

# An Augmented Galerkin Algorithm for First Kind Integral Equations of Hammerstein Type

S. Abbasbandy

and

E. Babolian

Institute of Mathematics,  
University of Teacher Education,  
Tehran 15614, IRAN

## Abstract

Recent papers, [1],[2] & [3], describe some algorithms for linear first kind integral equations. These algorithms are based on augmented Galerkin method and Cross-validation scheme [5]. The results show that, these algorithms work well for linear equations.

In this paper we apply algorithms of [1] & [2] on non-linear first kind integral equations of Hammerstein type with bounded solution. In order to obtain a posteriori error estimate, we apply fifteen-point Gauss-Kronrod quadrature rule [4]. Finally, we give a number of numerical examples showing that the algorithms work well in practice.

### 1. Introduction

We consider numerical solution of non-linear first kind integral equation of Hammerstein type

$$\int_{-1}^1 k(x,y) F(y) dy = g(x), \quad -1 \leq x \leq 1, \tag{1}$$

where

$$F(y) = \varphi(y, f(y)), \tag{2}$$

and  $k, g$  and  $\varphi$  are known functions and  $f$  is the unknown function which is assumed to be bounded, i.e.,

$$f_d \leq f(x) \leq f_u, \quad -1 \leq x \leq 1.$$

We can approximate  $F$  by

$$F_N(x) = \sum_{i=0}^N a_i T_i(x), \tag{3}$$

where  $T_i(x)$  is the Chebyshev polynomial of the first kind of degree  $i$ . We obtain  $a_0, \dots, a_N$  by one of the algorithms in [1] or [2].

$$\begin{aligned} & \underset{\mathbf{a}}{\text{Minimize}} \quad \|\mathbf{B}\mathbf{a} - \mathbf{g}\| \\ & \text{s.t.} \quad |a_i| \leq \delta_i = C_f \tilde{\tau}^{-i}, \\ & \quad \quad i = 0, 1, \dots, N, \\ & \quad \quad \tilde{\tau} = \max(1, i), \end{aligned}$$

where the elements of the coefficient matrix  $\mathbf{B}$  are given by

$$B_{ij} = \int_{-1}^1 \int_{-1}^1 \frac{k(x,y) T_i(x) T_j(y)}{(1-x^2)^{1/2}} dx dy, \quad i, j = 0, 1, \dots, N,$$

and the elements of the vector  $\mathbf{g}$  are

$$g_i = \int_{-1}^1 g(x) T_i(x) / (1-x^2)^{1/2} dx,$$

$$i = 0, 1, \dots, N,$$

$$\mathbf{a} = (a_0, a_1, \dots, a_N)^t,$$

and  $C_f$  and  $r$  are regularization parameters. Here we impose  $r > 0.5$  and set heuristically, and for  $C_f$ ,

$$C_f = \lambda \|\mathbf{g}\|_\infty / \|\mathbf{B}\|_\infty,$$

where  $\lambda$  must be set heuristically. There are some strategies in [1] for choosing  $r$  and  $\lambda$ .

Suppose  $P$  is a partition of  $[-1, 1]$ ,

$$-1 = x_0 < x_1 < \dots < x_{n-1} < x_n = 1,$$

where  $x_i = -1 + i/50$  for  $i \in I = \{0, 1, \dots, 100\}$  and  $n = 100$ .

Let  $Q$  be the partition of  $[-1, 1]$  corresponding to fifteen points of Gauss-Kronrod nodes [4], we denoted  $Q$  by

$$q_j, \quad j \in J = \{0, 1, \dots, 14\}.$$

### 2. The basic method

This method is pointwise, let  $\bar{x}$  be an arbitrary point in  $[-1, 1]$ , we approximate  $f(\bar{x})$ . Obviously, if we approximate  $f$  at Gauss-Chebyshev points, we would be able to approximate  $f$  by expansion method [1] & [2]. Here, we approximate  $f$  on  $Q$ , i.e., we evaluate approximate values of  $f_j =$

$f(q_j)$  for  $j \in J$ . Sometimes, to obtain more accuracy, we must approximate  $f$  on more points, see example 4.

From (2) and (3)

$$\varphi(q_j, f_j) - F_N(q_j) \approx 0, \quad j \in J.$$

If  $\varphi(x, y)$  has inverse with respect to  $y$ , we can compute  $f_j$  by

$$f_j = \varphi^{-1}(q_j, F_N(q_j)), \quad j \in J,$$

otherwise by knowing lower and upper bounds of  $f$  we continue as follows. For computing  $f_j$ , we may solve the following optimization problem

$$\begin{aligned} & \text{Minimize } \phi_j(x), \\ & f_d \leq x \leq f_u \end{aligned} \quad (4)$$

where

$$\phi_j(x) = (\varphi(q_j, x) - F_N(q_j))^2, \quad j \in J.$$

To solve (4), we apply Brent method which uses a combination of the golden section search and successive parabolic interpolation [4].

