https://orcid.org/0000-0003-1845-5014
https://orcid.org/0000-0001-5183-816X
https://orcid.org/0000-0003-1351-8178
https://orcid.org/0000-0002-3730-4759

УДК 004.047

О.О. КРЯЖИЧ[*], К.С. ЮЩЕНКО[*], В.Є. ІЦКОВИЧ[*], О.М. КУПРІН[*]

ОСОБЛИВОСТІ АЛГОРИТМІЗАЦІЇ ПРОЦЕСІВ МІНІМІЗАЦІЇ ПОХИБОК АПРОКСИМАЦІЇ ПРИ ВИРІШЕННІ ПРИКЛАДНИХ ЗАДАЧ

[*]Інститут телекомунікацій і глобального інформаційного простору НАН України, м. Київ, Україна

Анотація. *У роботі визначено математичні особливості, які виникають при алгоритмізації мінімізації похибок, що з'являються в результаті наближення функцій у прикладних задачах з екологічної безпеки, створення автоматизованих систем відбору персоналу, вдосконалення систем із рекомендаційним механізмом. Було доведено, що, здійснюючи алгоритмізацію будь-якого процесу, необхідно враховувати ігнорування деякої частини інформації у процесі формалізації. При перетворенні інформації будь-яку подію у процесі функціонування складної системи на певному проміжку часу можна розглядати як виникнення конкретної ситуації в деякій визначеній точці, до якої і висуваються вимоги забезпечення властивостей інформації. Робота базується на базових постулатах теорії Коші та роботах Г.С. Теслера щодо деяких тверджень із розкладання функцій за нев'язками. Використовуючи підхід щодо розв'язання задач із складними варіантами апроксимації функцій та існуючі методи мінімізації похибок при наближеннях, запропоновано алгоритм, який дозволяє у випадку пошуку рішення з великою кількістю ітерацій провести ряд перетворень, що дозволять отримати результат за допомогою розкладання в ряд Тейлора за ступенями. Було виявлено, що при відсутності чіткого рішення поставленої задачі найкраще застосовувати апроксимацію з деякою нев'язкою. Тобто, обирається параметр, який і виступає в ролі елемента адаптації до зазначених умов задачі. Звичайно, такий спосіб вирішення потребує функціональних перетворень, але у підсумку дозволяє використовувати базові тригонометричні функції, які спрощують алгоритмізацію. Був розглянутий приклад практичної реалізації алгоритму з мінімізацією похибок наближення. На основі прикладу доведено, що більш раціональним є використання таблиць ряду широко вживаних функцій та розкладання функцій у ряди за похибками.*
Ключові слова: *крок, функція, формалізація, цикл, обмеження, система, відношення.*

Abstract. *The paper defines the mathematical features arising during the algorithmization of the minimization of errors that appear as a result of the approximation of functions in applied environmental safety problems, the creation of automated personnel selection systems, and the improvement of systems with a recommendation mechanism. It has been proved that when performing the algorithmization of any process, it is necessary to take into account the ignoring of some part of the information in the process of formalization. When transforming information, any event in the process of functioning of a complex system over a certain period of time can be considered as the occurrence of a specific situation at some specified point, to which the requirements for ensuring the properties of information are put forward. The work is based on the basic postulates of the Cauchy theorem and Dr. Tesler's works on some assertions on the decomposition of functions by incoherence. Using the approach to solving problems with complex variants of approximation of functions and existing methods of minimizing errors in approximations, there is proposed an algorithm that in the case of finding a solution with a large number of iterations allows for carrying out a number of transformations that will let obtain the result through expansion in the Taylor series by degrees. It has been found that in the absence of a clear solution to the given problem, it is best to use an approximation with some incoherence. That means choosing a parameter that acts as an element of adaptation to the specified conditions of the task. Of course, this method of problem solving requires functional transformations, but in the end, it allows you to use basic trigonometric functions that simplify algorithmization. An example of a practical implementation of the algorithm with the minimization of approximation errors has been considered. Based on it, it has been proved that it is more rational to use tables of a number of widely used functions and decompose functions into series by errors.*

Keywords: *step, function, formalization, cycle, constraint, system, relation.*

DOI: 10.34121/1028-9763-2023-1-118-129

1. Вступ

Алгоритмізація процесів є математичним описом деякого алгоритму [1], тобто, кроків для вирішення поставленого завдання [2]. Останнє впливає на швидкість алгоритму та забезпечує скінченність як обов’язковість його завершення [3].

Апроксимація широко використовується для наближеного вираження або заміни різноманітних величин за схожими ознаками [4]. Іноді це спрощує процес обчислень, але може призвести до похибок при збільшенні ітерацій – відбувається накопичення неточності, що у підсумку може призвести до зациклення алгоритму [5].

Наприклад, при побудові 3D-резюме [6] використовується деяка функція, яка наближає отриману відповідь до шаблона відповіді, що була закладена у базу даних системи. Це стосується відповідей, які не обираються зі списку, а вносяться письмово. Тоді співставлення відповіді відбувається поступово, визначаючи, наскільки відповідь наближується до заданих критеріїв істинності чи хиби (true or false).

Інший приклад – побудова сервіс-орієнтованої мережі інформування населення у разі виникнення надзвичайних ситуацій, що базується на основі вимірів індикативних параметрів повітря [7]. У цьому випадку прораховуються кроки, що наближають отримані параметри до ймовірної надзвичайної події, щоб запобігти їх розвитку. Помилка наближення може призвести до невірної відповіді і несвоєчасного та невідповідного ситуації рішення.

