УДК 004.942

С.В.Иваница, А.В. Меркулов
Донецкий национальный технический университет
isv@cs.dgtu.donetsk.ua

Методы контроля точности результирующих интервалов при вычислении интервальных полиномиальных функций

Предложены два метода (метод дифференцирования и метод замены интервала) контроля точности при вычислении интервальных полиномиальных функций с целью оптимизации результирующих интервалов, т. е. предотвращения получения «шума» в процессе интервальных вычислений. Рассмотрены результаты работы методов на примере расчета интервальных полиномов 2-й и 5-й степени. Представлены результаты работы программной модели, демонстрирующей эффективность использования рассматриваемых методов оптимизации при интервальных вычислениях.

Интервал, интервальная математика, полином, единичный интервал, верификация, дифференцирование

Введение

Любое действительное число однозначно представимо на числовой оси, и приближенно — в памяти электронной вычислительной машины (ЭВМ). Возникшее приближение обусловлено машинными форматами представления вещественных чисел (например, форматами чисел с плавающей запятой), которые конечным набором двоичных значений отображают бесконечное множество действительных чисел [1]. Таким образом, поскольку в ЭВМ с конечной разрядной сеткой невозможно точное представление вещественных чисел, то результат каждого достаточно сложного расчета содержит некоторую ошибку, обусловленную погрешностями округления входных данных и промежуточных результатов. Это означает, что если в вычислении учувствуют приближенно представленные (округленные) значения, то и результат окажется приближенным. Кроме того, на погрешность, полученную при вычислении, накладывается и погрешность, возникшая при представлении полученного значения в формате с плавающей запятой (ошибка округления). Поэтому, в результате накопления погрешности, на каждом шаге вычисления степень «достоверности» результата снижается, приводя порой к совершенно неверному решению вычислительной задачи.

Указанные обстоятельства привели к тому, что в 60-70-е годы XX столетия в компьютерные технологии начинает внедряться более широкая парадигма понимания природы числа, которая формировалась в это время в различных областях прикладной математики. Так, любое число может быть представлено не только в виде точки на числовой оси, а и в виде некоего отрезка-интервала. Новые подходы к представлению числа вызвали появление как концепции многозначной логики (трилогики, тетралогики, и т. д.) [2, 3], теории нечетких множеств и нечетких чисел, так и интервального анализа [4, 5], в рамках которого реализуются представленные в данной статье методы оптимизации.

В последние годы наблюдается повышение интереса к интервальному анализу, как средству для реализации точных вычислений при проведении расчетов на ЭВМ. Интервальный анализ как научное направление сформировался относительно недавно, в основном как метод автоматического контроля ошибок округления на ЭВМ, обусловленный тем, что во многих вычислительных задачах возникла потребность не только вычисления приближенных решений, но и гарантированных оценок их близости к точным решениям [6].

Однако для сложных задач применение интервального анализа часто дает неудовлетворительные результаты из-за чрезмерной длины получаемых интервалов, поскольку в ряде случаев оценки точности оказываются на порядок хуже, чем реально достигаемая точность результатов [7, 8].

Чтобы проиллюстрировать данную ситуацию, приведем простой пример. Рассмотрим следующую функцию

$$f(x) = x - x, \qquad (1)$$

которая для любого значения x в пределах множества действительных чисел R принимает нулевое значение. Однако число можно представить в виде множества числовых значений, которое задается его крайними границами. В таком случае данное число — интервальное (определенное на множестве интервалов действительных чисел IR), а его крайние границы — левая (меньшая) и правая (большая) границы интервала. Все арифметические операции над такими числами образуют класс интервальных арифметических операций, которые оперируют границами интервалов-операндов [9]. При переходе к интервальным вычислениям функция (1) примет вид интервальной функции $\boldsymbol{f}(\boldsymbol{x})$, где аргумент $\boldsymbol{x} = [x_1, x_2]$ — интервальное число с границами x_1 (левая) и x_2 (правая), причем $x_1 \le x_2$. Тогда

$$\boldsymbol{f}(\boldsymbol{x}) = \boldsymbol{x} - \boldsymbol{x} = [x_1, x_2] - [x_1, x_2] = \\ = [-|x_2 - x_1|, |x_1 - x_2|]. \quad (2)$$