### 3. A posteriori error

By approximating  $f$  on  $Q$ , we can compute

$$\int_{-1}^1 k(x_i, y) \varphi(y, f(y)) dy, \quad i \in I,$$

by Gauss-Kronrod quadrature rule,  $u_i$ , which has a posteriori error estimate,  $v_i$ , [4].

Let

$$e_1 = \left[ \sum_{i=0}^{100} \{u_i - g(x_i)\}^2 / 101 \right]^{\frac{1}{2}},$$

$$e_2 = \left[ \sum_{i=0}^{100} v_i^2 / 101 \right]^{\frac{1}{2}}.$$

Usually, it is expected that  $e_1 \leq e_2$ .

Let  $z_j$  be the optimum value of (4),  $j \in J$ . If (1) has any solution, it is expected to have  $z_j \approx 0$ . Let

$$e_3 = \left[ \sum_{j=0}^{14} z_j^2 / 15 \right]^{\frac{1}{2}}.$$

It is expected that if  $e_1 \not\leq e_2$  for almost all  $N$ , and if  $e_3$  is not negligible, (1) has no solutions. Obviously, if we use composite Gauss-Kronrod quadrature rule, the definition of  $e_1$ ,  $e_2$  and  $e_3$  will change and their values become smaller.

### 4. Numerical examples and results.

We consider a set of six examples. All computations were carried out on an IBM-PC using C language and long double precision.

#### 4.1. Examples

1)

$$\begin{aligned} & \int_{-\pi}^{\pi} (\sin(x) + \cos(y))(1 + f(y))^2 dy \\ & = 3\pi \sin(x), \quad -\pi \leq x \leq \pi, \end{aligned}$$

with solution  $f(x) = \sin(x)$ .

2)

$$\int_0^1 k(x, y) f^2(y) dy = \frac{x^4 - x}{12}, \quad 0 \leq x \leq 1,$$

where

$$k(x, y) = \begin{cases} y(x - 1), & y < x, \\ x(y - 1), & x \leq y, \end{cases}$$

with solutions  $f(x) = \pm x$ .

3)

$$\int_0^x \cos(x - y)(1 + f(y))^2 dy = \sin(x),$$

$$0 \leq x \leq 2,$$

with solution  $f(x) = 1$ .

4)

$$\int_{-1}^x \ln(x - y) f^2(y) dy = g(x), \quad -1 \leq x \leq 1,$$

where

$$g(x) = \ln(1 + x) \frac{1 + x^3}{3} - \frac{11x^3 + 6x^2 - 3x + 2}{18},$$

with solutions  $f(x) = \pm x$ .

5)

$$\int_0^1 e^{xy} \sqrt{f(y)} dy = \frac{2}{x^3} - \frac{2e^x}{x^3} + \frac{2e^x}{x^2} - \frac{e^x}{x},$$

$$0 \leq x \leq 1,$$

with no solutions.

6)

$$\int_{-1}^0 e^{xy} \sqrt{f(y)} dy = \frac{(x + 1)e^{-x} - 1}{x^2},$$

$$-1 \leq x \leq 0,$$

with no solutions.

Results for the above examples are presented in tables 1-6, respectively.

Table 1 (Example 1)

$r = 5, \lambda = 4$

| N  | $e_1$   | $e_2$  | $e_3$   |
|----|---------|--------|---------|
| 2  | 3.4E-12 | 3.3E-8 | 1.1E-23 |
| 3  | 3.4E-12 | 3.3E-8 | 1.1E-23 |
| 4  | 1.0E-9  | 3.3E-8 | 5.7E-21 |
| 5  | 3.4E-12 | 3.3E-8 | 1.1E-23 |
| 6  | 3.4E-12 | 3.3E-8 | 1.1E-23 |
| 7  | 3.4E-12 | 3.3E-8 | 1.1E-23 |
| 8  | 3.4E-12 | 3.3E-8 | 1.1E-23 |
| 9  | 3.4E-12 | 3.3E-8 | 1.1E-23 |
| 10 | 3.4E-12 | 3.3E-8 | 1.1E-23 |

Table 2 (Example 2)

$r = 5, \lambda = 4$

| N  | $e_1$   | $e_2$   | $e_3$   |
|----|---------|---------|---------|
| 2  | 1.1E-10 | 4.0E-10 | 7.1E-21 |
| 3  | 1.1E-10 | 4.0E-10 | 7.1E-21 |
| 4  | 1.1E-10 | 4.0E-10 | 7.1E-21 |
| 5  | 1.1E-10 | 4.0E-10 | 7.1E-21 |
| 6  | 1.1E-10 | 4.0E-10 | 7.1E-21 |
| 7  | 1.1E-10 | 4.0E-10 | 7.1E-21 |
| 8  | 1.1E-10 | 4.0E-10 | 7.1E-21 |
| 9  | 1.1E-10 | 4.0E-10 | 7.1E-21 |
| 10 | 1.1E-10 | 4.0E-10 | 7.1E-21 |