Існують багато прикладів [8–10] алгоритмізації апроксимації функцій для вирішення прикладних задач, де зазначається, що питання мінімізації похибок є актуальним і потребує пошуку рішення при алгоритмізації процесів. Серед дослідників, що займалися вирішенням цієї задачі, можна назвати Шора Н.З. [11], Пізюра Я.В. [12], Малачівського П.С. [13], Рабіновича Є.В. [14]. Не зважаючи на багаторічні дослідження, питання алгоритмізації апроксимації функцій залишається відкритим через розвиток мов програмування та постійне вдосконалення інформаційних технологій.

Найбільшу цікавість за темою дослідження мають роботи Г.С. Теслера [15–17], в яких представлено не лише деякі складні варіанти апроксимації функцій та їх можливе розв’язання, а й методи мінімізації похибок при наближеннях.

Метою роботи є визначення математичних особливостей, які виникають при алгоритмізації мінімізації похибок, що з’являються в результаті наближення функцій у прикладних задачах (екологічна безпека, автоматизовані системи відбору персоналу, пошукові і рекомендаційні системи та ін.).

Задачі роботи:

– розглянути особливості розкладання за похибками з використанням елементарних функцій;

– представити алгоритмізацію 3D-резюме з мінімізацією похибок наближення відповіді;

– запропонувати алгоритм розкладання шуканої функції в ряд нев’язок.

2. Постановка задачі мінімізації похибок при наближеннях

Здійснюючи алгоритмізацію будь-якого процесу, варто пам’ятати, що формалізація ігнорує деяку частину доступної інформації [18], перетворюючи її на систему формул як інформацію про об’єкт дослідження. В цьому процесі дуже важливо не втратити основні властивості інформації, надаючи перевагу аксіомам і правилам без урахування семантики. При перетворенні інформації будь-яку подію у процесі функціонування складної системи на певному проміжку часу $[t; t']$ можна розглядати як виникнення конкретної ситуації в точці M з умовними координатами $[t_n; t_{n+1}]$. В цій точці M виникнуть певні вимоги до властивостей інформації, а саме:

– актуальність – забезпечення істотності даних для події у точці M $[t_n; t_{n+1}]$;

– вірогідність – відповідність щодо наявності даних про стан системи, які відповідають саме події у вказаний момент часу;

– об'єктивність – система обмежень, яка гарантує, що на отриману інформацію не вплинули ніякі сторонні чинники;

– повнота – отриманої інформації достатньо для того, щоб провести всі розрахунки, оцінити подію у точці M $[t_n; t_{n+1}]$ та прийняти відповідне рішення;

– адекватність – прийняте за наведеними ознаками рішення відповідає реальним потребам складної системи в точці M $[t_n; t_{n+1}]$.

Подібну формалізацію можна представити на прикладі онтологій за допомогою мережевого графа [19]. Таким чином можливо прослідкувати чіткі зв'язки відношення та збереження ієрархії між об'єктами, що забезпечує дотримання структури інформації, яка у підсумку дозволяє адекватно алгоритмізувати процеси обробки. Подібний підхід [20] апробований як у режимі прикладних програм, так і на веб-порталах, зокрема, і для рішення задач щодо мінімізації переходу на непрацююче посилання [21]. В основі онтологічного підходу [19] лежить механізм динамічного формування та використання ієрархій у вигляді таксономій [21]. При алгоритмізації такого підходу відбувається організація інформації, її класифікація та відображення упорядкованої множини інформаційних ресурсів.

Аналізуючи онтологічний граф [19], можна побачити вершини та терми-об'єкти відповідної онтології, пов'язані з цією вершиною. При реалізації алгоритму обробки, щоб отримати рішення слід пройти за ієрархічними відношеннями між різними класами об'єктів, роблячи переходи за заданими зв'язками.

Але припустимо, що непуста множина об'єктів не задовольняє вимогам [19], зокрема, немає визначеної ієрархічної структури скінченної множини понять щодо предмета дослідження, існує деяка вільна інтерпретація понять і відношень, функції інтерпретації не формалізовані, аксіоми не визначені. Тобто, аналогічно [20], масив інформації є необробленим і до початку алгоритмізації процесів обробки слід вирішити задачу щодо забезпечення структури інформації таким чином, щоб вона повністю відповідала вимогам інформаційної системи і могла бути представлена у вигляді певних залежностей, які можна відобразити у блок-схемі алгоритму.

У такому випадку обробку інформації та формалізацію можна провести за допомогою адаптивних алгоритмів, які дозволяють подавати та структурувати інформацію за певними правилами. Для цього на початку алгоритмізації виконується вибірка інформації, яка явно чи неявно відноситься до теми запиту, потім проводяться функціональні перетворення й застосовуються методи породжуваних алгоритмів у системі генерування алгоритмів із використанням нечіткої логіки та побудовою рішень у вигляді лінгвістичних правил-продукцій [21]. Подальший процес алгоритмізації можливий із застосуванням одного з базових методів наближення для отримання результату за максимумом чи мінімумом відповідності.

За таким підходом алгоритм обробки інформації буде утримувати не просто масиви інформації, структуровані за тематикою, а:

– сформовані масиви інформації на засадах символьних перетворень;

– використання як загальних, так і окремих схем виведення інформації на запит;

– переформулювання задач і запитів для виведення максимально повної та різної за структурою інформації за запитом.