Таким образом, результирующий интервал $\boldsymbol{f}(\boldsymbol{x})$ имеет ненулевую длину при $x_1 \ne x_2$, и, следовательно, $\boldsymbol{x} - \boldsymbol{x} \ne 0$. Однако, исходя из (1) $f(x) = x - x = 0$. Потеря точности произошла из-за того, что в выражении $\boldsymbol{x} - \boldsymbol{x}$ оба операнда не являются независимыми друг от друга, так как они равны. Аналогично, если $\boldsymbol{f}(\boldsymbol{x}) = \boldsymbol{g}(\boldsymbol{x}) \blacklozenge \boldsymbol{h}(\boldsymbol{x})$, где $\blacklozenge$ — некоторая арифметическая операция, реальный выходной интервал будет заведомо уже, чем при $[g_1, g_2] \blacklozenge [h_1, h_2]$, т. к. $\boldsymbol{g}$ и $\boldsymbol{h}$ — функции от одной и той же интервальной переменной.

Целью данной работы является разработка методов оптимизации при вычислении значения полиномиальной интервальной функции от интервального аргумента. В статье приводится обоснование неизбежности расширения результирующих интервалов и предлагается качественно новый подход к устранению данного явления.

Верификация значения интервальных функций методом дифференцирования

Рассмотрим некоторую интервальную функцию $\boldsymbol{f}(\boldsymbol{x})$, определенную на всех точках интервала $\boldsymbol{x} = [x_-, x_+]$, т. е. непрерывную и дифференцируемую на рассматриваемом участке $\boldsymbol{x}$. Фактически

$$\boldsymbol{f}(\boldsymbol{x}) = [f_-, f_+] = [\boldsymbol{f}_{\min}(\boldsymbol{x}), \boldsymbol{f}_{\max}(\boldsymbol{x})], \quad (3)$$

где f_-, f_+ — минимум и максимум функции $\boldsymbol{f}$ на области $[x_-, x_+]$. Если все коэффициенты — обычные действительные числа, то задачу нахождения точного (верифицированного) интервала $\boldsymbol{f}(\boldsymbol{x}) = [f_-, f_+]$ можно решить, используя значения производной функции $\boldsymbol{f}(\boldsymbol{x})$ на участке $[x_-, x_+]$.

Очевидно, что для нахождения экстремумов функции следует найти корни производной данной функции на области определения $\boldsymbol{x} = [x_-, x_+]$ и сравнить значения функции в точках x_-, x_+ и нулях производной. Среди перечисленных значений выбираются максимум и минимум — это и есть границы области значений функции для данного $\boldsymbol{x}$. Алгоритм работы метода дифференцирования приведен на рис. 1.

