ВІСНИК НАЦІОНАЛЬНОГО УНІВЕРСИТЕТУ
“ЛЬВІВСЬКА ПОЛІТЕХНІКА”
“Фізико-математичні науки”
Вип.696 № 696, (2011) с. 47–55

JOURNAL OF NATIONAL UNIVERSITY
“LVIVSKA POLITECHNIKA”
“Physical & mathematical sciences”
Vol.696 No 696, (2011) 47–55

ІТЕРАЦІЙНІ АЛГОРИТМИ ТА ГІЛЛЯСТІ ДРОБИ ДЛЯ ФАКТОРИЗАЦІЇ ПОЛІНОМІВ У ЧИСЛОВИХ ПОЛЯХ І БАНАХОВИХ АЛГЕБРАХ

А.Ф. Обшта, Б.А. Шувар

Національний університет “Львівська політехніка”
вул. С. Бандери, 12, 79013, Львів, Україна

(Отримано 9 грудня 2010 р.)

Запропоновані ітераційні алгоритми для апроксимації коренів поліномів у банахових алгебрах, які мають лінійну збіжність, і побудовані відповідні аналоги гіллястих дробів. На їх основі для поліномів з числовими коефіцієнтами запропоновані ітераційні алгоритми з квадратичною збіжністю, які не асоціюються з ньютонівськими методами.

Ключові слова: банахові алгебри, ітераційні алгоритми, гіллясті дроби, лінійна збіжність, квадратична збіжність.

2000 MSC: 12D05/32A65

УДК: 517.948/517.946

1. Застосувати ланцюгові дроби для наближеного розв'язання скалярних рівнянь запропонував 1956 року Ф.Б. Гільдебранд [1, с. 411, 412] (див. також [2, стор. 239]). Очевидною є актуальність вивчення можливостей використовувати задля цього гіллясті дроби В.Я. Скоробогатька (див. [3, 4], а також [5]), які узагальнюють поняття ланцюгових дробів. Запропонований в [6] ітераційний алгоритм для наближеної факторизації поліномів з коефіцієнтами із банахової алгебри з одиницею є водночас способом побудови аналогів гіллястих дробів, компонентами яких є елементи банахової алгебри. Встановлені в [6] оцінки забезпечують лінійну збіжність цього процесу. У частковому випадку отримуються умови існування коренів степеня $m \geq 2$ з елементів банахової алгебри. При $m = 2$ матимемо, зокрема, результат, який за формулюванням і за способом обґрунтування відрізняється як від теореми Віссера (див. [7], а також [8, с. 228–231]) про існування квадратного кореня з лінійного додатного оператора у гільбертовому просторі, так і від теореми У. Рудіна [9, с. 275], якою встановлені достатні умови існування кореня m-го степеня ($m \geq 2$) з елемента банахової алгебри. Зазначимо, що У. Рудін не пропонує конструктивного способу відшукання цього кореня. Згаданий алгоритм із [6] є новим також і як спосіб побудови гіллястих дробів у числових полях.

У цій замітці запропоновані способи побудови послідовних наближень і гіллястих дробів для коренів поліномів у числових полях і банахових алгебрах. Один із цих способів у застосуванні до рівняння

$$f(x) = \theta, \tag{1}$$

у якому $f(x)$ є поліномом вигляду

$$f(x) = x^m + \sum_{i=0}^{m-1} a_i x^i, \tag{2}$$

використовує структуру ітераційних формул із [6].

Якщо коефіцієнти a_i полінома $f(x)$ належать до банахової алгебри E з одиницею I (θ – нульовий елемент в E), то для запропонованих алгоритмів отримаємо результати, котрі охоплюють результати із [6]. Їх можна адаптувати і до ситуації, коли поліном $f(x)$ має вигляд

$$f(x) = x^m + \sum_{i=0}^{m-1} x^i a_i. \tag{3}$$

Крім аналогів побудованих і досліджених у [6] алгоритмів, пропонуємо ітераційний алгоритм, структура якого використовує як основу структуру алгоритму із [6], але принципово відрізняється від нього ітераційними формулами, що мають, зокрема, “гуртовий” характер. Це означає, що структура алгоритму дає змогу скористатися ним тільки за одночасного ітерування всіх коренів полінома $f(x)$. Алгоритм має квадратичну швидкість збіжності, якщо $f(x)$ є поліномом у числовому полі. Однак його не вдається пристосувати для поліномів $f(x)$ із коефіцієнтами з банахової алгебри навіть для випадку $m = 2$.