У підсумку зазначене призведе до збільшення кількості ітерацій та наближених відповідей, що, у свою чергу, призводить не лише до затримки часу на відшукання рішення, а й підвищує вірогідність неточного результату. У цьому випадку мінімізувати похибки можна шляхом розкладання функцій за нев'язками з використанням базового підходу Г.С. Теслера [15] до прямих та обернених функцій, а також адаптивних апроксимацій [16–17]. У підсумку це дозволить отримати розгалужений алгоритм наближення відповіді з максимально можливою точністю результату.

3. Розкладання за похибками з використанням елементарних функцій

Якщо наявність рішення поставленої задачі точно невідоме, використовується апроксимація рішення з деякою нев'язкою. Прикладом таких невідомих рішень є сумісність відповідей деякого *n*-ого кандидата на посаду масиву можливих відповідей у 3D-резюме. Особливо, коли кандидату на посаду слід набрати бали не менше встановленого порогового рівня, а від точності відповіді залежить кількість отриманих балів. У алгоритмі такої автоматизованої системи відбору персоналу закладено розв'язання функції, яка описує можливий рух до правильної відповіді з використанням нев'язки, адаптує кінцевий результат до умов застосування. Приймаємо, що $y = f(x)$ є функцією, яка описує деякий рух до вірної відповіді. Функція лінійна і описує прямий шлях до єдиної вірної відповіді. Відповідно до [15], величина нев'язки $Z_0 = F(x, y_0)$ характеризує близькість початкового наближення деякої відповіді y_0 до шуканої функції $y = f(x)$. Таким чином, параметр y_0 виступає в ролі елемента адаптації до умов застосування. Чим ближче y_0 до шуканої функції, тим величина нев'язки ближче до нуля, але для цього, як правило, слід зробити деякі функціональні перетворення. За основу для здійснення перетворень можна взяти класичне рівняння Коші для елементарних функцій з єдиним безперервним рішенням:

$$\begin{gathered} f(x+y)=f(x)+f(y), \quad f(x+y)=f(x)\cdot f(y), \\ f(x\cdot y)=f(x)+f(y), \;\; f(x\cdot y)=f(x)\cdot f(y), \;\; D(f)=R. \end{gathered} \tag{1}$$

Але похибка наближення може бути задана і в неявному вигляді. Тобто є багато змінних, і для розв'язання слід розглянути присвоєння одній змінній ролі значення, а інші представити як аргументи. Наприклад, елементи такої алгоритмізації зустрічаються при розробці рекомендаційних систем. На основі вибору з масиву різноманітних переглядів товарів у мережі Інтернет пропонується реклама товару, який на поточний момент може зацікавити потенційного клієнта.

Припустимо, що функція $y = f(x)$ задана в неявному вигляді

$$F(x, y) = 0. \tag{2}$$

Тоді нев'язка рівнянь (1) може бути представлена як

$$Z_0 = F(x, y_0), \tag{3}$$

де y_0 – наближення функції на заданому відрізку $[a, b]$ і $\lim\limits_{y_0 \to y} Z_0 = 0$. Величина похибки може бути отримана шляхом підстановки в вираз (3) величини $y_0 = y(1+\delta_0)$ або $y_0 = y + \Delta_0$, де δ_0, Δ_0 відповідно відносна і абсолютна похибка.

Використовуючи теорію Коші та роботи Г.С. Теслера [18–19] щодо деяких тверджень із розкладання функцій за нев'язками, можна представити функціональні перетворення щодо розкладання за похибками з використанням елементарних функцій.

Приймаємо, що при відповідному виборі структури нев'язки (3) для рівняння Коші (1) ряд елементарних функцій $(1+x)^n, x^a, a^x, \ln x, \arcsin x, \arccos x, arctg\, x$ задовольняють через взаємно обернену функцію рівняння:

$$f(x) = f^{-1}(y_0) \odot \varphi(z_0),$$

де $x_0 = f^{-1}(y_0)$ – взаємно обернена функція по відношенню до $y = f(x)$;

y_0 – наближене значення функції $y = f(x)$ для $x \in [a,b]$;

$\odot$ – знак операції множення для функцій $(1+x)^n, x^a, a^x$, знак складання-віднімання для $\ln x, \arcsin x, \arccos x, arctg\, x$.

Для функцій $(1+x)^n, x^a, a^x, \ln x, \arcsin x, \arccos x, arctg\, x$ функція $\varphi(z_0)$ може бути визначена на основі тотожності $\varphi(z_0) \equiv f(z_0)$, а для інших функцій буде вірним таке твердження:

$$y = x^\alpha \qquad \varphi(z_0) \equiv f\left((1 \pm z_0)^{\pm 1}\right),$$

$$y = a^x \qquad \varphi(z_0) \equiv f(z_0/\ln a),$$

$$y = \ln x \qquad \phi(z_0) \equiv \begin{cases} f\left((1 \pm z_0)^{\pm 1}\right), \\ f\left((1 \pm z_0)/(1 \mp z_0)\right). \end{cases}$$

Зазначене можна довести таким чином. Якщо взяти $y = (1+x)^n$ і застосувати нев'язку типу $z_0 = (1+x)/y_0^{-1/n} - 1$, то буде отримано

$$(1+x)^n = y_0(1+z_0)^n.$$

За підходом [19] можна розглянути й інші зазначені функції.