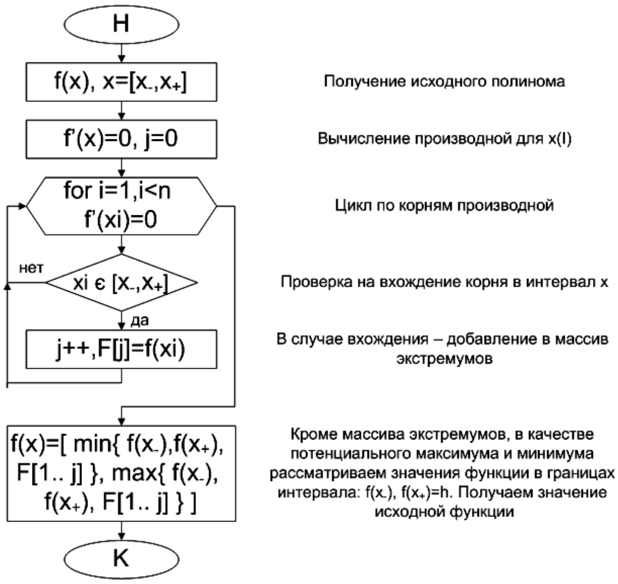

Рисунок 1 — Алгоритм верификации методом дифференцирования

Для полиномов такой подход очевиден и легко реализуем. Но для остальных функций подобный подход имеет очевидные слабые места, даже несмотря на то, что математически он полностью верен. Во-первых, допущения, которые мы приняли для $\boldsymbol{f}(\boldsymbol{x})$ (определенность, непрерывность, дифференцируемость) в общем случае не обязаны выполнятся. Во-вторых, в случае, если все допущения выполнены и возможна выборка производной, для некоторых функций процесс взятия производной будет более трудоемким, чем само вычисление результирующего интервала. В-третьих, если производная определима, нужно еще найти ее действительные корни (иначе применение данного подхода не имеет смысла), а известны сложные уравнения, для которых получить численные решения невозможно, т. к. из-за специфических особенностей функции, можно получить совершенно неверные результаты. Если организовать просчет решения производной с интервальными аргументами, то может возникнуть проблема верификации еще одного интервала, т. е. нужно повторно проводить дифференцирование (если это возможно) и т. д. В итоге, процесс верификации можно производить бесконечно и так и не достичь приемлемого результата. Для подобных «тяжелых» функций $\boldsymbol{f}(\boldsymbol{x})$ выгодна их аппроксимация на входном интервале $\boldsymbol{x}$ полиномом с необходимой точностью. Это сразу снимает весь комплекс проблем, но достоверность решения будет соответствовать достоверности аппроксимации функции.

Введение интервальных коэффициентов $\boldsymbol{a}_i$ в полиномиальную функцию $\boldsymbol{f}(\boldsymbol{a}_i, \boldsymbol{x})$ добавляет новые проблемы к получению верифицированных значений. Очевидно, что в таком случае любой действительный корень производной будет интервалом. Поэтому функция $\boldsymbol{f}(x_i)$, где x_i — корень производной, нуждается в верификации. Сделать это невозможно, так как $x_i = \boldsymbol{g}(\boldsymbol{a}_n, \ldots, \boldsymbol{a}_1)$, где $\boldsymbol{a}_i$, $i = n, n–1, \ldots, 0$, $n \in \boldsymbol{N}$ — коэффициенты $\boldsymbol{f}(\boldsymbol{x})$, но x_i — точечное значение. Таким образом,

$$\boldsymbol{f}(x_i) = \boldsymbol{h}(\boldsymbol{a}_n, \ldots, \boldsymbol{a}_0, \boldsymbol{g}) = \boldsymbol{h}(\boldsymbol{a}_n, \ldots, \boldsymbol{a}_0), \quad (4)$$

однако $\boldsymbol{g}$ представлено не функцией, что лишает смысла весь процесс верификации.

Верификация значения интервальных функций методом замены

Для верификации интервальных функций можно использовать замену, которую можно организовать следующим образом. Необходимо выразить интервальную переменную $\boldsymbol{x}$ через единичный интервал $\boldsymbol{I} = [0, 1]$:

$$\boldsymbol{x} = [x_1, x_2] = [0, 1]\cdot(x_2 - x_1) + x_1 = \boldsymbol{I}\cdot \mathrm{d} + x_1,$$

где $\mathrm{d} = x_2 - x_1$ — расстояние между границами интервала $\boldsymbol{x}$.