Table 3 (Example 3)

$$r = 4, \lambda = 4$$

| N  | $e_1$   | $e_2$  | $e_3$   |
|----|---------|--------|---------|
| 2  | 1.0E-3  | 2.5E-4 | 1.9E-7  |
| 3  | 1.1E-6  | 1.3E-7 | 3.3E-10 |
| 4  | 1.2E-8  | 5.9E-9 | 1.8E-16 |
| 5  | 1.6E-11 | 5.9E-9 | 2.6E-20 |
| 6  | 9.1E-12 | 5.9E-9 | 1.4E-22 |
| 7  | 9.1E-12 | 5.9E-9 | 1.4E-22 |
| 8  | 9.1E-12 | 5.9E-9 | 1.4E-22 |
| 9  | 9.1E-12 | 5.9E-9 | 1.4E-22 |
| 10 | 9.1E-12 | 5.9E-9 | 1.4E-22 |

Table 5 (Example 5)

$$r = 2, \lambda = 5$$

| N  | $e_1$ | $e_2$   | $e_3$  |
|----|-------|---------|--------|
| 2  | 1.028 | 4.5E-14 | 4.4E-1 |
| 3  | 1.028 | 1.2E-6  | 4.4E-1 |
| 4  | 1.028 | 4.5E-14 | 4.4E-1 |
| 5  | 1.028 | 4.5E-14 | 4.4E-1 |
| 6  | 1.028 | 4.5E-14 | 4.4E-1 |
| 7  | 1.028 | 1.9E-4  | 4.4E-1 |
| 8  | 1.028 | 4.5E-14 | 4.4E-1 |
| 9  | 1.030 | 3.0E-3  | 4.5E-1 |
| 10 | 1.030 | 3.0E-3  | 4.5E-1 |

Table 4 (Example 4)

$$r = 4, \lambda = 4$$

| N  | $e_1$  | $e_2$  | $e_3$   |
|----|--------|--------|---------|
| 2  | 2.4E-2 | 4.5E-2 | 3.4E-21 |
| 3  | 1.2E-2 | 4.9E-2 | 9.0E-5  |
| 4  | 2.0E-4 | 4.7E-2 | 3.2E-21 |
| 5  | 2.0E-4 | 4.7E-2 | 3.2E-21 |
| 6  | 2.0E-4 | 4.7E-2 | 3.2E-21 |
| 7  | 2.0E-4 | 4.7E-2 | 3.2E-21 |
| 8  | 2.0E-4 | 4.7E-2 | 3.2E-21 |
| 9  | 2.0E-4 | 4.7E-2 | 3.2E-21 |
| 10 | 2.0E-4 | 4.7E-2 | 3.2E-21 |

Table 6 (Example 6)

$$r = 2, \lambda = 5$$

| N  | $e_1$ | $e_2$   | $e_3$  |
|----|-------|---------|--------|
| 2  | 1.470 | 4.5E-14 | 5.2E-1 |
| 3  | 1.470 | 4.5E-14 | 5.2E-1 |
| 4  | 1.470 | 4.5E-14 | 5.2E-1 |
| 5  | 1.470 | 4.5E-14 | 5.2E-1 |
| 6  | 1.470 | 4.5E-14 | 5.2E-1 |
| 7  | 1.470 | 4.5E-14 | 5.2E-1 |
| 8  | 1.470 | 4.5E-14 | 5.2E-1 |
| 9  | 1.470 | 4.5E-14 | 5.2E-1 |
| 10 | 1.470 | 4.5E-14 | 5.2E-1 |

#### 4.2. Comments

In all examples, it is assumed  $f_d = -1$ ,  $f_u = 1$ . The results in tables 1-4 show that, approximate solutions are very accurate, values of  $e_3$  are very small and  $e_1 \leq e_2$  for almost all  $N$ , but tables 5 and 6 show that, these examples have no solutions.

Example 4 is a Volterra singular integral equation and to obtain accurate result, we used a composite Gauss-Kronrod quadrature rule with 5 panels for each interval.

#### 4.3. Conclusion

From the results, we conclude that automatic augmented Galerkin method work well for linear and Hammerstein first kind integral equations. The cost of operations is not high. We can also compute a posteriori error estimate and use it as an indication of accuracy of approximate solution. Also with knowing the inverse of  $\varphi$ , we can apply this method without knowing lower and upper bounds of  $f$ .

**Acknowledgment.** We thank Prof. Du Jin-Yuan for thoroughly reading first draft of this paper and suggesting valuable comments.

## References

- [1] Babolian E. & Delves L.M. "An augmented Galerkin method for first

kind Fredholm equations", *J. Inst. Maths. Applics.* 24 (1979), 157-174.

- [2] Delves L.M. & Abd-Elal L.F. & Hendry J.A. "A Fast Galerkin Algorithm for Singular Integral Equations", *J. Inst. Maths. Applics.* 23 (1979), 139-166.
- [3] Essah W.A. & Delves L.M. "The Numerical Solution of First Kind Integral Equations", *J. Comput. Appl. Math.* 27 (1989), 363-387.
- [4] Kahaner D. & Moler C. & Nash S. "Numerical Methods and Software", Prentice-Hall, London, 1989.
- [5] Wahba G. "Practical Approximate Solutions to Linear Operator Equations When the Data are Noisy", *SIAM J. Numer. Anal.* 14 (1977) 651-667.