

Expansion methods for solving integral equations with multiple time lags using Bernstein polynomial of the second kind

M. Paripour^{a*}, Z. Shojaei^b and S. Abdolahi^c

^a*Department of Mathematics, Hamedan University of Technology, Hamedan, 65156-579, Iran.*

^b*Department of Mathematics, Lorestan University, Khoramabad, Iran.*

^c*Department of Mathematics, Arak Branch, Islamic Azad University, Arak, Iran.*

Received 26 May 2014; revised 3 August 2014; accepted 20 August 2014.

Abstract. In this paper, the Bernstein polynomials are used to approximate the solutions of linear integral equations with multiple time lags (IEMTL) through expansion methods (collocation method, partition method, Galerkin method). The method is discussed in detail and illustrated by solving some numerical examples. Comparison between the exact and approximated results obtained from these methods is carried out.

© 2014 IAUCTB. All rights reserved.

Keywords: Integral equation with multiple time lags; Expansion methods; Bernstein polynomial.

2010 AMS Subject Classification: 45G10; 45J05.

1. Introduction

One of the most important and applicable subjects of applied mathematics, and developing modern mathematics is the integral equations. The names of many modern mathematicians like Volterra, Fredholm, and Cauchy are associated with this topic [1]. The name integral equation was introduced by Boissier-Reymond in 1888 [2]. The most recent kind of equation that is worth studying is the delay integral equation. These equations have many applications including: a model to explain the observed periodic outbreak of certain infectious disease [3]. Another application is the discontinuous change in conductivity [4]. Bernstein polynomials have been recently used to solve some linear and non-linear differential equations, both partial and ordinary, by Bhatta and Bhatti

*Corresponding author.

E-mail address: paripour@hut.ac.ir, paripour@gmail.com (M. Paripour).

[5] and Bhatti and Bracken [6]. Also, these have been used to solve some classes of integral equations of both first and second kinds, by Mandal and Bhattacharya [7]. In this paper, we have developed a very simple method to solve Volterra integral equations of both first and second kinds and have regular as well as weakly singular kernels, using Bernstein polynomials. To facilitate, a brief review of some background on the linear integral equations with multiple time lags and their types is given in the following section.

2. linear integral equations with multiple time lags

The significance of these equations lies in their ability to describe processes with retarded (delay) time which may appear in the function $u(t)$ involved in the integrand or may appear on the left side of the equation or in one of the limits of the integrations [8]. The linear integral equations with multiple time lags (IEMTL) have two lags τ_1 and τ_2 such that $\tau_1, \tau_2 \in R$, τ_1 and $\tau_2 > 0$ and they can be classified into the following cases: The τ_1 appears in the unknown function $u(t)$ inside the integral sign such that:

$$h(t)u(t - \tau_1) = g(t) + \int_a^{b(t)} k(t, x)u(x - \tau_2)dx.$$

The τ_1 appears in the unknown function $u(t)$ outside the integral sign and τ_2 appears in one of the limits of integration such that:

$$h(t)u(t) = g(t) + \int_a^{\tau_2} k(t, x)u(x - \tau_1)dx,$$

or

$$h(t)u(t) = g(t) + \int_{\tau_2}^{b(t)} k(t, x)u(x - \tau_2)dx.$$

The τ_1 appears in the unknown function $u(t)$ inside the integral sign and τ_2 appears in one of the limits of integration:

$$h(t)u(t) = g(t) + \int_a^{\tau_2} k(t, x)u(x - \tau_1)dx,$$

or

$$h(t)u(t) = g(t) + \int_{\tau_2}^{b(t)} k(t, x)u(x - \tau_2)dx.$$

Where $h(t)$ and $g(t)$ are unknown functions of t , and $k(t, x)$ is called the kernel of the IEMTL. Remarks [8]

- If $h(t) = 0$, then the above equations are called IEMTL of the first kind.
- If $h(t) = 1$, then the above equations are called IEMTL of the second kind.
- If $g(t) = 0$, then the above equations are called homogeneous IEMTL; otherwise, if $g(t) \neq 0$, then the above equations are called nonhomogeneous IEMTL.

- If $b(t) = t$, then the above equation is called Volterra integral equation with multiple time lags while if $b(t) = b$, b is a constant, then the above equation is called Fredholm integral with multiple time lags.