Але при_відповідному виборі нев'язки з наближенням до шуканого результату рівняння (1) базові елементарні функції $\sin x, \cos x, tg\, x$ будуть задовольняти таким функціональним рівнянням:

$$\begin{aligned} f_1(x) &= f_1(x_0) f_2(z_0) - f_2(x_0) f_1(z_0), \\ f_2(x) &= f_2(x_0) f_2(z_0) + f_1(x_0) f_1(z_0), \end{aligned} \tag{4}$$

де $f_1(x) = \sin x$, $f_2(x) = \cos x$, $f_i(x_0)$ – наближене значення $\sin x$ при $i = 1$ або $\cos x$ при $i = 2$ для $x \in [a,b]$. Для $f(x) = tg\, x$ це виражатиметься таким чином:

$$f(x) = \frac{f(x_0) \mp f(z_0)}{1 \pm f(x_0) f(z_0)} \text{ для } \quad f(x) = tg\, x, \tag{5}$$

де $f(x_0) = y_0$ – наближене значення функції $y = tg\, x$ для $x \in [a,b]$.

Подібне можна розглянути через функцію $y = \sin x$. Для цього можна використати наближення $z_0 = arc\sin y_0 - x$, звідки $\sin x = \sin(arc\sin y_0 - z_0)$. Враховуючи, що $arc\sin y_0 = x_0$ і $y_0 = \sin x_0$, можна записати:

$$\sin(arc\sin y_0 - z_0) = \sin x_0 \cos z_0 - \cos x_0 \sin z_0.$$

Аналогічно, якщо взяти $z_0 = arc\cos y_0 - x$, можна записати таким чином:

$$\cos x = \cos(arc\cos y_0 - z_0) = \cos x_0 \cos z_0 + \sin x_0 \sin z_0 .$$

Система рівнянь (4) буде отримана при однакових значеннях x_0 для $\sin x$ і $\cos x$. Аналогічно доводиться і твердження відносно $tg\, x$.

Наведений аналіз базових тригонометричних функцій обумовлений тим, що інші функції визначаються через перші дві. Для вирішення поставлених задач наближення раціонально застосовувати саме базові функції для спрощення підготовчих операцій у процесі перетворень для спрощення подальшої алгоритмізації.

4. Алгоритмізація 3D-резюме з мінімізацією похибок наближення відповіді

Представимо приклад 3D-резюме зі шляхами можливої відповіді у площині системи координат (рис. 1). Власне, 3D-резюме дозволяє пройти шлях відбору в режимі он-лайн, вибираючи часовий період автоматизованої співбесіди, шлях відповідей на питання та виконання завдань, а також самостійно визначити готовність до наступного етапу співбесіди (часу відкриття системою наступного ступеня завдань). Безпосередньо система може відкривати уточнюючі завдання чи питання, якщо попередні відповіді не дозволяють чітко представити 3D-резюме кандидата на роботу.

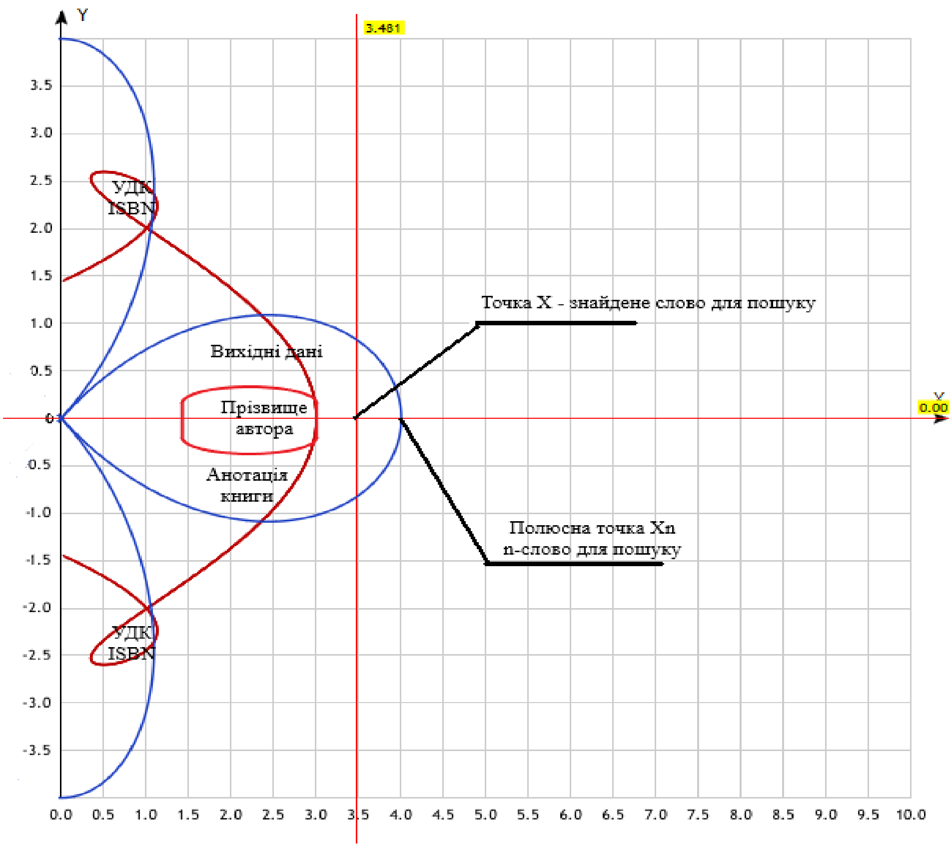

Рисунок 1 – Приклад проходження 3D-резюме в системі координат

Це можна розглянути на прикладі 3D-резюме технічного редактора. Є чотири рівні, представлені питаннями за темами стосовно індексу УДК та ISBN, вхідних даних, анотації та авторства, що є початковими точками для окремих напрямів руху відповідей до заповнення резюме з набором необхідної кількості балів:

– найкраща відповідь – 4 бали;

– відповідь на всі питання із прохідним балом – 3 бали;

– частково вірна відповідь оцінюється в 0,5 бала.

