$$

for $i = 1,\dots,N-1$, so that $|E_{i+1}(\delta,r,h)| \le h\|\boldsymbol{\tau}\|_1 2\delta \le \varepsilon(r)\delta$ for $\varepsilon(r) \equiv 2M(r)$. A similar bound is obtained for $|E_1(\delta,r,h)|$.

Finally, for $i = 1,\dots,N-1$,

$$\begin{aligned} Z_{i+1}(r,h) = h\Bigg[&\sum_{j=1}^{r}\tau_j\int_0^{j-1} k(t_{j-1}-sh)\bar{u}'(\psi_{i+1,h}(s))\, ds \\ &- \sum_{j=1}^{r}\tau_j\int_0^{j-1} k(t_{j-1}-sh)\bar{u}'(\psi_{i+1,h})\, ds \\ &+ h\sum_{\mu=1}^{i}\int_0^1\sum_{j=1}^{r}\tau_j k'(\psi_{i+1,\mu,h}(s,t_{j-1}))(s-1)\bar{u}'(\zeta_\mu((s+\mu-1)h))\, ds \\ &+ \int_0^1\sum_{j=1}^{r}\tau_j k(t_{j-1}+(1-s)h)(s-1)\bar{u}'(\zeta_{i+1}((s+i)h))\, ds\Bigg], \end{aligned}$$

so that

$$|Z_{i+1}(r,h)| \le 2\alpha(\Delta r)\|\bar{u}'\| + ih\|k'\|\,\|\bar{u}'\|\,\|\boldsymbol{\tau}\|_1 h + \|k\|\,\|\bar{u}'\|\,\|\boldsymbol{\tau}\|_1 h.$$

Here we have used (4.31) to write $\alpha(\Delta r) = h\sum_{j=1}^{r}\tau_j\displaystyle\int_0^{j-1} k(t_{j-1}-sh)\, ds$. Thus $|Z_{i+1}(r,h)| \le z(r)$ for $i = 1,\dots,N-1$, where $z(r) = (2w(r) + (\|k\|+\|k'\|)M(r))\|\bar{u}'\|$. A similar calculation shows that $|Z_1(r,h)| \le z(r)$.

We thus have the required bounds in (4.27)–(4.30), from which the desired convergence follows. Using the results in [20, 21], the convergence rate ($\mathcal{O}(\delta^{1/2})$) is seen to be best possible with respect to the level δ of error in the data. □

5. Degenerate Cases. Recall from Theorem 3.1 that if the kernel k and parameters r and d are such that $\alpha(\Delta r) \neq 0$, we may view the future polynomial method as a discretization of a particular second-kind Volterra equation. Because such equations are stable with respect to perturbations in data, it is reasonable to expect the future polynomial method to do a better job of regularizing the original first-kind Volterra problem than would simple collocation alone. Indeed the numerical findings in the next section support this expectation. Though of importance, a complete analysis of the stability of the future polynomial method in the case of $\alpha(\Delta r) \neq 0$ is beyond the scope of this paper and will be presented elsewhere.

For the present we will not focus further on the $\alpha(\Delta r) \neq 0$ case, but rather will narrow our consideration to those degenerate cases in which the matrix equation (one of the equivalent formulations of the future polynomial method, as seen in §3),

$$(\tilde{\boldsymbol{A}} + \tilde{\alpha}\boldsymbol{I})\boldsymbol{c} = \tilde{\boldsymbol{f}}, \tag{5.1}$$

reduces precisely to the (unregularized) discretized equation formulated in §2 for the original problem, i.e., to

$$\boldsymbol{A}\boldsymbol{c} = \boldsymbol{f}. \tag{5.2}$$

The matrix $\boldsymbol{A} \in \mathbb{R}^{N\times N}$ and vector $\boldsymbol{f} \in \mathbb{R}^N$ are defined in (2.4), while $\tilde{\boldsymbol{A}} \in \mathbb{R}^{N\times N}$, $\tilde{\alpha} \in \mathbb{R}$, and $\tilde{\boldsymbol{f}} \in \mathbb{R}^N$ are defined for (5.1) via (3.13)–(3.17).

Our first result is that the future polynomial approximation equation (5.1) always reduces precisely to the original discretization equation (5.2) when $r = d+1$, for $d = 0, 1, \ldots$. This is expected because the future polynomial method regularizes by requiring (at least temporarily at each step) that $d+1$ future solution values lie along a polynomial of degree d. Because this process may be performed exactly for the case of $r = d+1$ without imposing any additional constraints on solution values, there is no resulting regularizing effect.

THEOREM 5.1. *Let $r = d+1$ for fixed $d = 0, 1, \ldots$ and assume $\boldsymbol{A}$ in (5.2) is nonsingular. Then the future polynomial equation (5.1) and the original discretization equation (5.2) are identical for all possible $\boldsymbol{f} \in \mathbb{R}^N$.*

Proof. When $r = d+1$, the matrix $\boldsymbol{V} \in \mathbb{R}^{r\times r}$ is invertible, and thus so is $\boldsymbol{K}^T$. Since $\boldsymbol{K}^T\boldsymbol{e}_1 = \Delta_1(1,\cdots,1)^T$, it follows that $\boldsymbol{\tau} = (\boldsymbol{K}^T)^+(1,\cdots,1)^T = (1/\Delta_1)\boldsymbol{e}_1$. Using (3.14)–(3.16) we thus have that $\tilde{\boldsymbol{A}} + \tilde{\alpha}\boldsymbol{I} = (1/\Delta_1)\boldsymbol{A}$ and $\tilde{\boldsymbol{f}} = (1/\Delta_1)\boldsymbol{f}$. ∎

Therefore, the more interesting degenerate cases are those that arise when $r \geq d+2$. In understanding these cases, it is helpful to observe that equation (5.2) need not only be associated with the collocation-based discretization of an ill-posed Volterra problem such as that considered in this paper, but can also arise as a discretization of a much more general problem. Indeed, all that is required in a generalized setting is that the discretized equation (5.2) be such that $\boldsymbol{A}$ is an arbitrary lower-triangular, Toeplitz matrix in $\mathbb{R}^{N\times N}$, and that $\boldsymbol{f}$ be an arbitrary vector in $\mathbb{R}^N$. Working from a given equation (5.2), the future polynomial method then constructs a well-defined perturbation (5.1) of this equation, where the quantities $\tilde{\boldsymbol{A}}$, $\tilde{\alpha}$, and $\tilde{\boldsymbol{f}}$, are defined using only r, d, and the entries in $\boldsymbol{A}$ and $\boldsymbol{f}$ (see, e.g., (3.13)–(3.17)).