3. Expansion Methods

Expansion methods or weighted residual methods are presented [8] by considering the following functional equation:

$$L[u(t)] = g(t), \quad t \in D, u \in U, g \in G, \quad (1)$$

Where L denotes an operator which maps a set of functions, say U , into a set of functions, say G , such that $u \in U$, $g \in G$ and D is a prescribed domain. The epitome of the expansion method is to approximate the unknown solution $u(t)$ of eq. (1) by a set of known functions as:

$$u(t) \cong u_N(t) = \sum_{i=0}^N c_i \phi_i(t), \quad (2)$$

where $N > 0$ and c_0, c_1, \dots, c_N are $N + 1$ unknown coefficients. The function $\phi_i(t)$ is chosen in this work to be Bernstein polynomial which is prescribed in section (4). An approximated solution $u_N(t)$ given by eq. (2) will not in general, satisfy eq. (1) exactly; therefore, a term, say $E_N(t)$, called the residual $E_N(t)$ depends on t as well as on the way that the parameters c_i 's are chosen. It is obvious that when $E_N(t) = 0$, the exact solution is obtained which is difficult to be achieved; therefore, we shall try to minimize $E_N(t)$ in some sense. In the expansion method, the unknown parameters c_i 's are chosen to minimize the residual $E_N(t)$ by setting weighted integral equal to zero, i.e.

$$\int_D w_j E_N(t) dt = 0; \quad j = 0, \dots, N, \quad (3)$$

where w_j is a prescribed weighting function, $t \in D$ and D is a prescribed domain. The technique based on eq. (3) is called weighted residual method. Different choices of w_j yield different approximate solutions. The expansion methods that will be discussed in this work are collocation, partition and Gherkin methods.

4. Bernstein polynomial

Polynomials are incredibly useful mathematical tools as they are simply defined, can be calculated quickly on computer systems and represent a tremendous variety of functions [5-7]. They can be differentiated and integrated easily, and can be pieced together to form spline curves that can approximate any function to any accuracy desired. Most students are introduced to polynomials at a very early stage in their studies of mathematics, and would probably recall them in the form below:

$$p(t) = a_n t^n + a_{n-1} t^{n-1} + \dots + a_1 t + a_0,$$

which represents a polynomial as a linear combination of certain elementary polynomials $\{1, t, t^2, \dots, t^n\}$. In general, any polynomial function that has degree less than or equal to n , can be written in this way, and the reasons are simply

- The set of polynomials of degree less than or equal to n forms a vector space: polynomials can be added together and can be multiplied by a scalar, therefore all the vector space properties hold.
- The set of functions $\{1, t, t^2, \dots, t^n\}$ form a basis for this vector space that is, any polynomial of degree less than or equal to n can be uniquely written as a linear combinations of these functions.

This basis, commonly called the power basis, is only one of the infinite number of bases for the space of polynomials. In this work, the choice of basis functions $\phi_i(t)$ is Bernstein polynomial.

The Bernstein polynomial can be defined as:

$$B_{i,n}(t) = \binom{n}{i} t^i (1-t)^{n-i}, \quad i = 0, 1, \dots, n,$$

that

$$\binom{n}{i} = \frac{n!}{i!(n-i)!}.$$

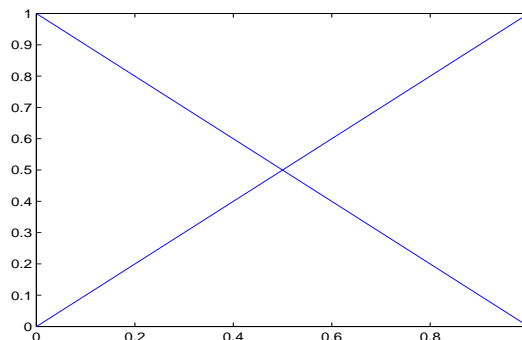
There are $n + 1$ n th-degree Bernstein polynomials. For mathematical convenience, we usually set $B_{i,n} = 0$, if $i < 0$ or $i > n$. These polynomials are quite easy to write down: the coefficients $\binom{n}{i}$ can be obtained from Pascal's triangle; the exponents on the t term increase by one as i increases; and the exponents on the $1 - t$ term decrease by one as i increases. In the simple cases, we obtain