Ламана, обмежена координатою $X = 3$, є моделлю з базисними точками. Від них будується ланцюг із питаннями, які дозволяють розкрити потенціал спеціаліста. Ламана, обмежена координатою $X = 4$, є шляхом отримання максимально точних відповідей із найменшою кількістю ітерацій. Тобто, кожна правильна відповідь (адекватність, результат у точці M $[t_n;\, t_{n+1}]$) приводить до переходу без зайвих ітерацій до наступної точки M $[t_n;\, t_{k+1}]$ для відповіді на питання за іншою темою. У разі неточності відповіді продовжуються питання за розкритою темою, тобто виникають відповіді в точках M_1 $[t_n;\, t_{n+2}]$ … M_m $[t_n;\, t_{n+m}]$.

Розташування цих початкових точок для дослідження масиву можна задати такими рівняннями:

$$x(t) = -2\cos(t) + \cos(4t),$$

$$y(t) = 2\sin(t) + \sin(-2t),$$

де t знаходиться в інтервалі [0; 10]. На графіку (рис. 1) це представлено кривою, що обмежується $X = 3$.

Реалізуючи алгоритм 3D-резюме, знаходяться точки з питаннями, які не підпадають під однозначну відповідь «істина/хиба». Тоді даються письмова відповідь, слова чи фрагменти слів, що можуть виступити як базові слова для співвіднесення з масивом відповідей, які приймаються до аналізу із присвоєнням певної оцінки.

На векторі, побудованому від базової точки «Прізвище автора» (що у прикладі співпадає з віссю x), знаходяться полюсна точка X_n та точка X, яка ілюструє деяке знайдене слово для пошуку, за яким отримано найкращий результат за відповіддю.

Проте рішення подібної задачі викликає дії з великою кількістю ітерацій, що може ускладнити процес алгоритмізації. Крім того, для кожного запиту щодо окремого об'єкта 3D-резюме (питання за темою) потрібно будувати окрему модель або, навіть, моделі опису, виходячи з обраних базових точок. Крім труднощів реалізації, виникатимуть і труднощі в розрахунках, бо зростатиме і помилка наближення до відповідей, яким присвоєно критерій відповідності.

Для реалізації такої задачі більш раціональним є використання таблиць ряду широко вживаних функцій та розкладання функцій у ряди за похибками.

За основу перетворення функцій у ряди за похибками може бути взятий ряд послідовних ітераційних формул, оптимальних за Траубом [22]. Такі функціональні перетворення можуть знадобитися у випадку, коли аналізується неструктурована інформація змін у часі, яка поступає від точок-джерел інформації, що знаходяться у тривимірному просторі. У цьому випадку можна знову звернутися до роботи [17], де проаналізовано підхід до представлення функції $y = arctg\, x$ із заданою нев'язкою.

Традиційно береться деяка функція $y = f(x)$ із точкою x, що належить деякому відрізку $x \in [a,b]$. Цю функцію можна записати в неявному вигляді як (2). Замінивши у виразі (2) шукану функцію наближеним значенням y_0 на заданому інтервалі, отримаємо нев'язку у вигляді (3). Згідно з [15], значення нев'язки (3) для $x \in [a,b]$ при y_0, достатньо наближеному до y, буде достатньо малим для функцій, які є аналітичними в деяких межах точки x_0, і це відповідатиме значенню y_0, тобто:

$$\lim_{y_0 \to y} z_0 = 0.$$

У цьому випадку розкладення в ряд Тейлора-Маклорена за нев'язками z_0 може виявитися досить ефективним для реалізації різних додатків, включаючи веб-додатки з алгоритмами пошуку рекомендацій [23] або сервіс-орієнтованої мережі інформування населення у разі виникнення надзвичайних ситуацій [7].

Разом із (3) можна розглянути такий вираз:

$$z = F(x, y). \tag{6}$$

Для функції (2) вираз (6) може бути представлений рядом способів із критеріями оптимізації. Наприклад, із переходом у тривимірну систему координат рівняння (6) може бути представлене як

$$\Phi(x, y_0, z_0) = 0. \tag{7}$$

Власне функція $\Phi(x, y_0, z_0)$ визначена та безперервна в області

$$D = \left\{x^0 - \Delta_1,\ x^0 + \Delta_1,\ y_0^0 - \Delta_2,\ y_0^0 + \Delta_2,\ z_0^0 - \Delta_3,\ z_0^0 + \Delta_3\right\}$$

з центром у точці $O\ \left(x_1^0,\ y_1^0,\ z_0^0\right)$ Тобто всі правильні відповіді 3D-резюме, за якими набирається прохідний бал, будуть розташовані на кривій, представленій півколом із центром у точці із вказаними координатами.

При цьому приймається:

– частинні похідні $\Phi'_x, \Phi'_y, \Phi'_z$ існують і безперервні в області D;

– функція Φ у точці $\left(x^0, y^0, z_0^0\right)$ обертається на нуль;

– частинна похідна $\Phi'_x\left(x^0, y_0^0, z_0^0\right) \neq 0$.

