A Computerizable Iterative-Algorithmic Quadrature Operator
Using an Efficient Two-Phase Modification of Bernstein Polynomial

Shanaz Ansari Wahid, *Ashok Sahai
Department of Mathematics & Computer Science;
Faculty of Science & Agriculture
St. Augustine Campus; The University of The West Indies.
Trinidad & Tobago; West Indies.

M. Raghunadh Acharya
Department of Statistics and Computer Science,
Aurora's Post Graduate College; Osmania University,
Hyderabad (Andhra Pradesh); India.
Department of Mathematics & Computer Science
The University of the West Indies, Trinidad & Tobago

Abstract.: A new quadrature formula has been proposed which uses modified weight functions derived from those of 'Bernstein Polynomial' using a 'Two-Phase Modification' therein. The quadrature formula has been compared empirically with the simple method of numerical integration using the well-known "Bernstein Operator". The percentage absolute relative errors for the proposed quadrature formula and that with the "Bernstein Operator" have been computed for certain selected functions, with different number of usual equidistant node-points in the interval of integration~ [0, 1]. It has been observed that both of the proposed modified quadrature formulae, respectively after the 'Phase-I' and after the 'Phases-I & II' of these modifications, produce significantly better results than that using, simply, the "Bernstein Operator". Inasmuch as the proposed "Two-Phase Improvement" is available iteratively again-and-again at the end of the current iteration, the proposed improvement algorithm, which is 'Computerizable', is an "Iterative-Algorithm", leading to more-and-more efficient "Quadrature-Operator", till we are pleased!

Keywords: Quadrature Formula; Percentage Absolute Relative Errors; Weight Functions.

AMS Classification Number: 41A10; 41A36; 41A55; 65D32; 68W25.

I. Introduction

Classically, even the celebrated Bernstein polynomial approximation operator has been used for numerical integration in the interval [0, 1], without any loss of generality (in the sense of change of origin-and-scale). In a rather simple set-up, such similar quadrature methods were of the form

$$\int_a^b f(x)dx \approx \sum_{i=1}^{i=n} w_i . f(x_i)$$

Where x_i's are equally spaced nodes & w_i's are the respective weights. For example:

Trapezoid: $\int_a^b f(x)dx \approx [(b-a)/2]*f(a) + [(b-a)/2]*f(b).$

Simpson:
$$\int_a^b f(x)dx \approx [(b-a)/6]*f(a) + [2(b-a)/3]*f((a+b)/2) + [(b-a)/6]*f(b).$$

Similarly, for higher order polynomial Newton-Cotes rules.

We note one known thing already from interpolation: equally-spaced nodes result in wiggle.

The preceding fact motivated us to ponder-n-wonder as to whether or not we would be able to retain the simplicity of "equidistant points in the interval of integration" in our proposed quadrature formula, and still aspire successfully for its optimality. In fact we were successful. The next section details the 'How' part of it.

And the section following that numerically illustrates that the potential of this new/proposed 'two-phase' modifications of the usual 'Bernstein Operator' was very significant.

Moreover, Inasmuch as the proposed "Two-Phase Improvement" is available iteratively again-and-again at the end of the current iteration, the proposed improvement algorithm, which is 'Computerizable', is an "Iterative-Algorithm", leading each time to a more-and-more efficient "Quadrature-Operator", till we are pleased!

Hence, the proposed "Computerizable Iterative-Algorithmic Two-Phase Improvement of the Bernstein's Quadrature Operator" is almost 'Optimal'!

That it could be almost the 'Optimal' one, inasmuch as it could be used to lead us to an exceedingly superior 'quadrature-operator' vis-à-vis to the simple-and-famous "Bernstein Operator Quadrature". The "Bernstein Operator" happened to be the foundational structure for the build-up of the proposed one, using weight-functions derived using the 'two-phase' modifications. We are going to detail this in following section. And this modification, as noted above, would be available "Iteratively".

II. THE PROPOSED PROBABILISTICALLY WEIGHTED QUDRATURE OPERATOR.

The problem of approximation arises in many contexts of 'Numerical Analysis and Computing', and 'Quadrature' is one such. Weierstrass (1885) proved his celebrated approximation theorem: → if f ε C [a, b]; for every $\delta > 0$; there is a polynomial 'p' such that $\lVert f - p \rVert < \delta$.

In other words, the result established the existence of an algebraic polynomial, in the relevant variable, capable of approximating the unknown function in that variable, as closely as we please!

This result was a big beginning of the mathematicians' interest in 'Polynomial Approximation' of an unknown function using its values generated experimentally or known otherwise at certain chosen 'Knots' of the domain of the relevant variable, as of interest to the scientist concerned. The Great Russian mathematician S. N. Bernstein proved the Weierstrass' theorem in a manner which was very stimulating and interesting in many ways.