Although this more general setting enlarges the number of problems to which we may want to apply the future polynomial regularization method, it also introduces situations in which regularization is simply not necessary. For example, the $\mathbb{R}^{N\times N}$

matrix

$$\hat{\boldsymbol{A}} = \frac{1}{h}\begin{pmatrix} 1 & & & 0 \\ -1 & 1 & & \\ & \ddots & \ddots & \\ 0 & & -1 & 1 \end{pmatrix}, \tag{5.3}$$

associated with a discretization of a differential operator, is a lower-triangular matrix in Toeplitz form which makes it a suitable choice of $\boldsymbol{A}$ in the approximating equation (5.2). We note that $\hat{\boldsymbol{A}} = (\boldsymbol{A}_0)^{-1}$, where $\boldsymbol{A}_0$ is the usual $\boldsymbol{A}$-matrix associated with the collocation-based discretization of the Volterra operator $\mathcal{A}_0 u(t) = \int_0^t u(s)\,ds$, $t \in [0,1]$. Because $\mathcal{A}_0$ is a smoothing operator, discretizations of the Volterra equation $\mathcal{A}_0 u = f$ require regularization in order to obtain a stable solutions as N gets large; however the same cannot be said of reasonable discretizations of the differential equation $\mathcal{A}_0^{-1} u = f$. Correspondingly, there is no reason to believe that use of the future polynomial method as given by equation (5.1) will offer an improvement over equation (5.2) when $\boldsymbol{A} = \hat{\boldsymbol{A}}$ appears in the latter. In fact, this is exactly the case, as Theorem 5.2 below shows: the two equations are identical for this particular choice of matrix, regardless of the choice of $\boldsymbol{f}$ in (5.2) and for all $d = 0, 1, \ldots$, and $r \geq d+2$. So there are examples of degenerate cases where (5.1) reduces exactly to (5.2) when $r \geq d+2$, but perhaps these degenerate cases are the ones where regularization is simply not needed.

The following theorem shows that the above example is actually representative of the general degenerate case for $r \geq d+2$. That is, the perturbed equation (5.1) associated with the future polynomial method is identical to equation (5.2) when $\boldsymbol{A}$ is a generalization of the matrix $\hat{\boldsymbol{A}}$ above, namely, when $\boldsymbol{A}$ is the inverse of a linear combination of discretizations $\boldsymbol{A}_j$ of well-understood Volterra operators, for $j = 0, \ldots, d$. Here the matrix $\boldsymbol{A}_j \in \mathbb{R}^{N\times N}$ denotes the usual collocation-based discretization (defined in §2) of the operator $\mathcal{A}_j$, where $\mathcal{A}_j$ is a canonical $(j+1)$-smoothing Volterra operator,

$$\mathcal{A}_j u(t) = \int_0^t (t-s)^j u(s)\,ds, \quad t \in [0,1], \tag{5.4}$$

for $j = 0, \ldots, d$, and $u \in L_2(0,1)$. Since we are only interested in applying the future polynomial method to discretizations of Volterra operators like those in (1.1)–(1.2) (i.e., to *smoothing* operators), it is thus unlikely that a discretization of such an operator will act like the inverse of other smoothing operators.

The exact statement of this result (providing necessary and sufficient conditions for degeneracies to occur) follows. After the theorem we state a corollary in which we show that, when applying the future polynomial method to problems of the type considered in Theorem 4.1 (the main convergence theorem), one will never encounter the situation of equation (5.1) reducing exactly to (5.2).

THEOREM 5.2. *Let $r \geq d+2$, for fixed $d = 0, 1, \ldots$, and assume that N is sufficiently large so that $N \geq r$. Let $\boldsymbol{A} \in \mathbb{R}^{N\times N}$ in (5.2) be nonsingular. Then the future polynomial equation (5.1) reduces exactly to equation (5.2) (for all possible $\boldsymbol{f}$ in (5.2)) if and only if there exist scalars $\gamma_0, \ldots, \gamma_d$ such that*

$$\boldsymbol{L}^{-1} = \sum_{j=0}^{d} \gamma_j \boldsymbol{L}_{r,j}, \tag{5.5}$$

where $\boldsymbol{L} \in \mathbb{R}^{r\times r}$ denotes the leading $r\times r$ submatrix of $\boldsymbol{A}$, and $\boldsymbol{L}_{r,j}$ is the leading $r\times r$ submatrix of $\boldsymbol{A}_j \in \mathbb{R}^{N\times N}$, for $j = 0,\dots,d$. (Here $\boldsymbol{A}_j$ is the usual collocation-based discretization of the $(j+1)$-smoothing Volterra operator $\mathcal{A}_j$ defined in (5.4).)

COROLLARY 5.3. *Let $r \ge d+2$ for $d = 0,1,\dots$. Assume $\mathcal{A}$ is a Volterra integral operator with kernel $k \in C^1[0,1]$, $k(0)\neq 0$. Then for all $h>0$ sufficiently small, the sequential future polynomial algorithm (2.12)–(2.14) cannot be equivalent to a simple collocation-based discretization (2.8)–(2.9) of equation (1.1).*

5.1. Proofs of main results in §5. The proofs of Theorem 5.2 and Corollary 5.3 will follow after we establish some preliminary technical results. We assume throughout that N is sufficiently large to ensure that $N \ge r$.

LEMMA 5.4. *Let $r \ge d+2$ for fixed $d = 0,1,\dots$, and in equation (5.2), assume that $\boldsymbol{A} \in \mathbb{R}^{N\times N}$ is nonsingular and that $\boldsymbol{f} \in \mathbb{R}^N$ is arbitrary. Then the future polynomial method equation (5.1) is identical to the original discretization equation (5.2) if and only if $\boldsymbol{\tau}$ (defined in (3.17)) satisfies $\boldsymbol{\tau} = c\,\boldsymbol{e}_1$, $c\neq 0$, for $\boldsymbol{e}_1 = (1,0,\dots,0)^T \in \mathbb{R}^r$.*

Proof. We will first show that $\boldsymbol{\tau} \in \operatorname{span}\{\boldsymbol{e}_1\}$ if and only if $\boldsymbol{e}_1 \in \mathcal{R}(\boldsymbol{K})$. To this end, first suppose that $\boldsymbol{\tau} \in \operatorname{span}\{\boldsymbol{e}_1\}$. We know that $\boldsymbol{\tau}$ is the minimum norm solution of

$$\boldsymbol{K}^T\boldsymbol{x} = (1,\dots,1)^T. \tag{5.6}$$

So $\boldsymbol{\tau} \in \mathcal{N}(\boldsymbol{K}^T)^\perp = \mathcal{R}(\boldsymbol{K})$ and it follows that $\boldsymbol{e}_1 \in \mathcal{R}(\boldsymbol{K})$.