Використовуючи теорему про неявні функції багатьох змінних, за рівнянням (7) x виступатиме як однозначна функція від $y_0,\ z_0$, тобто

$$x = \varphi(y_0,\ z_0). \tag{8}$$

При $y_0 = y_0^0,\ z_0 = z_0^0$ ця функція приймає значення $x^0 = \varphi\left(y_0^0,\ z_0^0\right)$ і, крім цього, функція $\varphi(y_0,\ z_0)$ неперервна за сукупністю своїх аргументів і має безперервні частинні похідні $\varphi'_{y_0}, \varphi'_{z_0}$.

Базуючись на рівнянні (2), можна записати рівність

$$f(x) = f[\varphi(y_0, z_0)],$$

де $f(x) \equiv y$ – тотожність, що представляє шукану функцію;

$f[\varphi(y_0, z_0)]$ – суперпозиція функцій f та φ.

5. Алгоритм розкладання шуканої функції в ряд нев'язок

Повертаючись до вирішення прикладних задач, варто зазначити, що найчастіше приходиться працювати з масивами неструктурованої інформації, яка є вхідною для рекомендаційних систем та мереж інформування населення у разі виникнення надзвичайних ситуацій. У цьому випадку пошук рішення буде пов'язаний з великою кількістю ітерацій, поки не буде отриманий результат, що задовольняє встановленим обмеженням та дозволяє отримати рішення з найменшою похибкою.

Вирішуючи подібне завдання за допомогою базових тригонометричних функцій, можна розкласти функцію $f[\varphi(y_0, z_0)]$ за ступенями z_0 у межах точки $(y_0, 0)$ до кратного ряду Тейлора з отриманням такого ряду:

$$\begin{aligned} y = f[\varphi(y_0, 0)] &+ \frac{\partial}{\partial z_0}\left(f[\varphi(y_0, z_0)]\Big|_{z_0=0}\right) z_0 + \\ &+ \frac{1}{2!}\frac{\partial^2}{\partial z_0^2}\left(f[\varphi(y_0, z_0)]\Big|_{z_0=0}\right) z_0^2 + \ldots + \\ &+ \frac{1}{k!}\frac{\partial^k}{\partial z_0^k}\left(f[\varphi(y_0, z_0)]\Big|_{z_0=0}\right) z_0^k + \ldots . \end{aligned} \tag{9}$$

Слід зауважити, що члени розкладання в ряд Тейлора функції $f[\varphi(y_0, z_0)]$ з похідними за y в точці $(y_0, 0)$ дорівнюють нулю, бо помножуються на величину $(y - y_0)^k\big|_{y=y_0} = 0$.

У деяких випадках для отримання ряду нев’язок у вигляді

$$y = \psi(y_0)\sum_{k=0}^{\infty} a_k t_0^k,\ y = y_0 + \sum_{k=1}^{\infty} b_k t_0^k \tag{10}$$

робиться заміна $t_0 = \mathrm{H}(y_0, z_0)$. В окремому випадку можливі тотожності $t_0 \equiv z_0$, $\psi(y_0) \equiv y_0$.

Для збіжності отриманого розкладання до функції $f(x)$ у межах точки $(y_0, 0)$ необхідно і достатньо, щоб функція $f[\varphi(y_0, z_0)]$ мала область D безперервних частинних похідних будь-якого порядку і $\lim\limits_{n\to\infty} R_n = 0$, де R_n – залишковий член кратного ряду Тейлора (1).

При значенні y_0, яке дорівнює константі, розкладання (9) та (10) перетворюються у розкладання функції в ряд Тейлора.

На практиці може виникнути необхідність використати обчислення ряду елементарних і спеціальних функцій як безпосередньо з використанням розкладання в ряд нев’язок, так і у вигляді ітераційних формул.

Аналізуючи наведене та роботи [17, 19, 22], можна коротко представити алгоритм розкладання шуканої функції в ряд нев’язок у такому вигляді:

1. Вибирається вид нев’язки $z_0 = F(x, y_0)$. Враховуючи, що $y_0 = y$, повинна виконуватися рівність $z = F(x, y) = 0$.

2. З рівняння нев’язки отримується x у неявному вигляді, тобто $x = \varphi(y_0, z_0)$.

3. Відбувається знаходження функції $f(x) = f(\varphi(y_0, z_0))$.

4. Функція $f(\varphi(y_0, z_0))$ розкладається в ряд Тейлора за ступенями z_0.

5. Отримане розкладання і буде розкладанням шуканої функції в ряд нев’язок. Досліджуємо збіжність отриманого ряду і оцінюємо величину залишкового члена.

Проте, в деяких випадках прикладного застосування такого підходу раціональним буде застосування розкладання в ряд нев’язок із використанням правил диференціювання складної функції.

Тоді алгоритм доповнюється такими кроками:

6. Крок А: береться функція $f(x) = f[\varphi(y_0, z_0)]$. Її розкладання в ряд нев’язок має вид:

$$y = f\left[\varphi(y_0, z_0)\right]\Big|_{Z_0=0} + \frac{\partial}{\partial z_0}\left(f\left[\varphi(y_0, z_0)\right]\Big|_{Z_0=0} + \frac{1}{2!}\frac{\partial^2}{\partial z_0^2}\left(f\left[\varphi(y_0, z_0)\right]\Big|_{Z_0=0}\right)\right) z_0^2 + \dots \ . \quad (11)$$