2. Запропонований в [6] підхід використаємо для побудови та дослідження збіжності ітераційних алгоритмів, обмежуючись ситуацією, коли $f(x)$ є поліномом з комплексними коефіцієнтами.

Задамо попарно числа α_i таким способом, що

$$\sum_{i=1}^{m} \alpha_i = \sum_{i=1}^{m} \beta_i = -a_{m-1}, \tag{4}$$

де β_i є коренями полінома $f(x)$, які задля спрощення викладення вважаємо простими. Позначимо

$$\varphi(x) = \prod_{j=1}^{m} (x - \alpha_i), \tag{5}$$

$$\varphi_i(x) = \frac{\varphi(x)}{(x - \alpha_i)} = \prod_{j=1, j\neq i}^{m} (x - \alpha_i) \quad (i = \overline{1, m}). \tag{6}$$

Будемо вважати, що жодне з чисел α_i не тотожне з жодним із чисел β_i. Подамо рівняння (1) у вигляді

$$x = x - \frac{f(x)}{\varphi_i(x)} \quad (i = \overline{1,m}). \tag{7}$$

Це можна записати також у вигляді

$$x = \alpha_i + \sum_{j=\overline{1,m};j\neq i} \frac{k_{ij}}{x-\alpha_j} \quad (i = \overline{1,m}), \tag{8}$$

де

$$k_{ij} = -\frac{f(\alpha_j)}{\varphi_j'(\alpha_j)}, \tag{9}$$

$$\varphi_i'(\alpha_j) = \prod_{l=\overline{1,m};l\neq i,j} (\alpha_j - \alpha_l). \tag{10}$$

Лема 1. *Нехай поліном $f(x)$ має тільки прості корені β_i, а числа α_i є попарно різними і не тотожні з жодним з коренів β_i. Тоді рівняння (1) рівносильне з кожним з рівнянь (7) та з кожним з рівнянь (8).*

□ *Доведення.* Очевидною є еквівалентність рівняння (1) з рівностями (7) при $x \neq \alpha_i$. Оскільки рівняння (8) отримуються з рівностей (7) шляхом виокремлення цілої частини дробу $\dfrac{f(x)}{\varphi(x)}$ і розкладання остачі на елементарні дроби, то потрібне твердження можна вважати обґрунтованим.

Зафіксуємо значення індексу i й побудуємо послідовність наближень до кореня β_i за допомогою формули

$$x_i^{(n+1)} = \alpha_i + \sum_{j=\overline{1,m};j\neq i} \frac{k_{ij}}{x_i^{(n)} - \alpha_j}, \tag{11}$$

прийнявши

$$x_i^{(0)} = \alpha_i. \tag{12}$$

Замість формули (11) можна користуватися формулою

$$x_i^{(n+1)} = x_i^{(n)} - \frac{f(x_i^{(n)})}{\varphi_i(x_i^{(n)})}, \tag{13}$$

рівносильність якої з (11) перевіряють тривіально. Нехай для $q_i \geq |Q_i(x)|$ маємо

$$Q_i(x) = \sum_{j=\overline{1,m};j\neq i} \frac{f(\alpha_i)}{(\beta_i - \alpha_j)(x-\alpha_j)\varphi_i'(\alpha_j)}. \tag{14}$$

Маючи на увазі співвідношення

$$f(x) = \prod_{j=\overline{1,m}} (x - \beta_i), \tag{15}$$

вираз для $Q_i(x)$ можна переписати у вигляді

$$Q_i(x) = -\sum_{j=\overline{1,m};j\neq i} \frac{\prod\limits_{l=\overline{1,m};l\neq j} (\alpha_j - \beta_l)}{(x-\alpha_j)\prod\limits_{l=\overline{1,m};j,l\neq i} (\alpha_j - \alpha_l)}.$$

■

Теорема 1. *Нехай справджуються зазначені припущення, тобто корені полінома $f(x)$ мають одиничну кратність і задані попарно різні комплексні числа α_i, жодне з яких не тотожне з жодним з коренів β_i, причому мають місце рівності (4). Якщо для зафіксованого значення індексу i маємо*

$$q_i < 1, \tag{16}$$

то послідовність $\left\{x_i^{(n)}\right\}$ збігається до кореня β_i не повільніше від геометричної прогресії зі знаменником q_i.