Now suppose that $\boldsymbol{e}_1 \in \mathcal{R}(\boldsymbol{K})$, with $\boldsymbol{\phi} \in \mathbb{R}^{d+1}$ satisfying $\boldsymbol{K}\boldsymbol{\phi} = \boldsymbol{e}_1$. Since $\boldsymbol{K} = \boldsymbol{L}\boldsymbol{V}$ and $\boldsymbol{\tau}$ is the minimum norm solution of (5.6), we have

$$\boldsymbol{\tau} = \min\{\boldsymbol{x} \in \mathbb{R}^r \,|\, \boldsymbol{x} = (\boldsymbol{L}^T)^{-1}\boldsymbol{y},\ \boldsymbol{V}^T\boldsymbol{y} = (1,\dots,1)^T\}.$$

Further, since $\boldsymbol{V}^T\boldsymbol{e}_1 = (1,\dots,1)^T$, the solution to the equation $\boldsymbol{V}^T\boldsymbol{y} = (1,\dots,1)^T$ must lie in $\{\boldsymbol{e}_1 + \mathcal{N}(\boldsymbol{V}^T)\}$, where $\mathcal{N}(\boldsymbol{V}^T) = \operatorname{span}\{\boldsymbol{n}_1,\dots,\boldsymbol{n}_{r-(d+1)}\}$ for some $\boldsymbol{n}_i \in \mathbb{R}^r$, $i = 1,\dots,r-(d+1)$. Thus there is a unique choice of constants $a_1,\dots,a_{r-(d+1)}$ for which

$$\begin{aligned}\boldsymbol{\tau} &= (\boldsymbol{L}^T)^{-1}\left(\boldsymbol{e}_1 + \sum_{i=1}^{r-(d+1)} a_i\boldsymbol{n}_i\right)\\ &= \frac{1}{\Delta_1}\boldsymbol{e}_1 + \sum_{i=1}^{r-(d+1)} a_i(\boldsymbol{L}^T)^{-1}\boldsymbol{n}_i.\end{aligned}$$

It follows that

$$\boldsymbol{K}^T\boldsymbol{\tau} = \frac{1}{\Delta_1}\boldsymbol{K}^T\boldsymbol{e}_1 + \sum_{i=1}^{r-(d+1)} a_i\boldsymbol{V}^T\boldsymbol{L}^T(\boldsymbol{L}^T)^{-1}\boldsymbol{n}_i \tag{5.7}$$

$$= \frac{1}{\Delta_1}\boldsymbol{K}^T\boldsymbol{K}\boldsymbol{\phi} + \sum_{i=1}^{r-(d+1)} a_i\boldsymbol{V}^T\boldsymbol{n}_i \tag{5.8}$$

$$= \frac{1}{\Delta_1}\boldsymbol{K}^T\boldsymbol{K}\boldsymbol{\phi}. \tag{5.9}$$

Thus $\boldsymbol{K}\boldsymbol{K}^{+}\boldsymbol{\tau} = \boldsymbol{K}(\boldsymbol{K}^T\boldsymbol{K})^{-1}\boldsymbol{K}^T\boldsymbol{\tau} = (1/\Delta_1)\boldsymbol{K}\boldsymbol{\phi} = (1/\Delta_1)\boldsymbol{e}_1$. But $\boldsymbol{K}\boldsymbol{K}^{+}$ is the orthogonal projector onto $\mathcal{R}(\boldsymbol{K})$ and $\boldsymbol{\tau} \in \mathcal{R}(\boldsymbol{K})$. It follows that $\boldsymbol{\tau} = \boldsymbol{K}\boldsymbol{K}^{+}\boldsymbol{\tau} \in \operatorname{span}\{\boldsymbol{e}_1\}$, so we have proven the claim that $\boldsymbol{\tau} \in \operatorname{span}\{\boldsymbol{e}_1\}$ if and only if $\boldsymbol{e}_1 \in \mathcal{R}(\boldsymbol{K})$.

Now we turn to the proof of the theorem. Clearly if $\boldsymbol{\tau} = c\,\boldsymbol{e}_1$, $c \neq 0$, we have from (3.13)–(3.16) that equations (5.1) and (5.2) are identical for all possible $\boldsymbol{f} \in \mathbb{R}^N$. Now suppose that the two methods give the same results. From the above arguments, it is sufficient to show that the equivalence of the two methods implies that $\boldsymbol{e}_1 \in \mathcal{R}(\boldsymbol{K})$.

Let c_1 be the solution constant found after the first step of the sequential future polynomial algorithm (2.12)–(2.14), and s_1 be the solution constant found after one step of the simple collocation algorithm (2.8)–(2.9). We know from (3.11) that $c_1 = (1,\ldots,1)\boldsymbol{b}_1 = \boldsymbol{e}_1^T\boldsymbol{V}\boldsymbol{b}_1$ since the first row of $\boldsymbol{V}$ is $(1,\ldots,1)$. From (3.10), $\boldsymbol{b}_1 = \boldsymbol{K}^{+}\boldsymbol{f}_{1,r}$ and since $\boldsymbol{K}$ is of full rank, we have that $\boldsymbol{b}_1$ is the unique vector for which

$$\|\boldsymbol{K}\boldsymbol{b}_1 - \boldsymbol{f}_{1,r}\|^2 = \min_{\boldsymbol{b}\in\mathbb{R}^{d+1}} \|\boldsymbol{K}\boldsymbol{b} - \boldsymbol{f}_{1,r}\|^2.$$

Thus $\boldsymbol{K}\boldsymbol{b}_1 = \boldsymbol{L}\boldsymbol{V}\boldsymbol{b}_1 = P_{\boldsymbol{K}}\boldsymbol{f}_{1,r}$ where $P_{\boldsymbol{K}}$ denotes the orthogonal projection onto $\mathcal{R}(\boldsymbol{K})$. That is, $\boldsymbol{V}\boldsymbol{b}_1 = \boldsymbol{L}^{-1}P_{\boldsymbol{K}}\boldsymbol{f}_{1,r}$, and

$$c_1 = \boldsymbol{e}_1^T\boldsymbol{L}^{-1}P_{\boldsymbol{K}}\boldsymbol{f}_{1,r}.$$

Now consider s_1. We know that $s_1 = f_1/\Delta_1$, but, because $\boldsymbol{L}$ is lower triangular, we have that s_1 is also the first component of $\boldsymbol{L}^{-1}\boldsymbol{f}_{1,r}$, where $\boldsymbol{L}$ and $\boldsymbol{f}_{1,r}$ are as given above for the future polynomial algorithm. That is, $s_1 = \boldsymbol{e}_1^T\boldsymbol{L}^{-1}\boldsymbol{f}_{1,r}$. Thus, if the future polynomial method is equivalent to simple collocation, we must have $s_1 = c_1$ for all possible f (and so all possible $\boldsymbol{f}_{1,r}$); that is,

$$\boldsymbol{e}_1^T\boldsymbol{L}^{-1}(\boldsymbol{I} - P_{\boldsymbol{K}})\boldsymbol{f}_{1,r} = \boldsymbol{0}$$

for all $\boldsymbol{f}_{1,r} \in \mathbb{R}^r$. Equivalently, we must have $\boldsymbol{e}_1^T\boldsymbol{L}^{-1}\boldsymbol{g} = \boldsymbol{0}$ for all $\boldsymbol{g} \in \mathcal{R}(\boldsymbol{K})^{\perp}$. But $\boldsymbol{e}_1^T\boldsymbol{L}^{-1}\boldsymbol{g} = (1/\Delta_1)g_1 = \boldsymbol{e}_1^T(1/\Delta_1)\boldsymbol{g}$, equivalently we must have $\boldsymbol{e}_1^T\boldsymbol{g} = 0$ for all $\boldsymbol{g} \in \mathcal{R}(\boldsymbol{K})^{\perp}$, or $\boldsymbol{e}_1 \in \mathcal{R}(\boldsymbol{K})$.