- The Bernstein polynomials of degree 1 are

$$B_{0,1}(t) = 1 - t,$$

$$B_{1,1}(t) = t,$$

and can be plotted for $0 \leq t \leq 1$ as



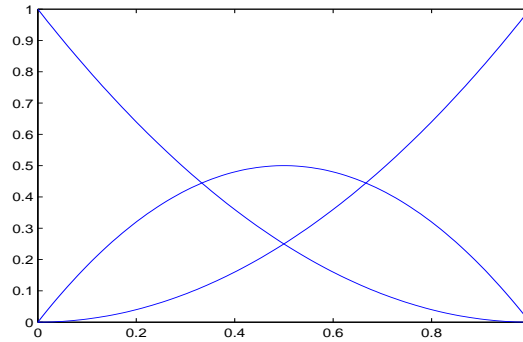
polynomials of degree 1.

- The Bernstein polynomials of degree 2 are $B_{0,2}(t) = (1 - t)^2$

$$B_{1,2}(t) = 2t(1 - t)$$

$$B_{2,2}(t) = t^2$$

and can be plotted for $0 \leq t \leq 1$ as



polynomials of degree 2.

4.1 Hermite Polynomials

The Hermite polynomials $H_n(x)$ are an important set of orthogonal functions over the interval $(-\infty, +\infty)$ and the general form of these polynomials is:

$$H_{n+1}(x) = -2xH_n(x) - 2nH_{n-1}(x) \quad n \geq 1$$

where $H_0(x) = 1$ and $H_1(x) = -2x$.

5. The solution of Linear IEMTL Using expansion Methods through of Bernstein polynomial

Expansion methods are one of the most efficient methods used to solve integral equations without time lags. In this section, expansion methods with the aid of Bernstein polynomial obtain the approximated solutions for IEMTL as follows.

Consider the linear IEMTL of the second kind:

$$u(t - \tau_1) = g(t) + \int_a^{b(t)} k(t, x)u(x - \tau_2)dx, \quad t \in [a, b(t)] \quad (4)$$

Where $\tau_1, \tau_2 \in R$ and $\tau_1, \tau_2 > 0$ Expansion methods are based on approximating the unknown function $u(t)$ by eq. (2)

$$u(t) \cong u_N(t) = \sum_{i=0}^N c_i B_{i,N}(t)$$

where $B_{i,n}(t)$ are Bernstein polynomial. By statuting eq. (2) into eq. (4), one gets the following formula:

$$\sum_{i=0}^N c_i B_{i,N}(t - \tau_1) = g(t) + \int_a^{b(t)} k(t, x) \sum_{i=0}^n c_i B_{i,N}(x - \tau_2) dx \quad (5)$$

Then

$$\sum_{i=0}^N c_i (B_{i,N}(t - \tau_1) - \int_a^{b(t)} k(t, x) B_{i,N}(x - \tau_2) dx) = g(t) \quad (6)$$

Hence, the residual equation $E_N(t)$ in eq. (3) for eq. (4) is defined by:

$$E_N(t) = \sum_{i=0}^N c_i (B_{i,N}(t - \tau_1) - \int_a^{b(t)} k(t, x) B_{i,N}(x - \tau_2) dx) - g(t) \quad (7)$$

6. The solution of Linear IEMTL Using Collocation Method

In collocation method [10], the weighting functions are chosen to be:

$$w_j = \begin{cases} 1, & t = t_j, \\ 0, & o.w, \end{cases} \quad (8)$$

Where the fixed points $t_j \in D$, $j = 0, 1, \dots, N$, are called collocation points. Inserting eq. (8) in eq. (3) gives:

$$\int_D w_j E_N(t) dt = E_N(t_j) \int_D w_j dt = 0 \Rightarrow E_N(t_j) = 0. \quad (9)$$

This turns eq. (7) into:

$$E_N(t_j) = \sum_{i=0}^N c_i [B_{i,N}(t_j - \tau_1) - \int_a^{b(t_j)} k(t_j, x) B_{i,N}(x - \tau_2) dx] - g(t_j) = 0.$$

Hence,

$$\sum_{i=0}^N c_i [B_{i,N}(t_j - \tau_1) - \int_a^{b(t_j)} k(t_j, x) B_{i,N}(x - \tau_2) dx] = g(t_j). \quad (10)$$