Таким образом, полином

$$\boldsymbol{f}(a_i, \boldsymbol{x}) = a_n\cdot x^n + a_{n-1}\cdot x^{n-1} + \ldots + a_1\cdot x + a_0$$

принимает вид

$$\boldsymbol{f}(a_i, \boldsymbol{x}, \boldsymbol{I}) =$$
$$= a_n \cdot x_1^n + a_{n-1} \cdot x_1^{n-1} + \ldots + a_1 \cdot x_1 + a_0 +$$
$$+ \boldsymbol{g}(\mathrm{d}, x_1, \boldsymbol{I}). \quad (5)$$

Единичный интервал $\boldsymbol{I}$ удобен тем, что при возведении в степень, он не расширяется и не сужается. Однако при возведении в степень единичного интервала справедливо выражение $\boldsymbol{I}^n \neq \boldsymbol{I}$, хотя границы этих интервалов и совпадают. Очевидно, что для $x \in (0, 1)$ справедливо неравенство $x^n < x$. Таким образом, для преобразованного выражения также как и в первом методе необходимо применять дифференцирование. Однако выражение (5) лучше в плане удобства представления входного интервала и четкого разделения на действительную и интервальную часть.

Для полиномов с действительными коэффициентами данный метод позволяет получить такие же результаты, что и метод дифференцирования, но обладает большими вычислительными затратами за счет наличия дополнительных преобразований. Но для полиномиальных функций с интервальными коэффициентами, эти преобразования оправдывают себя, т. к. они переносят независимые интервальные значения в свободный член полинома. При этом длина интервала не увеличивается в такой же мере, как в предыдущем методе, что позволяет получить более оптимизированный выходной интервал. Алгоритм работы данного метода приведен на рис. 2.

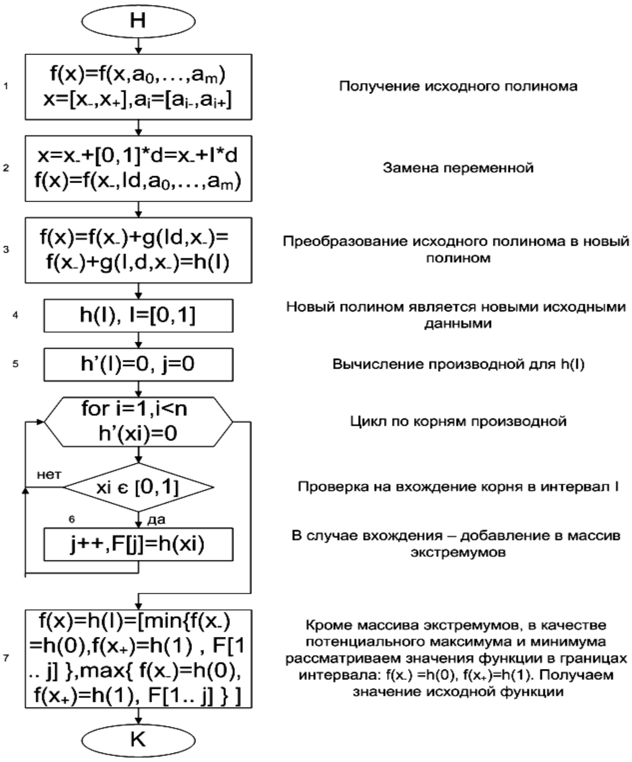

Рисунок 2 — Алгоритм верификации методом замены

Данный алгоритм также применим и для функций нескольких переменных $\boldsymbol{f}(\boldsymbol{x}_i)$, $i = 1, \ldots, m$, $m \in \boldsymbol{N}$. В этом случае для каждой переменной вводятся свои собственные единичные интервалы, $\boldsymbol{Ix}_1, \boldsymbol{Ix}_2, \ldots, \boldsymbol{Ix}_m$, т. к они независимы друг от друга.

Проверка эффективности предложенных методов

В поддерживающей интервальные вычисления системе компьютерной алгебры Wolfram Mathematica 7.0 была проведена проверка эффективности верификации интервала на примере полиномиальных интервальных функций второй и пятой степени:

$$\boldsymbol{f}_1(\boldsymbol{x}) = \boldsymbol{x} - \boldsymbol{x}^2; \quad (6)$$

$$\boldsymbol{f}_2(\boldsymbol{x}) = \boldsymbol{x}^5 - 5\boldsymbol{x}^4 + 4\boldsymbol{x}^3 + \boldsymbol{x}^2 - \boldsymbol{x}. \quad (7)$$