Therefore, the equivalence of the two methods implies that $\boldsymbol{e}_1 \in \mathcal{R}(\boldsymbol{K})$ and the proof of the lemma is complete. ◻

LEMMA 5.5. *Let $d = 0, 1, \ldots$ and $r \geq d+2$, and let $\boldsymbol{L}_{r,d}$ be the $r \times r$ matrix*

$$\boldsymbol{L}_{r,d} = \begin{pmatrix} \bar{\Delta}_{1,d} & & 0 \\ \vdots & \ddots & \\ \bar{\Delta}_{r,d} & \cdots & \bar{\Delta}_{1,d} \end{pmatrix} \tag{5.10}$$

where, for $i = 1, \ldots, r$,

$$\bar{\Delta}_{i,d} = i^{d+1} - (i-1)^{d+1}. \tag{5.11}$$

Then $\boldsymbol{L}_{r,d}\boldsymbol{e}_1 \in \mathcal{R}(\boldsymbol{V}_{r,d})$, where $\boldsymbol{e}_1 \in \mathbb{R}^r$ and $\boldsymbol{V}_{r,d}$ is the usual $r \times (d+1)$ $\boldsymbol{V}$-matrix defined by

$$\boldsymbol{V}_{r,d} = \begin{pmatrix} 1 & 1 & \cdots & 1^d \\ 1 & 2 & \cdots & 2^d \\ \vdots & \vdots & & \vdots \\ 1 & r & \cdots & r^d \end{pmatrix}. \tag{5.12}$$

Proof. For $i = 2, 3, \ldots, r$,

$$
\begin{aligned}
[\boldsymbol{L}_{r,d}\boldsymbol{e}_1]_i &= i^{d+1} - (i-1)^{d+1} \\
&= (i - (i-1)) \sum_{j=0}^{d} i^{d-j}(i-1)^j
\end{aligned}
$$

where the last equality is obtained by canceling like terms in $\sum_{j=0}^{d} i^{d-j+1}(i-1)^j - \sum_{j=0}^{d} i^{d-j}(i-1)^{j+1}$. Thus,

$$
\begin{aligned}
[\boldsymbol{L}_{r,d}\boldsymbol{e}_1]_i &= \sum_{j=0}^{d} i^{d-j}(i-1)^j \\
&= \sum_{j=0}^{d} i^{d-j} \sum_{k=0}^{j} \binom{j}{k} i^k (-1)^{j-k} \\
&= \sum_{j=0}^{d} \sum_{k=0}^{j} \binom{j}{k} i^{d+k-j}(-1)^{j-k} \\
&= \sum_{l=0}^{d} i^{d-l}(-1)^l \sum_{m=0}^{d-l} \binom{l+m}{m} \\
&= \sum_{l=0}^{d} i^{d-l}(-1)^l \binom{d+1}{d-l}
\end{aligned}
$$

where we have used a combinatorial identity from [8, page 7, (1.49)].

Thus, for $i = 2, \ldots, r$,

$$
\begin{aligned}
[\boldsymbol{L}_{r,d}\boldsymbol{e}_1]_i &= \sum_{l=0}^{d} \binom{d+1}{l} (-1)^{d-l}\, i^l \\
&= \left[\boldsymbol{V}_{r,d}\boldsymbol{c}^{(d)}\right]_i
\end{aligned}
$$

where $\boldsymbol{c}^{(d)} \in \mathbb{R}^{d+1}$ is defined by

$$
\boldsymbol{c}^{(d)} = \left((-1)^d \binom{d+1}{0}, (-1)^{d-1}\binom{d+1}{1}, \ldots, (-1)^0 \binom{d+1}{d}\right)^T.
$$

We know by (5.11) that $[\boldsymbol{L}_{r,d}\boldsymbol{e}_1]_1 = 1$ and by inspection we see that $\left[\boldsymbol{V}\boldsymbol{c}^{(d)}\right]_1 = 1$, so the proof of the lemma is complete. ◻

LEMMA 5.6. *Let $d = 0, 1, \ldots,$ and $r \geq d+2$ be fixed. Then the set of vectors $\{\boldsymbol{L}_{r,0}\boldsymbol{e}_1, \ldots, \boldsymbol{L}_{r,d}\boldsymbol{e}_1\}$ is a basis for $\mathcal{R}(\boldsymbol{V}) = \mathcal{R}(\boldsymbol{V}_{r,d})$.*

Proof. From Lemma 5.5 we know that there is a $\boldsymbol{c}^{(d)} \in \mathbb{R}^{d+1}$ such that $\boldsymbol{L}_{r,d}\boldsymbol{e}_1 = \boldsymbol{V}_{r,d}\boldsymbol{c}^{(d)}$. Similarly, for $k = 0, 1, \ldots, d$, there is a $\boldsymbol{c}^{(k)} \in \mathbb{R}^{k+1}$ such that $\boldsymbol{L}_{r,k}\boldsymbol{e}_1 =$

$\boldsymbol{V}_{r,k}\boldsymbol{c}^{(k)}$. That is,

$$\begin{aligned}
\boldsymbol{L}_{r,k}\boldsymbol{e}_1 &= \begin{pmatrix} 1 & 1 & \cdots & 1^k \\ 1 & 2 & \cdots & 2^k \\ \vdots & \vdots & & \vdots \\ 1 & r & \cdots & r^k \end{pmatrix} \boldsymbol{c}^{(k)} \\
&= \begin{pmatrix} 1 & 1 & \cdots & 1^k & 1^{k+1} & \cdots & 1^d \\ 1 & 2 & \cdots & 2^k & 2^{k+1} & \cdots & 2^d \\ \vdots & \vdots & & \vdots & \vdots & & \vdots \\ 1 & r & \cdots & r^k & r^{k+1} & \cdots & r^d \end{pmatrix} \begin{pmatrix} \boldsymbol{c}^{(k)} \\ 0 \\ \vdots \\ 0 \end{pmatrix} \\
&= \boldsymbol{V}_{r,d}\tilde{\boldsymbol{c}}^{(k)}
\end{aligned}$$

for $k = 0, 1, \ldots, d$, where $\tilde{\boldsymbol{c}}^{(k)} = (\boldsymbol{c}^{(k)} \,|\, 0, \ldots, 0)^T \in \mathbb{R}^{d+1}$. Thus, $\boldsymbol{L}_{r,k}\boldsymbol{e}_1 \in \mathcal{R}(\boldsymbol{V}_{r,d})$ for $k = 0, 1, \ldots, d$. Further,