So, by expanding and simplifying eq. (10), we have $N + 1$ simultaneous equations with $N + 1$ unknown coefficients c_i . Hence, eq. (10) can be written in matrix form as $DC = G$,

where

$$D = \begin{bmatrix} d_{00} & d_{01} & \cdots & d_{0N} \\ d_{10} & d_{11} & \cdots & d_{1N} \\ \vdots & \vdots & \vdots & \vdots \\ d_{N0} & d_{N1} & \cdots & d_{NN} \end{bmatrix}, C = \begin{bmatrix} c_0 \\ c_1 \\ \vdots \\ c_N \end{bmatrix}, G = \begin{bmatrix} g(t_0) \\ g(t_1) \\ \vdots \\ g(t_N) \end{bmatrix} \tag{11}$$

$$d_{i,j} = B_{i,N}(t_j - \tau_1) - \int_a^{b(t_j)} k(t_j, x) B_{i,N}(x - \tau_2) dx, \quad i, j = 0, \dots, N$$

Then, Gauss elimination method is applied to find the coefficients c_i 's, $i = 0, 1, \dots, N$ which satisfy eq. (2).

6.1 The Solution of Linear IEMTL Using Partition Method

In partition method [10], the domain D is divided into N non-overlapping sub domains $D_j, j = 0, 1, \dots, N$, and the weighting functions w_j in eq. (3) are defined:

$$w_j = \begin{cases} 1, & t \in D_j, \\ 0, & t \notin D_j. \end{cases} \tag{12}$$

Then, eq.(4) is satisfied on the average in each of N sub domains D_j . Substituting eq. (7) and eq. (12) into eq. (3) yields:

$$\int_{D_j} [\sum_{i=0}^N c_i (B_{i,N}(t - \tau_1) - \int_a^{b(t)} k(t, x) B_{i,N}(x - \tau_2) dx)] dt = 0, \tag{13}$$

$D_j \in D, \quad j = 0, \dots, N$

Hence,

$$\sum_{i=0}^N \int_{D_j} c_i [B_{i,N}(t - \tau_1) - \int_a^{b(t)} k(t, x) B_{i,N}(x - \tau_2) dx] dt = \int_{D_j} g(t) dt, \tag{14}$$

$(D_j \in D, \quad j = 0, \dots, N)$

So, by expanding and simplifying eq. (14), we have $N + 1$ simultaneous equations with $N + 1$ unknown coefficients c_0, c_1, \dots, c_N . Hence, eq. (14) can be written in matrix form as $DC = G$, where

$$D = \begin{bmatrix} d_{00} & d_{01} & \cdots & d_{0N} \\ d_{10} & d_{11} & \cdots & d_{1N} \\ \vdots & \vdots & \vdots & \vdots \\ d_{N0} & d_{N1} & \cdots & d_{NN} \end{bmatrix}, C = \begin{bmatrix} c_0 \\ c_1 \\ \vdots \\ c_N \end{bmatrix}, G = \begin{bmatrix} \int_{D_0} g(t) \\ \int_{D_1} g(t) \\ \vdots \\ \int_{D_N} g(t) \end{bmatrix} \tag{15}$$

$$d_{i,j} = \int_{D_j} (B_{i,N}(t - \tau_1) - \int_a^{b(t)} k(t, x) B_{i,N}(x - \tau_2) dx)$$

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

Then, Gauss elimination method is applied to find the coefficients c_i 's, $i = 0, 1, \dots, N$ which satisfy eq. (2) (the approximate solution $u_N(t)$ of eq. (4)).

6.2 The Solution of linear IEMTL Using Galerkin's Method

In Galerkin's method [10], the weight functions w_j in eq. (3) are defined as:

$$w_j = \frac{\partial u_N(t)}{\partial c_j} \quad j = 0, \dots, N,$$

Then,

$$w_j = \frac{\partial \sum_{i=0}^N c_i B_{i,N}(t)}{\partial c_j} = B_{i,N}(t) \quad j = 0, \dots, N, \quad (16)$$

Substituting eq. (7) and eq. (16) into eq. (3) yields:

$$\int_{D_j} B_{j,N} [\sum_{i=0}^N c_i (B_{i,N}(t - \tau_1) - \int_a^{b(t)} k(t, x) B_{i,N}(x - \tau_2) dx)] - g(t) dt = 0, \quad (17)$$