He first noted a simple but important fact that if the Weierstrass' theorem holds for the interval C [0, 1], it also holds for C [a, b] and holds conversely. Essentially C [0, 1] and C [a, b] are identical, for all practical purposes; as they are linearly isometric as normed spaces, order isomorphic as algebras (rings). Most important contribution in the Bernstein's proof of this theorem consisted in the fact that Bernstein actually displayed a sequence of polynomials that approximate a given function f(x) ε C [0, 1]. If f (x) is any bounded function on C [0, 1], the sequence of Bernstein's Polynomials for f (x) is defined by:

$$\mathbf{(Bn\ (f))(x)} = \sum_{k=0}^{k=n} w_{k,n} * f\left(\frac{k}{n}\right) \qquad \mathbf{(2.1)}$$

Wherein, $w_{k,n} = \binom{n}{k} x^k (1-x)^{n-k}$ are the respective weights for the values '$f\left(\frac{k}{n}\right)$' of the function at the knots "$\left(\frac{k}{n}\right)$" [k = 0 (1) n].Thus if we assume that, without any loss of generality [subject to the change of origin-and-scale of the impugned-variable, if necessary], our interest is that in developing the "Quadrature Operator" for the numerical integration of the definite integral $\int_0^1 f(x)dx$; (2.2)

"f (x)" being a bounded unknown function with known/knowable values at **equi-distant knots "$\left(\frac{k}{n}\right)$" [k = 0 (1) n].**

Hence, one straightforward "Quadrature" using the Bernstein's **Operator** $\sim \int_0^1 f(x)dx =$

$$\sum_{k=0}^{k=n} w_{k,n} * f\left(\frac{k}{n}\right) = IB_{k,n}(f;\, x); \textbf{ Say.} \qquad \mathbf{(2.3)}$$

Now, we are set to launch the "Two-Phase Modification Perspective" seminal to the proposition of our novel 'Quadrature Operator" in the same "Equi-Distant Knots" 'Set-Up' of Bernstein's Operator, in what follows.

Consider the fact that f (x) is bounded unknown function in its domain C [0, 1]. Let us, call by "**PhIMIB$_{k,n}$(f; x)**" the 'Phase I Modified Bernstein's Quadrature Operator', considering "**M_0 *IB$_{k,n}$(f; x)**" wherein the constant "**M_0**" is so chosen as to minimize the "Estimable Summed-Squared Error" using this 'Phase I Modified Bernstein's Quadrature Operator'.

To get this constant "**M_0**" using the "Estimable Error", we consider the use of the perturbed operator "b* IB$_{k,n}$ (f; x) " as an nth.-degree 'Polynomial Approximation Operator'. For each 'Knot' 'k/n (k = 0(1) n)' we get the value of "b*IB$_{k,n}$ (f; k/n) "; say '**Est.[b. (f; k/n)]**' vis-à-vis the actual value (f; k/n) ~ namely**, f(k/n).**

Consider now, the **'Overall Estimable Summed-Squared-Error' ~**

$$\sum_{k=0}^{k=n} \{Est.[b*(f;k/n)] - (f;k/n)\}^2 =$$

$$\mathbf{A*b^2 - B*b + C;\ Say.}$$

This **'Overall Estimable Summed-Squared-Error'** will be zero for the roots of the equation ~ **$A*b^2 - B*b + C = 0$.** However, the above equation turns up to be having the imaginary roots. Therefore, we set $b_0 \equiv M_0$ **= B/ (2*A), to have reduced 'Overall Estimable Summed-Squared-Error', for the real-root** ~ **"M_0".**

Hence, our '**Phase-I Modified Bernstein's Quadrature Operator'** gets to be:

PhIMIB$_{k,n}$ (f; x) = M_0 *IB$_{k,n}$ (f; x); wherein M_0 = B/ (2*A). (2.4)

Whereas, in the **"Phase-I"** detailed above, we had a modification of the 'Bernstein's Quadrature Operator **"PhIMIB$_{k,n}$ (f; x)"**, consisting in a change-of-scale; we attempt an optimal polynomial shift in the **"Phase-II"** detailed in what follows.

We estimate the **'(nth. Degree) Polynomial Approximation' to the "Estimable Error"** by the 'Bernstein Operator $\sum_{i=0}^{i=n}(w_{k,n} *$ **(Est.[b_0. (f; k/n)] - (f; k/n))** '; Say, **EPhI$_n$(x)**'. Thence, the resultant **'Phase-II Modified Bernstein's Quadrature Operator'** gets to be:

PhIIMIB$_{k,n}$ (f; x) = PhIMIB$_{k,n}$ (f; x) - EPhI$_n$(x). (2.5)

III. Numerical Simulation Study.

This section is also of prime interest, as herein we try to illustrate the potential of our proposed quadrature operators **"(On (f)) (x)"**. As apparent in the second section, as a prelude to our proposed operator, the mother-operators are the **"Bernstein's Polynomials for f (x) (Bn (f)) (x)" in (2.1)**.