$$\begin{aligned}
&(\boldsymbol{L}_{r,0}\boldsymbol{e}_1 \,|\boldsymbol{L}_{r,1}\boldsymbol{e}_1 \,| \cdots |\boldsymbol{L}_{r,d}\boldsymbol{e}_1) \\
&= \begin{pmatrix} 1^1 - 0^1 & 1^2 - 0^2 & \cdots & 1^{d+1} - 0^{d+1} \\ 2^1 - 1^1 & 2^2 - 1^2 & \cdots & 2^{d+1} - 1^{d+1} \\ \vdots & \vdots & & \vdots \\ r^1 - (r-1)^1 & r^2 - (r-1)^2 & \cdots & r^{d+1} - (r-1)^{d+1} \end{pmatrix},
\end{aligned}$$

and we see by elementary row operations that this matrix has the same rank as a Vandermonde matrix. Since $\boldsymbol{V}_{r,d}$ has rank $d+1$, the proof of the lemma is complete. □

We now return to the proofs of Theorem 5.2 and Corollary 5.3.

Proof of Theorem 5.2. It follows from Lemmas 5.4 and 5.6 that equations (5.1) and (5.2) are identical for all $\boldsymbol{f} \in \mathbb{R}^N$ if and only if $\boldsymbol{e}_1 \in \mathcal{R}(\boldsymbol{K})$, where $\mathcal{R}(\boldsymbol{K}) = \mathcal{R}(\boldsymbol{L}\boldsymbol{V}_{r,d}) = \operatorname{span}\{\boldsymbol{L}\boldsymbol{L}_{r,0}\boldsymbol{e}_1, \ldots, \boldsymbol{L}\boldsymbol{L}_{r,d}\boldsymbol{e}_1\}$, i.e., if and only if there exist real scalars $\gamma_0, \ldots, \gamma_d$ such that

$$\boldsymbol{e}_1 = \sum_{j=0}^{d} \gamma_j \boldsymbol{L}\boldsymbol{L}_{r,j}\boldsymbol{e}_1 \tag{5.13}$$

$$= \left(\sum_{j=0}^{d} \gamma_j \boldsymbol{L}_{r,j} \right) \boldsymbol{L}\boldsymbol{e}_1 \tag{5.14}$$

where we have used the fact that lower-triangular Toeplitz matrices commute. Because $\left(\sum_{j=0}^{d} \gamma_j \boldsymbol{L}_{r,j}\right)$ is a lower triangular Toeplitz matrix with diagonal entries given by $\sum_{j=0}^{d} \gamma_j$, it follows from the definition (3.7) of $\boldsymbol{L}$ that the first entry of $\left(\sum_{j=0}^{d} \gamma_j \boldsymbol{L}_{r,j}\right)\boldsymbol{L}\boldsymbol{e}_1$ is $\Delta_1 \sum_{j=0}^{d} \gamma_j$. So if there are scalars $\gamma_0, \ldots, \gamma_d$ such that $\boldsymbol{e}_1 = \left(\sum_{j=0}^{d} \gamma_j \boldsymbol{L}_{r,j}\right) \boldsymbol{L}\boldsymbol{e}_1$, it must be true that $\sum_{j=0}^{d} \gamma_j = 1/\Delta_1 \neq 0$ and the lower-triangular Toeplitz matrix $\sum_{j=0}^{d} \gamma_j \boldsymbol{L}_{r,j}$ is nonsingular.

Thus equation (5.1) reduces exactly to (5.2) if and only if there exist $\gamma_0, \ldots, \gamma_d$ with $\sum_{j=0}^{d} \gamma_j = 1/\Delta_1$ and $\left(\sum_{j=0}^{d} \gamma_j \boldsymbol{L}_{r,j}\right)^{-1} \boldsymbol{e}_1 = \boldsymbol{L}\boldsymbol{e}_1$. Since both $\left(\sum_{j=0}^{d} \gamma_j \boldsymbol{L}_{r,j}\right)^{-1}$

and $\boldsymbol{L}$ are lower triangular Toeplitz matrices, knowing the first column is the same as knowing the whole matrix. So the two methods are equivalent if and only if $\boldsymbol{L} = \left(\sum_{j=0}^{d} \gamma_j \boldsymbol{L}_{r,j}\right)^{-1}$ for some $\gamma_0, \ldots, \gamma_d$ with $\sum_{j=0}^{d} \gamma_j = 1/\Delta_1$. □

Proof of Corollary 5.3. Without loss of generality we let $k(0) = 1$ and use (4.11) to write $\Delta_i = h(1 + \theta_i(h))$, where $\theta_i = \mathcal{O}(h)$ for $i = 1, \ldots, r$. From this we get that the usual $\boldsymbol{L}$-matrix associated with the future polynomial method is given by $\boldsymbol{L} = \boldsymbol{L}_1 + h\boldsymbol{H}$, where $\boldsymbol{H} = \boldsymbol{H}(h)$ is defined in (4.12), with $\|\boldsymbol{H}\| = \mathcal{O}(h)$, and where $\boldsymbol{L}_1$ is defined by (4.7).

It follows from Theorem 5.2 that if the future polynomial method reduces to simple collocation, then there is a choice of $\gamma_0, \ldots, \gamma_d$ for which $\boldsymbol{L}\left(\sum_{j=0}^{d} \gamma_j \boldsymbol{L}_{r,j}\right)\boldsymbol{e}_1 = \boldsymbol{e}_1$ (where $\boldsymbol{L}_{r,j}$ is defined in (5.10)). This yields a system of r equations in $d+1$ unknowns,

$$\boldsymbol{e}_1^T \boldsymbol{L}\left(\sum_{j=0}^{d} \gamma_j \boldsymbol{L}_{r,j}\right)\boldsymbol{e}_1 = 1 \tag{5.15}$$

$$\boldsymbol{e}_k^T \boldsymbol{L}\left(\sum_{j=0}^{d} \gamma_j \boldsymbol{L}_{r,j}\right)\boldsymbol{e}_1 = 0, \quad k = 2, \ldots, r. \tag{5.16}$$

We note that $\boldsymbol{e}_i^T(\sum_{j=0}^{d} \gamma_j \boldsymbol{L}_{r,j})\boldsymbol{e}_1 = \sum_{j=0}^{d} \gamma_j \bar{\Delta}_{i,j}$ so that

$$\begin{aligned}
\boldsymbol{e}_k^T \boldsymbol{L}\left(\sum_{j=0}^{d} \gamma_j \boldsymbol{L}_{r,j}\right)\boldsymbol{e}_1 &= \sum_{i=1}^{k} \Delta_{k-i+1}\left(\sum_{j=0}^{d} \gamma_j \bar{\Delta}_{i,j}\right) \\
&= h\sum_{i=1}^{k}(1 + \theta_{k-i+1}(h))\left(\sum_{j=0}^{d} \gamma_j \left(i^{j+1} - (i-1)^{j+1}\right)\right) \\
&= h\sum_{j=0}^{d} \gamma_j \sum_{i=1}^{k}(i^{j+1} - (i-1)^{j+1}) \\
&\quad + h\sum_{j=0}^{d} \gamma_j \sum_{i=1}^{k} \theta_{k-i+1}(h)\left(i^{j+1} - (i-1)^{j+1}\right) \\
&= h\sum_{j=0}^{d} k^{j+1}\gamma_j + h\sum_{j=0}^{d} \hat{H}_{k,j}\gamma_j,
\end{aligned}$$

where $\hat{H}_{k,j} = \sum_{i=1}^{k} \theta_{k-i+1}(h)(i^{j+1} - (i-1)^{j+1})$, for $k = 1, \ldots, r,\ j = 0, \ldots, d$.