$D_j \in D, \quad j = 0, \dots, N,$

Then

$$\sum_{i=0}^N c_i [\int_{D_j} B_{j,N} (\phi_i(t - \tau_1) - \int_a^{b(t)} k(t, x) B_{i,N}(x - \tau_2) dx)] = \int_{D_j} B_{j,N} g(t) dt, \quad (18)$$

$D_j \in D, \quad j = 0, \dots, N,$

So, by expanding and simplifying eq. (18), we have $N + 1$ simultaneous equations with $N + 1$ unknown coefficients c_0, c_1, \dots, c_N . Hence, eq. (18) can be written in matrix form as $DC = G$, where

$$D = \begin{bmatrix} d_{00} & d_{01} & \cdots & d_{0N} \\ d_{10} & d_{11} & \cdots & d_{1N} \\ \vdots & \vdots & \vdots & \vdots \\ d_{N0} & d_{N1} & \cdots & d_{NN} \end{bmatrix}, C = \begin{bmatrix} c_0 \\ c_1 \\ \vdots \\ c_N \end{bmatrix}, G = \begin{bmatrix} \int_{D_j} B_{0,N} g(t) dt \\ \int_{D_j} B_{1,N} g(t) dt \\ \vdots \\ \int_{D_j} B_{N,N} g(t) dt \end{bmatrix} \quad (19)$$

$$d_{i,j} = \int_D [B_{i,N}(t - \tau_1) - \int_a^{b(t)} k(t, x) B_{i,N}(x - \tau_2) dx] dt$$

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

Then, Gauss elimination method is applied to find the coefficients c_i 's ($i = 0, 1, \dots, N$) which satisfy eq. (2) (the approximate solution $u_N(t)$ of eq. (4)).

7. Numerical examples

In this section, we present two examples and their numerical results.

Example 1. Consider the following Volterra integral equation with multiple time lags:

$$u(t - 0.5) = (0.5 + t - \frac{t^4}{3}) + \int_0^t txu(x - 1)dx, \quad t \in [0, 1]. \tag{20}$$

The exact solution of eq. (19) is:

$$u(t) = t + 1, \quad 0 \leq t \leq 2.$$

Assume that the approximate solution of the example is:

$$u(t) \cong u_1(t) = \sum_{i=0}^1 c_i B_{i,1}(t).$$

When the three approaches are applied, Table 1. presents the comparison between the exact and collocation, partition and Galerkin methods with the aid of Bernstein polynomial for example 1. depending on least square error (L.S.E.) where $m = 10$, $h = 0.2$, $t_j = jh$, $j = 0, 1, \dots, m$.

Table 1
Numerical results of Example 1.

t	Exact	Collocation		Partition		Galerkin	
		Bernstein	Hermite	Bernstein	Hermite	Bernstein	Hermite
0	1	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000
0.2	1.2	1.2000	1.2000	1.2000	1.2000	1.2000	1.2000
0.4	1.4	1.4000	1.4000	1.4000	1.4000	1.4000	1.4000
0.6	1.6	1.6000	1.6000	1.6000	1.6000	1.6000	1.6000
0.8	1.8	1.8000	1.8000	1.8000	1.8000	1.8000	1.8000
1	2	2.0000	2.0000	2.0000	2.0000	2.0000	2.0000
1.2	2.2	2.2000	2.2000	2.2000	2.2000	2.2000	2.2000
1.4	2	2.4000	2.4000	2.4000	2.4000	2.4000	2.4000
1.6	2.6	2.6000	2.6000	2.6000	2.6000	2.6000	2.6000
1.8	2.8	2.8000	2.8000	2.8000	2.8000	2.8000	2.8000
2	3	3.0000	3.0000	3.0000	3.0000	3.0000	3.0000
L.S.E		0.0000	0.0000	0.0000	0.0000	0.0000	0.0000

Example 2. Consider the following Ferdholm integral equation with multiple time lags:

$$u(t - 0.2) = (e^{(t-1/5)} - te + t - 1) + \int_0^{\tau_2} (t + x)u(x)dx, \quad t \in [0, 1], \tag{21}$$

where $\tau_2 = 1$ and the exact solution of eq. (29) is:

$$u(t) = e^t, \quad 0 \leq t \leq 1.$$

Assume that the approximate solution of the example is:

$$u(t) \cong u_5(t) = \sum_{i=0}^5 c_i B_{i,5}(t).$$

When the three approaches are applied, Table 2. presents the comparison between the exact and collocation, partition and Galerkin methods with the aid of Bernstein polynomial for example 2. depending on least square error (L.S.E.) where $m = 10$, $h = 0.1$, $t_j = jh$, $j = 0, 1, \dots, m$.