Скориставшись правилом диференціювання складної функції, формулу (11) можна записати за [15] у вигляді

$$y = f\left[\varphi(y_0, z_0)\right] = f\left[\varphi(y_0, z_0)\right]\Big|_{z_0=0} + \left\{\frac{df(\varphi)}{d\varphi}\frac{\partial\varphi(y_0, z_0)}{\partial z_0}\right\}\Big|_{z_0=0} z_0 +$$

$$+\frac{1}{2!}\left\{\frac{d^2 f(\varphi)}{d\varphi^2}\left[\frac{\partial\varphi(y_0, z_0)}{\partial z_0}\right]^2 + \frac{df(\varphi)}{d\varphi}\left[\frac{\partial^2\varphi(y_0, z_0)}{\partial z_0^2}\right]\right\}\Big|_{z_0=0} z_0^2 +$$

$$+\frac{1}{3!}\left\{\frac{d^3 f(\varphi)}{d\varphi^3}\left[\frac{\partial\varphi(y_0, z_0)}{\partial z_0}\right]^3 + 3\frac{d^2 f(\varphi)}{d\varphi^2}\ \frac{\partial\varphi(y_0, z_0)}{\partial z_0}\ \frac{\partial^2\varphi(y_0, z_0)}{\partial z_0^2} + \frac{df(\varphi)}{d\varphi}\ \frac{\partial^3\varphi(y_0, z_0)}{\partial z_0^3}\right\}\Big|_{z_0=0} z_0^3 +$$

$$+\frac{1}{4!}\left\{\begin{matrix}\dfrac{d^4 f(\varphi)}{\partial\varphi^4}\left[\dfrac{\partial\varphi(y_0, z_0)}{\partial z_0}\right]^4 + 6\dfrac{d^3 f(\varphi)}{\partial\varphi^3}\left[\dfrac{\partial\varphi(y_0, z_0)}{\partial z_0}\right]\dfrac{\partial^2\varphi(y_0, z_0)}{\partial z_0^2} + \dfrac{d^2 f(\varphi)}{d\varphi^2} \\ \left\{3\left[\dfrac{\partial^2\varphi(y_0, z_0)}{\partial z_0^2}\right]^2 + 4\dfrac{\partial\varphi(y_0, z_0)}{\partial z_0}\ \dfrac{\partial^3\varphi(y_0, z_0)}{\partial z_0^3}\right\} + \dfrac{df(\varphi)}{d\varphi}\ \dfrac{\partial^4\varphi(y_0, z_0)}{\partial z_0^4}\end{matrix}\right\}\Big|_{z_0=0} z_0^4 + \dots$$

7. Крок Б: у тих випадках, коли структура нев'язки досить складна і знаходження x з рівняння $F(x, y_0) = Z_0$ важко здійснити, можна скористатися наведеним нижче методом із використанням взаємно обернених функцій.

Для цього поряд з рівнянням (7) варто розглянути рівняння $Z = F(x, y)$, але за способом, схожим до модифікованого метода Чебишова, коли

$$y = \Phi(x, z) = \Phi(x, F(x, y)).$$

Якщо замінити у $F(x, y)$ величину y на наближену до неї $y \approx y_0$ і величину x – на x_0, можна отримати

$$y = \Phi(x_0, F(x, y_0)) = \Phi(x_0, z_0) \quad (12)$$

з урахуванням, що $z_0 = F(x, y_0)$.

Далі можна розкласти функцію $y = \Phi(x, z)$ за нев'язками z_0. Якщо при цьому розкласти функцію $\Phi(x_0, z_0)$ в ряд Тейлора-Маклорена, то буде отримане розкладання, еквівалентне модифікованому методу Чебишова для побудови ітерацій вищих порядків.

У цьому випадку варто визначити умови, за яких існує подання (12). Для цього слід проаналізувати функцію $\lambda(x, y, z) = F(x, y) - z = 0$.

Нехай $\lambda(x, y, z)$ – функція трьох змінних, яка безперервно диференційована в межах точки (x_0, y_0, z_0) і $\lambda(x_0, y_0, z_0) = 0$, $\lambda'(x, y, z) \neq 0$ в точці, що розглядається.

Тоді за теоремою про існування неявної функції є таке число $\xi > 0$, що в ξ-межах точки (x_0, y_0, z_0) рівняння $\lambda(x, y, z) = 0$ визначає однозначну безперервну і диференційовану функцію $y = \Phi(x, z)$, яка обертає це рівняння в тотожність і задовольняє рівність $y_0 = \Phi(x_0, y_0)$.

На цьому алгоритм можна вважати закінченим. Проте варто зазначити, що у деяких випадках шляхом обернення ряду нев'язок можуть бути отримані більш складні конструкції розкладань. Такі розкладання мають більшу стійкість до помилок округлення в порівнянні з безпосередніми обчисленнями відрізків ряду нев'язок.

6. Висновки

Отже, у підсумку проведеної роботи можна зробити такі висновки та узагальнення:

1. Якщо наявність рішення поставленої задачі точно невідоме, використовується апроксимація рішення з деякою нев'язкою. Тобто, обирається параметр y_o, який і виступає в ролі елемента адаптації до зазначених умов задачі. Звичайно, такий спосіб вирішення потребує функціональних перетворень, але у підсумку дозволяє використовувати базові тригонометричні функції, які спрощують алгоритмізацію.

2. Процес алгоритмізації з мінімізацією похибок наближення розглянуто на прикладі 3D-резюме, яке дозволяє пройти шлях відбору персоналу в режимі он-лайн майже так само, як і при очній співбесіді. Виявлено, що для реалізації задачі алгоритмізації резюме більш раціональним є використання таблиць ряду широко вживаних функцій та розкладання функцій у ряди за похибками.

3. Представлено алгоритм, який дозволяє у випадку пошуку рішення з великою кількістю ітерацій провести ряд перетворень, які дозволять отримати результат за допомогою розкладання в ряд Тейлора за ступенями z_o. Це спрощує конструкції для розрахунків та дозволяє отримати результат із більшою стійкістю до помилок округлення.