Taking together only the $k = 2$ through $k = d+2$ equations in (5.16), we have a system of $d+1$ equations in the vector $\boldsymbol{\gamma} = (\gamma_0, \ldots, \gamma_d)^T \in \mathbb{R}^{d+1}$,

$$(\hat{\boldsymbol{V}} + \hat{\boldsymbol{H}})\boldsymbol{\gamma} = \boldsymbol{0} \tag{5.17}$$

where $\hat{\boldsymbol{V}}$, $\hat{\boldsymbol{H}} \in \mathbb{R}^{(d+1)\times(d+1)}$ with $\hat{\boldsymbol{H}} = (\hat{H}_{k,j})$ and

$$\hat{\boldsymbol{V}} = \begin{pmatrix} 2 & 2^2 & \cdots & 2^{d+1} \\ 3 & 3^2 & \cdots & 3^{d+1} \\ \vdots & \vdots & & \vdots \\ d+2 & (d+2)^2 & \cdots & (d+2)^{d+1} \end{pmatrix}.$$

Defining the nonsingular diagonal matrix $\boldsymbol{D} = \mathrm{diag}(2, 3, \ldots, (d+2))$, we have that $\hat{\boldsymbol{V}} = \boldsymbol{D}\tilde{\boldsymbol{V}}$, where $\tilde{\boldsymbol{V}}$ is a $(d+1) \times (d+1)$ Vandermonde matrix of full rank. Thus $\hat{\boldsymbol{V}}$ is nonsingular. Further, since r and d are fixed, $\|\hat{\boldsymbol{H}}\| = \mathcal{O}(h)$ as $h \to 0$, so for all h sufficiently small, $(\hat{\boldsymbol{V}} + \hat{\boldsymbol{H}})$ is nonsingular and $\boldsymbol{\gamma} = \boldsymbol{0}$ in (5.17).

But if we now consider (5.15), it follows that

$$\sum_{j=0}^{d} \gamma_j = 1/(1 + \theta_1(h)) = 1/\Delta_1, \tag{5.18}$$

a contradiction with the fact that $\boldsymbol{\gamma} = \boldsymbol{0}$. Thus the future polynomial method cannot be equivalent to simple collocation for the operator $\mathcal{A}$ given in the statement of the corollary. $\square$

6. Numerical Results. We give below some numerical examples illustrating the performance of the future polynomial regularization method with regard to the capture of sharp or discontinuous features in the true solution. In each of the examples, the kernel k and the true solution $\bar{u}$ were selected *a priori*. The data function f was then generated by integration, $f(t) = \int_0^t k(t-s)\bar{u}(s)\,ds$. Uniformly distributed random noise in the interval $[-z\|f\|_\infty, z\|f\|_\infty]$, for some $z > 0$, was then added to the function f to produce f^δ. Noise levels with a relative error of 1% are common in applications; the noise levels in the examples range up to 2%. In each example the kernel is $k(t) = 1 + t^2$, the true solution $\bar{u}$ is graphed as a dashed line, and the regularized solution is graphed as a solid line determined by connecting the midpoints of the piecewise constant functions $c_i\chi_i$ determined by each method.

We note that, in order to implement the future polynomial regularization method, it is necessary to compute $(\mathbf{K}^T)^+(1, \ldots, 1)^T$. If r or d is large, then the $(d+1) \times r$ matrix $\mathbf{K}^T$ may be quite ill-conditioned, requiring additional regularization methods (e.g., truncation of singular values, etc.) in the computation of this quantity. However, one of the advantages of a "local regularization" approach such as we take here is that one is typically able to keep both r and d small, thus avoiding excessive ill-conditioning. In the examples that follow, no additional regularization was used in the numerical process.

Example 1. The true solution in this example is $\bar{u} = 0.6 + 0.5\cos(4\pi t)$. There are $N = 20$ collocation points. The noise added to f is at the level of 1% relative error. The results are shown in Figure 6.1. The true solution $\bar{u}$ is smooth in this case to demonstrate that the future polynomial method stabilizes comparably to the future constant method (labeled as "Beck" in the figures) when there are no sharp features or discontinuities, but resists oversmoothing the true solution.

Example 2. Here we compare simple collocation, the future constant method

("Beck"), and the future polynomial method on a non-smooth function,

$$\bar{u}(t) = \begin{cases} .2, & 0 \le t < .2 \\ 8t - 1.4, & .2 \le t < .3 \\ -8t + 3.4, & .3 \le t < .4 \\ .2, & .4 \le t \le 1 \end{cases} . \tag{6.1}$$

We increase the number of collocation points to $N = 40$ and increase the relative error to approximately 2%. The output is shown in Figure 6.2. The lack of stability is evident in the highly oscillatory behavior of the solution provided by simple collocation. For each $d = 0, 1, 2$ we select the r that minimizes $\sum_{i=1}^{N} |\bar{u}(t_i) - u_h(t_i)|^2$ where u_h is the solution provided by each regularization method. While the future constant method stabilizes the solution, it also oversmooths the sharp spike. The future polynomial method stabilizes the solution while still preserving the sharp features in the true solution.

Example 3. Finally, we consider a discontinuous true solution $\bar{u}$. Again, let $N = 40$ and the relative error added to the true data be approximately 2%. Let

$$\bar{u}(t) = \begin{cases} .2, & 0 \le t < .2, .3 \le t < .6, .7 \le t \le 1 \\ 1, & .2 \le t < .3, .6 \le t < .7 \end{cases} . \tag{6.2}$$

We again compare collocation with the future constant method ("Beck") and the future polynomial method selecting r as in Example 2. In this case as well, the future polynomial method stabilizes the problem while capturing the peaks in the true solution.