Table 2
Numerical results of Example 2

t	Exact	Collocation		Partition		Galerkin	
		Bernstein	Hermite	Bernstein	Hermite	Bernstein	Hermite
0	1.0000	1.0000	1.0000	1.0000	1.0003	1.0000	1.0001
0.1	1.1052	1.1052	1.1058	1.1052	1.1060	1.1052	1.1058
0.2	1.2214	1.2214	1.2225	1.2214	1.2226	1.2214	1.2224
0.3	1.3499	1.3499	1.3513	1.3499	1.3514	1.3499	1.3512
0.4	1.4918	1.4918	1.4935	1.4918	1.4936	1.4918	1.4934
0.5	1.6487	1.6487	1.6504	1.6488	1.6505	1.6487	1.6504
0.6	1.8221	1.8221	1.8237	1.8222	1.8238	1.8221	1.8237
0.7	2.0138	2.0138	2.0150	2.0138	2.0150	2.0138	2.0150
0.8	2.2255	2.2255	2.2263	2.2256	2.2256	2.2255	2.2263
0.9	2.4596	2.4596	2.4596	2.4596	2.4596	2.4596	2.4597
1	2.7183	2.7182	2.7174	2.7182	2.7173	2.7182	2.7175
L.S.E.		7.2959e-9	1.446e-5	1.413e-8	1.66e-5	7.2951e-9	1.413e-5

8. Conclusion

The approximated solutions using three types of expansion methods containing (collocation, partition and Galerkin) with the aid of two different types of basis functions (Bernstein polynomial and orthogonal functions) were obtained for two examples. The results showed a marked improvement in the least square errors and the following conclusion points are listed:

1. In terms of the results, Galerkins method gave more accurate results than collocation and partition methods, see table (2).
2. For basis functions, Bernstein polynomial gave more accurate results than orthogonal function (Hermite polynomial), see table (2).

8.1 Acknowledgements

The authors would like to thank the anonymous referees for his/her constructive comments and suggestions.

References

- [1] A. Jerri, *Introduction to Integral equations with application*, 1st edition, Marcel Dekker USA, 1985.
- [2] T. A. Burton, *Integral equation with delay*, Acta Math. Hung. 72 (3) (1998), pp. 233–242.

- [3] H. Smith, *On periodic solution of delay integral equations modeling epidemics*, Jr. Math. Biology 4 (1997), pp. 69–80.
- [4] D. E. Kamen, P. P. Khargonekar, A. Tannenbaum, *Proper factorizations and feedback control of linear time-delay system*, International Jr. of control 43 (2006), pp. 837–857.
- [5] D. D. Bhatta, M. I. Bhatti, *Numerical solution of KdV equation using modified Bernstein polynomials*, Appl. Math. Comput. 174 (2006), pp. 1255–1268.
- [6] M. I. Bhatti, P. Bracken, *Solutions of differential equations in a Bernstein polynomial basis*, J. Comput. Appl. Math. 205 (2007), pp. 272–280.
- [7] B. N. Mandal, S. Bhattacharya, *Numerical solution of some classes of integral equations using Bernstein polynomials*, Appl. Math. Comput. 190 (2007), pp. 1707–1716.
- [8] A. J. Kadhim, *Expansion Methods for Solving Linear Integral Equations with Multiple Time Lags Using B-Spline and Orthogonal Functions*, Eng. Tech. Journal 29 (9) (2011), pp. 1651–1661.
- [9] K. Engelborghs, T. Luzyanina, D. Roose, *Collocation Methods for the computation of periodic solution of delay differential equation*, SIAM Jr. Sci. Comput. 22 (5) (2000), pp. 1593–1609.
- [10] M. Gelfand, S. Famines, *Methods of Numerical approximation*, Oxford university computing Laboratory, 2006.