As such, as we could not have an idea about their relative supremacy in terms of better estimation-potential otherwise; we have to discover their relative supremacy of efficient estimation only via a 'Numerical Simulation Study', as attempted in what follows in this section.

In this simulated numerical study we have chosen four illustrative example-functions: **exp (x), 10^x, sin (2+x), and ln (2+x); assumed to be known in the sense of "Simulation"**.

For the simplicity of the numerical illustration, we have confined to chosen illustrative **n-values to be 5, 10, and 15.**

We have considered numerical values (per the illustrative numerical study) of the **"Percentage Relative Absolute Errors"** in using the relevant operators by the evaluation of the expressions: namely **"PRAbsErr (●) (In %)"** for **"(On (f)) (x)"; &** for **"(Bn (f)) (x)",** respectively→

$$\left[\left|\int_0^1 O_n(f;x) - \int_0^1 f(x)dx\right| \text{x} \quad 100\right] / \left[\int_0^1 f(x)dx\right] = \textbf{PRAbsErr } (\textbf{O}_n)\textbf{; Say.}$$

And

$$\left[\left|\int_0^1 B_n(f;x) - \int_0^1 f(x)dx\right| \text{x} \quad 100\right] / \left[\int_0^1 f(x)dx\right] = \textbf{PRAbsErr } (\textbf{B}_n)\textbf{; Say.}$$

These "Percentage Relative Absolute Errors" [~ "PRAbsErr (●) (In %)"], calculated using the *"MAPLE RELEASE 12(Evaluation-Version)"* code, are tabulated in the following four tables (Tables 1.1 to 1.4) in the APPENDIX.

This "Numerical Simulation Illustration" has amply supported the fact that the proposed "Two-Phases" of modifications to the usual "Bernstein's Quadrature Operator" are quite gainful, inasmuch as the "Percentage Absolute Errors' Numerical-Values" for our proposed Phase I/Phase II modified quadrature-operators "(On (f)) (x)" are significantly lower than those for "Bernstein's Polynomials for f (x); (Bn (f)) (x)" in (2.1).

The most important fact, to be re-iterated here, is that this betterment could be availed iteratively; till we please! Secondly this "Iterative-Algorithm" could well be computerized, rather!!

References:

[1] E. W. Cheney and A. Sharma, Bernstein power series, Canad. J. Maths. 16 (1964), 241- 252.

[2] P.P. Korovkin. Linear Operators and Approximation Theory. Hindustan Publishing, Delhi, 1960.

[3] http://www.maplesoft.com/contact/webforms/maple_evaluation.aspx ~ Maple 12 Evaluation-Version

APPENDIX

Table 1.1.

Percentage Relative Absolute Error [~ "PRAbsErr (●) (In %)] of Various Modified [After Phase-I & After Phases-I &II; respectively] Bernstein Operators (In %) for Example-Function: f(x) = exp(x).

Operator↓	For n =5↓	For n =10↓	For n =15↓
PhIIMIB$_{k,n}$(f; x)	0.3474210635	0.0877024930	0.0389968624
PhIMIB$_{k,n}$(f; x)	0.4707192306	0.1388406699	0.0658716738
IB$_{k,n}$(f; x)	1.6438712520	0.8209813297	0.5500521420

Table 1.2.

Percentage Relative Absolute Error [~ "PRAbsErr (●) (In %)] of Various Modified [After Phase-I & After Phases-I &II; respectively] Bernstein Operators (In %) for Example-Function: f(x) = 10^x.

Operator↓	For n =5↓	For n =10↓	For n =15↓
PhIIMIB$_{k,n}$(f; x)	1.997008283	0.520904512	0.234668062
PhIMIB$_{k,n}$(f; x)	3.988609739	1.414582407	0.787830334
IB$_{k,n}$(f; x)	8.253158947	4.102544208	2.732405685

Table 1.3.

Percentage Relative Absolute Error [~ "PRAbsErr (●) (In %)] of Various Modified [After Phase-I & After Phases-I &II; respectively] Bernstein Operators (In %) for Example-Function: f(x) = sin (2+x).

Operator↓	For n =5↓	For n =10↓	For n =15↓
PhIIMIB$_{k,n}$(f; x)	0.317408726	0.0784870970	0.0326542506
PhIMIB$_{k,n}$(f; x)	0.456275124	0.1307803739	0.0691217565
IB$_{k,n}$(f; x)	1.690566589	0.8462742402	0.5733955363

Table 1.4.

Percentage Relative Absolute Error [~ "PRAbsErr (●) (In %)] of Various Modified [After Phase-I & After Phases-I &II; respectively] Bernstein Operators (In %) for Example-Function: f(x) = ln (2+x).

Operator↓	For n =5↓	For n =10↓	For n =15↓
$PhIIMIB_{k,n}(f; x)$	0.0603252620	0.0149794319	0.0050210408
$PhIMIB_{k,n}(f; x)$	0.0676422922	0.0192393427	0.0143450140
$IB_{k,n}(f; x)$	0.3012250535	0.1504128494	0.1004587904