В интервальной математике результирующий интервал функции $\boldsymbol{f}(\boldsymbol{x})$ вычисляется следующим образом:

$$\boldsymbol{f}(\boldsymbol{x}) = [\min(f(x)), \max(f(x))] \text{ при } x \in \boldsymbol{x}. \quad (8)$$

Однако при таких вычислениях наблюдается неконтролируемое увеличение длины интервала. На рис. 3 показаны результаты вычисления функции (6) для интервалов различной длины в виде прямоугольных областей (светлые области — результаты вычисления функции средствами интервальной математики, темные области — с помощью метода дифференцирования). Полученные результаты при вычислении функции $\boldsymbol{f_1(x)}$ при различных входных значениях приведены в таблице 1.

На рис. 4 показаны результаты вычисления функции (7) для интервалов различной длины в виде прямоугольных областей при входном интервале $\boldsymbol{x} = [-0.1, 0.1]$. Получение результирующего интервала $\boldsymbol{f_2(x)}$:

– средствами интервальной математики: [–0.10451, 0.11401];

– с помощью метода дифференцирования: [–0.08649, 0.10549].

Таблица 1 — результаты вычисления функции $\boldsymbol{f_1(x)}$ с указанием длины результирующих интервалов

i	Входной интервал $\boldsymbol{x_i}$	Получение результирующего интервала $\boldsymbol{f_1(x)}$			
		средствами интервальной математики		с помощью метода дифференцирования	
1	[0.7, 1.4]	[–1.26, 0.91]	$d_1 = 2.17$	[–0.56, 0.21]	$d_1 = 0.77$
2	[0.3, 0.6]	[–0.06, 0.51]	$d_2 = 0.62$	[0.21, 0.25]	$d_2 = 0.04$
3	[–0.4, 0.2]	[–0.56, 0.2]	$d_3 = 0.76$	[–0.56, 0.16]	$d_3 = 0.72$

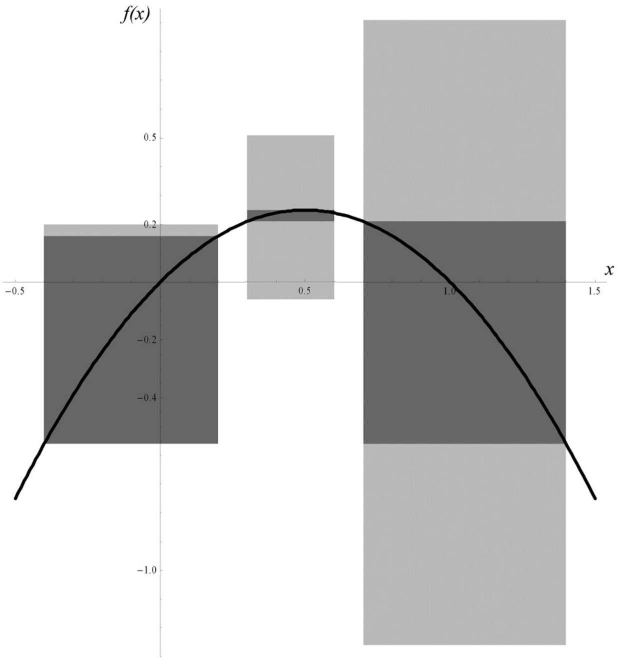

Рисунок 3 — Результат вычисления полинома $\boldsymbol{f_1(x)}$

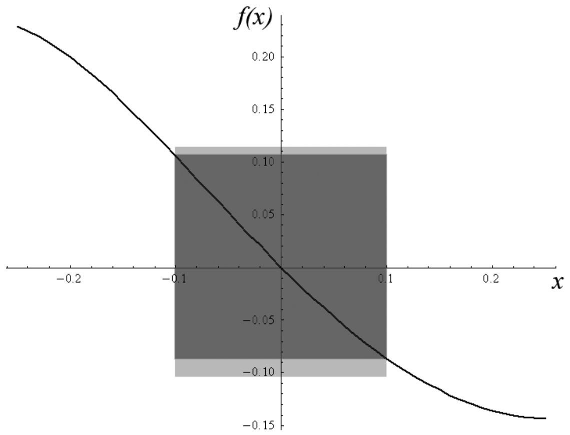

Рисунок 4 — Результат вычисления полинома $\boldsymbol{f_2(x)}$

Таким образом, при решении полиномов (6) и (7) были получены интервалы наименьшей длины, что, безусловно, доказывает эффективность предложенных методов.