7. Conclusions. In this paper, we have described a new discrete regularization method for ill-posed first-kind Volterra problems. The future polynomial regularization method is easily implemented and preserves the causal, Volterra nature of the problem, allowing sequential solution in near real time. Because this method is based on a local regularization strategy (in contrast to classical methods such as Tikhonov regularization), it also has the potential for better resolving sharp features in solutions than is typically the case for classical methods. We have shown that the future polynomial method generalizes a class of regularization methods studied in [12, 13, 14]. We note that other recent generalizations may be found in [15] (a sequential Tikhonov regularization method for Volterra problems) and [16, 17] (a future constant method with variable $r = r(t)$ regularization parameter and penalty parameter $\mu = \mu(t)$, with an adaptive procedure for the sequential selection of μ).

We have presented a convergence theory for an important class of these methods and have shown convergence proceeds at the best possible rate with respect to the error in the data when applied to C^1 kernels k with $k(0) \neq 0$. We have also characterized the degenerate cases in which the method cannot be expected to regularize better than simple collocation alone, but have seen that the conditions for degeneracy are unlikely to occur when the method is applied to problems of the type considered in this paper. In particular, if k is a C^1 kernel with $k(0) \neq 0$, then we have proven that the future polynomial method cannot degenerate to simple collocation. Finally, we have presented numerical results which provide evidence that the method works well to recover sharp or discontinuous features in the true solution.

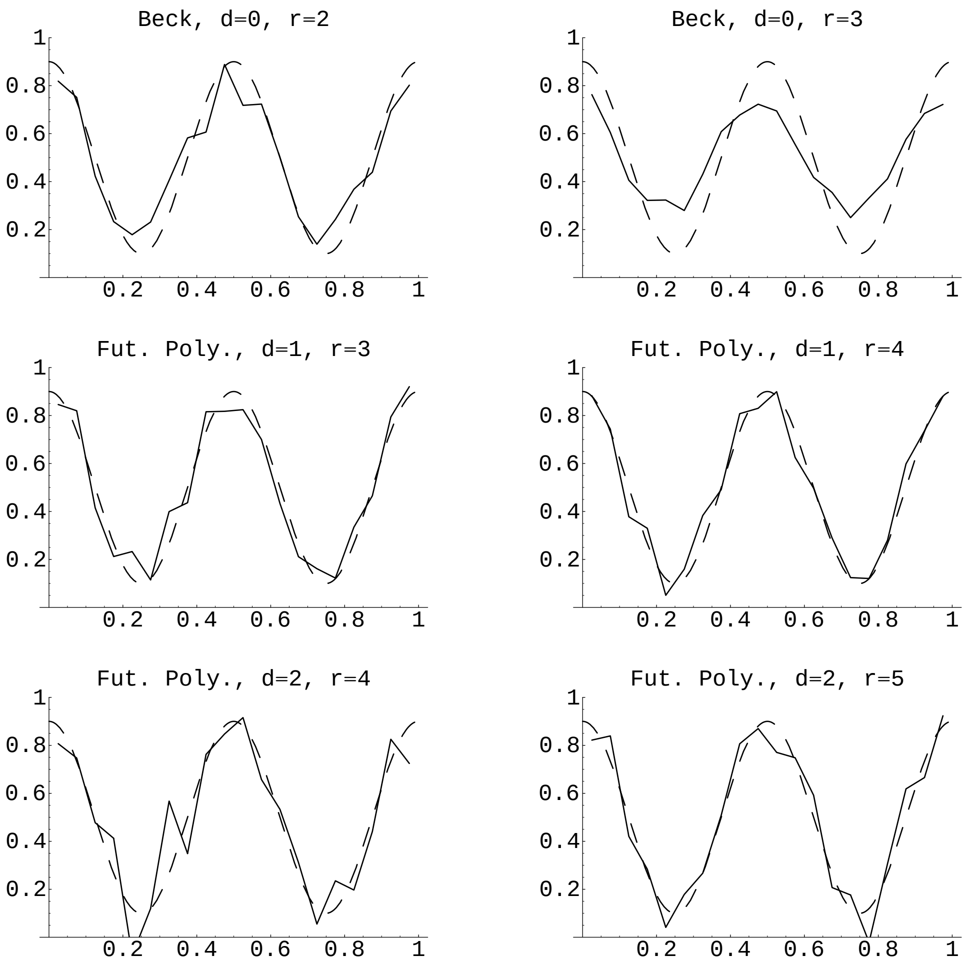

FIG. 6.1. *Example 1*

Current and future investigations include a stability analysis of the discrete equations obtained using the future polynomial method; in this case a condition number analysis (along the lines of [14]) could give much insight. In addition, we are currently investigating an extension of the main convergence theorem, Theorem 4.1, to general ν-smoothing kernels, as well as a convergence theory for the fully-continuous version (3.18) of this regularization method. The proof of Theorem 4.1 does not readily extend to the theoretical case of $\bar{u} \notin C^1$ (although numerical results seem to indicate good results with non-smooth $\bar{u}$), but we are currently looking at this issue. Finally, the selection of the appropriate regularization parameters is an important problem. It is hoped that we may eventually use local discrepancy principles to develop an adaptive regularization method, one in which the regularization parameters r and d are adaptively determined throughout the domain of the solution.

REFERENCES

[1] R. ACAR AND C. VOGEL, *Analysis of bounded variation penalty methods for ill-posed problems*,