□ *Доведення.* Співвідношення (8) – (11) після очевидних процедур призводять до рівності

$$x_i^{(n+1)} - \beta_i = Q_i(x_i^{(n)})(x_i^{(n)} - \beta_i), \tag{17}$$

яка є підставою для того, щоб вважати теорему доведеною. ■

3. Якщо $x_i^{(0)}$ вибрати за формулою (12), то при фіксованому значенні індексу i реалізацію ітераційного процесу (11) можна розглядати як спосіб побудови гіллястого дробу В.Я. Скоробогатька, яким апроксимуємо корінь β_i полінома $f(x)$. Отримується гіллястий дріб з $(m-1)$ розгалуженнями на кожному його поверсі. Цей дріб має раціональні компоненти і є періодичним з одиничним періодом. Збіжність гіллястого дробу до кореня β_i випливає з теореми 1.

4. Теорему 1 можна використати для отримання інших докладніших оцінок збіжності процесу. Використовуючи формули (13), запишемо

$$x_i^{(n+1)} - \beta_i = x_i^{(n)} - \beta_i - \frac{f(x_i^{(n)})}{\varphi_i(x_i^{(n)})}. \tag{18}$$

Звідси, маючи на увазі формули (6), знаходимо

$$x_i^{(n+1)} - \beta_i = x_i^{(n)} - \beta_i - \frac{\prod\limits_{j=\overline{1,m}} (x_i^{(n)} - \beta_j)}{\prod\limits_{j=\overline{1,m};j\neq i} (x_i^{(n)} - \alpha_j)},$$

тобто,

$$x_i^{(n+1)} - \beta_i = (x_i^{(n)} - \beta_i)\left(1 - \frac{\prod\limits_{j=\overline{1,m};j\neq i} (x_i^{(n)} - \beta_j)}{\prod\limits_{j=\overline{1,m};j\neq i} (x_i^{(n)} - \alpha_j)}\right).$$

Отже, можна записати

$$\begin{aligned} Q_i(x_i^{(n)}) &= 1 - \prod_{j=\overline{1,m};j\neq i} \frac{x_i^{(n)} - \beta_j}{x_i^{(n)} - \alpha_j} = \\ &= 1 - \prod_{j=\overline{1,m};j\neq i} \left(1 + \frac{\alpha_j - \beta_j}{x_i^{(n)} - \alpha_j}\right). \end{aligned} \tag{19}$$

Нехай числа α_i при $j = \overline{1,m}$ фігурують як початкові наближення до коренів β_i відповідно. Припустимо, що

$$|\alpha_j - \beta_j| \leq \varepsilon \quad (j = \overline{1,m}), \tag{20}$$

$$|x_i^{(n)} - \alpha_j| \geq h(m-1) \quad (i \neq j; i,j = \overline{1,m}). \tag{21}$$

У цьому випадку, враховуючи (15), (16), (19), можна одержати оцінку

$$q < e^{\frac{\varepsilon}{h}}. \tag{22}$$

Очевидно, що оцінка (22) тим більше є завищеною, чим меншим є степінь m полінома $f(x)$. Наприклад, для $m = 2$ формули (19) матимуть вигляд

$$\begin{aligned} Q_1(x_1) &= 1 - \left(1 + \frac{\alpha_2 - \beta_2}{x_1 - \alpha_2}\right) = \frac{\alpha_2 - \beta_2}{x_1 - \alpha_2}, \\ Q_2(x_2) &= 1 - \left(1 + \frac{\alpha_1 - \beta_1}{x_2 - \alpha_1}\right) = \frac{\alpha_1 - \beta_1}{x_2 - \alpha_1}. \end{aligned} \tag{23}$$