В рамках внедрения данных методов в интервальный анализ, разработана программная модель *math*, способная выполнять вычисление полиномиальных интервальных функций. В качестве вычислительного ядра модели выбрано ядро математического пакета Wolfram Mathematica, которое поддерживает интервальные вычисления. Применение данного программного продукта как самостоятельного вычислительного средства обусловлено, во-первых, возможностью оптимизации промежуточных результатов на различных этапах сложных интервальных вычислений, и, во-вторых, способностью провести полный цикл вычислений с гарантированием точности и максимальным «сужением» границ результирующих интервалов.

На рис. 5. приведен скриншот программы, при вычислении полинома $f_2(x)$, показывающий порядок ввода входных значений и получение оптимизированного и обычного результатов.

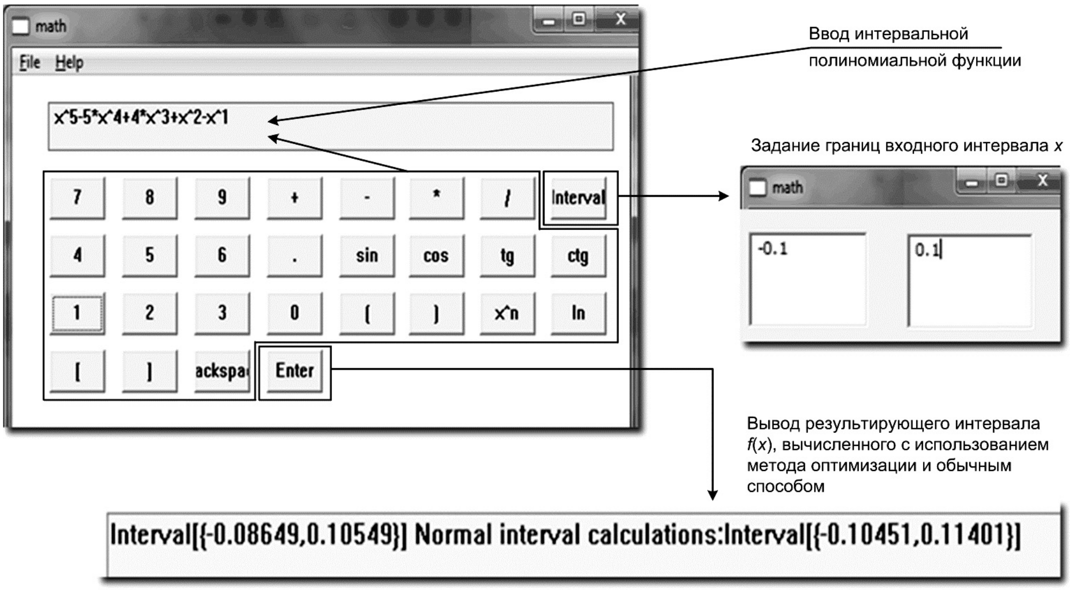

Рисунок 5 — Использование программы *math* для получения результатов вычисления полинома $f_2(x)$

Выводы

Многие системы компьютерной алгебры (например, Mathematica, SciLab, Mapple, и др.) обладают возможностью оперировать интервальными данными. Однако ни одна система не показала достаточного владения средствами интервального анализа, ограничившись базовым инструментарием интервальной математики. Разработанные методы позволяют оптимизировать результирующие интервалы и, несомненно, могут быть использованы как средство для уточнения промежуточных результатов при интервальных вычислениях.