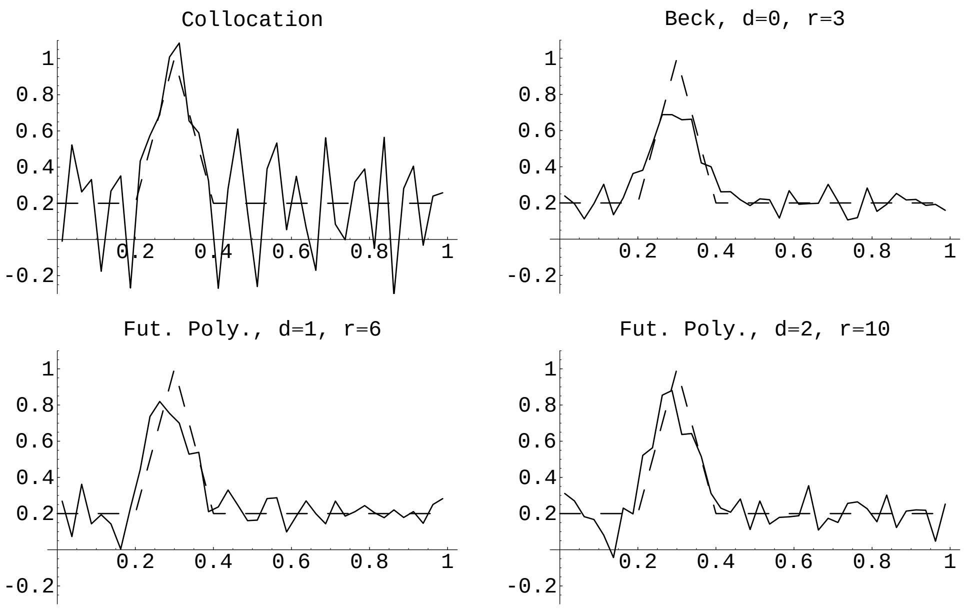

FIG. 6.2. *Example 2*

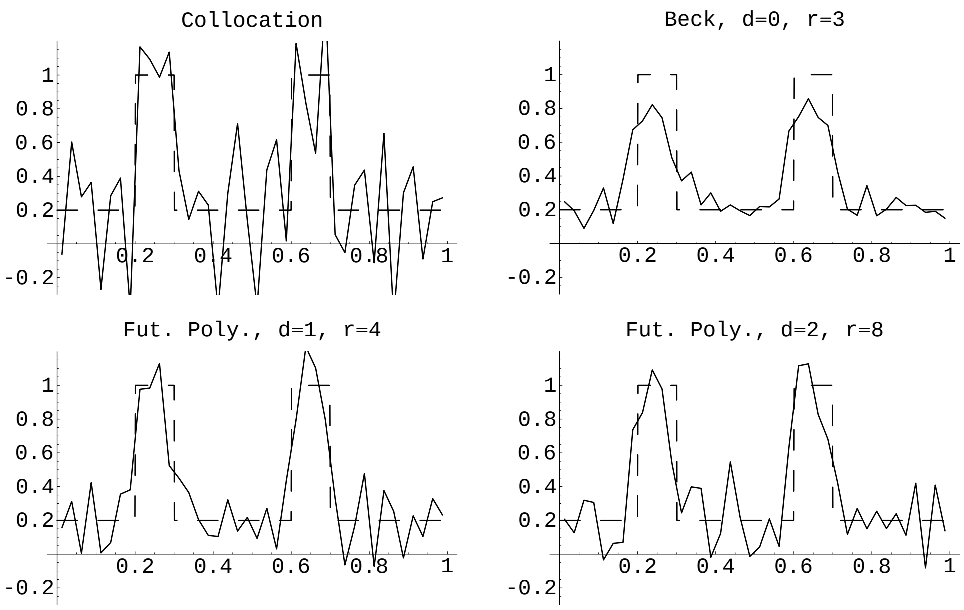

FIG. 6.3. *Example 3*

Inverse Problems, 10 (1994), pp. 1217–1229.

[2] J. V. BECK, B. BLACKWELL, AND C. R. ST. CLAIR, JR., *Inverse Heat Conduction*, Wiley-Interscience, New York, 1985.

[3] H. BRUNNER AND P. J. VAN DER HOUWEN, *The Numerical Solution of Volterra Equations*, North-Holland Publishing Co., Amsterdam-New York, 1986.

[4] A. CHAMBOLLE AND P.-L. LIONS, *Image recovery via total variation minimization and related problems*, Numer. Math., 72 (1997), pp. 167–188.

[5] P. G. Ciarlet, J. L. Lions, eds., *Handbook of Numerical Analysis, Vol. 1*, North-Holland, Amsterdam, New York, 1992.

[6] A. C. Cinzori, *Future Polynomial Regularization of Ill-Posed Volterra Problems*, Ph.D. thesis, Michigan State University, East Lansing, MI, 1998.

[7] D. C. Dobson and C. R. Vogel, *Convergence of an iterative method for total variation denoising*, SIAM J. Numerical Analysis, 34 (1997), pp. 1779-1971.

[8] H. W. Gould, *Combinatorial Identities*, Morgantown Printing and Binding Co., Morgantown, WV, 1972.

[9] G. Gripenberg, S. O. Londen, and O. Staffens, *Volterra Integral and Functional Equations*, Cambridge University Press, Cambridge, New York, 1990.

[10] K. Ito and K. Kunisch, *An active set strategy based on the augmented Lagrangian formulation for image restoration*, R.A.I.R.O. Math. Mod. and Numer. Anal., to appear.

[11] ———, *Augmented Lagrangian formulation of nonsmooth convex optimization in Hilbert spaces*, in Control of Partial Differential Equations and Applications, Proceedings of the IFIP TC7/WG-7.2 International Conference, Laredo, Spain, 1994, Lect. Notes Pur Appl. Math. 174, E. Casas, ed., Dekker, New York, 1996, pp. 107–117.

[12] P. K. Lamm, *Approximation of ill-posed Volterra problems via predictor-corrector regularization methods*, SIAM J. Appl. Math., 56 (1996), pp. 524–541.

[13] ———, *Future-sequential regularization methods for ill-posed Volterra equations*, J. Math. Analysis Appl., 195 (1995), pp. 469–494.

[14] ———, *Regularized inversion of finitely smoothing volterra operators: predictor-corrector regularization methods*, Inverse Problems, 13 (1997), pp. 375–402.

[15] P. K. Lamm and L. Eldén, *Numerical solution of first-kind Volterra equations by sequential Tikhonov regularization*, SIAM J. Numerical Analysis, 34 (1997), pp. 1432–1450.

[16] P. K. Lamm and T. L. Scofield, *Local regularization methods for the stabilization of ill-posed Volterra problems*, 1999, submitted.

[17] ———*Sequential predictor-corrector methods for variable regularization of ill-posed Volterra problems*, 1999, submitted.

[18] C. L. Lawson and R. J. Hanson, *Solving Least Squares Problems*, Prentice-Hall, Inc., Englewood Cliffs, NJ, 1974.

[19] P. Linz, *Analytical and numerical methods for Volterra problems*, SIAM, Philadelphia, PA, 1985.

[20] F. Natterer, *On the order of regularization methods*, in Improperly Posed Problems and their Numerical Treatment, G. Hämmerlin and K. H. Hoffman, eds., Birkhäuser, Basel, 1983, pp. 189–203.

[21] ———, *Error bounds for Tikhonov regularization in Hilbert scales*, Applicable Analysis, 18 (1984), pp. 29–37.

[22] A. Neubauer and O. Scherzer, *Reconstruction of discontinuous solutions from blurred data*, in Computational, Experimental, and Numerical Methods for Solving Ill-Posed Inverse Imaging Problems: Medical and Nonmedical Applications, R. L. Barbour, M. J. Carvlin and M. A. Fiddy, eds., ,Vol. 3171 of Proceedings of SPIE, SPIE, Washington, 1997, pp. 34–41.

[23] ———, *Regularization for curve representations: uniform convergence for discontinuous solutions of ill-posed problems*, SIAM J. Appl. Math., 58 (1998), pp. 1891–1900.

[24] L. I. Rudin, S. Osher, and E. Fatemi, *Nonlinear total variation based noise removal algorithms*, Physica D, 60 (1992), pp. 259–268.

[25] J. Stoer and R. Bulirsch, *Introduction to Numerical Analysis, 2nd edition*, Springer-Verlag, Berlin, New York, 1993.

[26] G. Stolz, Jr., *Numerical solutions to an inverse problem of heat conduction for simple shapes*, J. Heat Transfer, 82 (1960) pp. 20–26.

[27] A. N. Tikhonov and V. Y. Arsenin, *Solutions of Ill-Posed Problems*, V. H. Winston & Sons, New York, 1977.