Наведені особливості алгоритмізації процесів мінімізації похибок апроксимації при вирішенні прикладних задач можуть бути використані при розробці алгоритмів рекомендаційних систем різного спрямування, інформаційних технологій, призначених для відбору персоналу та визначення необхідності підвищення кваліфікації кадрів, а також мереж інформування населення у разі виникнення надзвичайних ситуацій техногенного та природного характеру.

СПИСОК ДЖЕРЕЛ

1. Козлов О.А., Аляев Ю.А. Алгоритмизация и языки программирования: Pascal, C++, Visual Basic. М.: Финансы и статистика. 2004. 320 с. ISBN: 5-279-02294-2.
2. Кормен Т. Алгоритмы: вводный курс / пер. с англ. М.: ООО «И.Д. Вильямс», 2014. 208 с. ISBN 978-5-8459-1868-0.
3. Вирт Н. Алгоритмы и структуры данных / пер. с англ. Ф.В. Ткачева. М.: ДМК Пресс, 2010. 272 с. ISBN 978-5-94074-584-6.
4. Поршнев С.В., Каплан А.В., Каплан В.Е., Мащенко М.В., Овечкина Е.В. Компьютерный анализ и интерпретация эмпирических зависимостей / под ред. С.В. Поршнева. Москва: Бином-Пресс, 2009. 336 с.
5. Дзядык В.К. Об эффективном построении многочленов, которые осуществляют близкое к наилучшему приближение функции. *Украинский математический журнал.* 1973. Вып. 25, № 4. С. 435–453.
6. Ющенко К.С. Підхід до автоматизації процесу підбору кадрів за допомогою 3D-резюме. *Математичні машини і системи*. 2022. № 2. С. 29–39.
7. Кряжич О.О., Іцкович В.Є. Інтернет речей в управлінні складними системами. *Наукові підсумки 2022 рок*у: XI наук. конф.: зб. наук. праць. Х.: Технологічний центр, 2022. С. 19. e-ISBN 978-617-7319-62-6.
8. Kryazhych O.O., Kovalenko O.V. Examining a mathematical apparatus of Z-approximations of function for construction of an adaptive algorithm. *Eastern-European Journal of Enterprise Technologies*. 2019. Vol. 3/4 (99). P. 6–13.
9. Шелудько Г.А., Угримов С.В. Адаптивное кусочно-линейное приближение трудновычислимых функций. *Проблеми машинобудування*. 2018. Т. 21, № 2. С. 60–67.

10. Шинкаренко Г.А., Козаревська Ю.С. Регуляризація чисельних розв'язків варіаційних задач міграції домішок: h-адаптивні МСЕ. Ч. 1. *Вісник Львівського університету. Прикладна математика та інформатика*. 2001. Вип. 5. С. 153–164.
11. Shor N.Z. Minimization methods for non-differentiable functions. Berlin: Springer-Verlag, 1985. 178 p. DOI: https://doi.org/10.1007/978-3-642-82118-9.
12. Пізюр Я.В. Наближення функцій ермітовими сплайнами з експоненціальними ланками. *Вісник НУ «Львівська політехніка». Фізико-математичні науки.* 2007. № 566. С. 68–75.
13. Малачивский П.С., Пизюр Я.В., Данчак Н.В., Оразов Е.Б. Чебышевское приближение экспоненциальным выражением с относительной погрешностью. *Кибернетика и системный анализ*. 2015. Т. 51, № 2. С. 145–150.
14. Рабинович Е.В., Рубан А.А., Цапенко М.П., Шефель Г.С. Адаптивная кусочно-линейная аппроксимация. *Автометрия*. 1993. N 1. С. 26–29.
15. Теслер Г.С. Обобщенные адаптивные аппроксимации функций. *Математические машины и системы*. 1998. № 2. С. 3–8.
16. Теслер Г.С. Адаптивные аппроксимации и итеративные процессы. *Математичні машини і системи*. 2004. № 2. С. 22–41.
17. Теслер Г.С., Гелемб'юк Р.В. Побудова ітераційних формул для розв'язування систем диференціальних рівнянь з використанням розвинення функцій за нев'язками. *Математичні машини і системи*. 2009. № 2. С. 68–75.
18. Энциклопедия кибернетики: в 2 т. / под. ред. В.М. Глушкова и др. К.: Главная редакция украинской советской энциклопедии, 1974. 1228 с.
19. Стрижак О.Є. Засоби онтологічної інтеграції і супроводу розподілених просторових та семантичних інформаційних ресурсів. *Екологічна безпека та природокористування*. 2013. № 12. С. 166–177.
20. Кряжич О.О. Визначення релевантності інформації, отриманої від пошуково-довідкового сервісу на веб-платформі. *Математичні машини і системи*. 2021. № 1. С. 52–63. DOI: https://doi.org/10.34121/1028-9763-2021-1-52-63.
21. Стрижак О.Є. Методика створення онтологічного інтерфейсу у середовищі WEB-порталу. *Радіоелектронні і комп'ютерні системи*. 2014. № 2. С. 78–84.
22. Трауб Дж. Итерационные методы решения уравнений. М.: Мир, 1985. 264 с.
23. Купрін О.М. Алгоритмізація процесів у рекомендаційних системах. *Математичні машини і системи*. 2022. № 1. С. 71–80. ISSN 1028-9763.

Стаття надійшла до редакції 21.01.2023