В дальнейшем, в качестве эффективности предложенных методов оптимизации планируется решение ряда проблемных вычислительных задач, таких как, например, полином Румпа [10], правильное решение которого обычными средствами математики (в том числе и интервальной) получить крайне затруднительно.

Список литературы

1. Standard for Floating-Point Arithmetic (IEEE 754): IEEE754-2008 — IEEE, From Wikipedia, the free encyclopedia. http://en.wikipedia.org/wiki/IEEE_754–2008.

2. Аноприенко А.Я. Тетралогика и тетракоды / А.Я. Аноприенко // Сборник трудов факультета вычислительной техники и информатики. – 1996. – Вып.1.

3. Аноприенко А.Я. Особенности постбинарного кодирования на примере интервального представления результатов вычислений по формуле Бэйли-Боруэйна-Плаффа / А.Я. Аноприенко, С.В. Иваница / Научные труды Донецкого национального технического университета. Серия: Информатика, кибернетика и вычислительная техника (ИКВТ-2010). – 2010. – Вып. 11 (164). – С. 19-23.

4. Moore R.E. Interval analysis. Eiiglewood Cliffs / R.E. Moore — N.J.:Prentic-e-llall, 1966.

5. Шокин Ю. И. Интервальный анализ / Ю. И. Шокин. – Новосибирск: Наука, 1981. – 111 с.

6. Аноприенко А.Я. Интервальные вычисления и перспективы их развития в контексте кодо-логической эволюции / А.Я. Аноприенко, С.В. Иваница // Научные труды ДонНТУ. Серия: Проблемы моделирования и автоматизации проектирования динамических систем (МАП-2010). – 2010. – Вып. 8 (168). – С. 150–160.

7. Марчук Г.И. Методы вычислительной математики / Г.И. Марчук. – М.: Наука. 1977. — 456 с.

8. Самарский А.А. Введение в теорию разностных схем / А.А. Самарский. – М.: Наука, 1971. – 552 с.

9. Шарый С.П. Конечномерный интервальный анализ / С.П. Шарый. – Новосибирск: Институт вычислительных технологий СО РАН, 2009. – 569 с.

10. Аноприенко А.Я. Пример Румпа в контексте традиционных, интервальных и постбинарных вычислений / А.Я. Аноприенко, В.А. Гранковский, С.В. Иваница // Научные труды Донецкого национального технического университета. Серия: Проблемы моделирования и автоматизации проектирования динамических систем (МАП-2011). – 2011. – Вып. 9 (179). – С. 324–343.

Надійшла до редколегії 18.10.2011

С.В. ІВАНИЦЯ, А.В. МЕРКУЛОВ
Донецький національний технічний університет

Методи контролю точності результуючих інтервалів при обчисленні інтервальних поліноміальних функцій

Запропоновано два методи (метод диференціювання і метод заміни інтервалу) контролю точності при обчисленні інтервальних поліноміальних функцій з метою оптимізації результуючих інтервалів, тобто запобігання отримання «шуму» у процесі інтервальних обчислень. Розглянуто результати роботи методів на прикладі розрахунку інтервальних поліномів 2-го і 5-го ступеня. Представлені результати роботи програмної моделі, яка демонструє ефективність використання розглянутих методів оптимізації при інтервальних обчисленнях.

інтервал, інтервальна математика, поліном, одиничний інтервал, верифікація, диференціювання

S.V. IVANITSA, A.V. MERKULOV
Donetsk National Technical University

Methods for Monitoring the Accuracy of the Calculation of Interval Polynomial Functions

We proposed two methods (the method of differentiation and the method of interval replacement) for monitoring the accuracy of calculating interval polynomial functions in order to optimize the resulting intervals that is to avoid making the "noise" in the interval calculations. Results of the methods on the example of calculation of interval polynomials of 2nd and 5th degree are considered. The results of the programming model demonstrating the effectiveness of these methods of optimization in interval computations are presented.

interval, interval mathematics, polynomial, unit interval, verification, differentiation