Маючи на увазі (20), (21), замість оцінки (22) можна отримати менш грубі оцінки, які мають простий вигляд для невисоких значень m. Однак зі збільшенням m ці оцінки надто громіздкі. Якщо $m = 2$, замість оцінки (22) із співвідношень (23) можна отримати

$$q \le \frac{\varepsilon}{h}. \tag{24}$$

Для $m = 3$ формули (19) записуються в вигляді

$$\begin{aligned} Q_1(x_1) &= 1 - \left(1 + \frac{\alpha_2 - \beta_2}{x_1 - \alpha_2}\right)\left(1 + \frac{\alpha_3 - \beta_3}{x_1 - \alpha_3}\right), \\ Q_2(x_2) &= 1 - \left(1 + \frac{\alpha_1 - \beta_1}{x_2 - \alpha_1}\right)\left(1 + \frac{\alpha_3 - \beta_3}{x_2 - \alpha_3}\right), \\ Q_3(x_3) &= 1 - \left(1 + \frac{\alpha_1 - \beta_1}{x_3 - \alpha_1}\right)\left(1 + \frac{\alpha_2 - \beta_2}{x_3 - \alpha_2}\right). \end{aligned} \tag{25}$$

Звідси замість оцінки (22) одержимо також оцінку

$$q \le \frac{\varepsilon}{h}\left(1 + \frac{\varepsilon}{4}\right). \tag{26}$$

5. Проілюструємо на прикладах застосування алгоритму (11), (12). Докладність проміжних обчислень і докладність їх записування зумовлена, зокрема, доцільністю порівняння структури ітераційних формул (11), (12) і подробиць реалізації відповідних обчислювальних процедур зі структурою запропонованих нижче ітераційних формул з квадратичною швидкістю збіжності і специфікою їх обчислювальної реалізації.

Приклад 1. Нехай $f(x) = (x-1)(x-5)$. Маючи на увазі, що $\beta_1 = 1$, $\beta_2 = 5$, приймемо $\alpha_1 = 0,5$, $\alpha_2 = 5,5$ і зазначимо, що умови (4) дотримано. Можна прийняти $\varepsilon = 0,5$, $h = 4,5$. Оцінка (24) має вигляд $q \le \dfrac{0,5}{4,5} = \dfrac{1}{9}$, а формули (11) – вигляд

$$x_1^{(n+1)} = \alpha_1 - \frac{f(\alpha_2)}{x_1^{(n)} - \alpha_2}, \quad x_2^{(n+1)} = \alpha_2 - \frac{f(\alpha_1)}{x_2^{(n)} - \alpha_1}.$$

Обчислюємо

$$x_1^{(1)} = 0,95; x_2^{(1)} = 5,05; x_1^{(2)} = 0,9945054;$$

$$x_2^{(2)} = 5,0054946; x_1^{(3)} = 0,9993902; x_2^{(3)} = 5,0006098.$$

Наведені обчислення підтверджують практичну збіжність послідовностей $\left\{x_1^{(n)}\right\}$, $\left\{x_2^{(n)}\right\}$ до коренів β_1, β_2 відповідно. Ця збіжність не повільніша від збіжності геометричної прогресії зі знаменником $q = \dfrac{1}{9}$, яку гарантує нерівність (24). Кожна з цих послідовностей генерує нескінченний з одиничним періодом ланцюговий дріб з раціональними компонентами. Отже, для коренів β_1, β_2 цього полінома матимемо

$$\beta_1 = 0,5 - \cfrac{2,25}{0,5 - 5,5 - \cfrac{2,25}{0,5 - 5,5 - ...}},$$

$$\beta_2 = 5,5 - \cfrac{2,25}{5,5 - 0,5 - \cfrac{2,25}{5,5 - 0,5 - ...}}.$$

Безпосередніми підрахунками можна підтвердити, що підхідні дроби цих ланцюгових дробів тотожні з відповідними ітераціями.

Приклад 2. Для полінома $f(x) = (x-2)(x-4)(x--10)$ приймемо $\alpha_1 = 1,5$, $\alpha_2 = 5$, $\alpha_3 = 16 - \alpha_1 - \alpha_2 = 9,5$, $h = \dfrac{5-1,5}{2}$, $\varepsilon = 1$, $q = \dfrac{5}{7}$. Очевидно, що рівність (4) справджується. Послідовні наближення $x_1^{(n)}$, $x_2^{(n)}$, $x_3^{(n)}$, до коренів $\beta_1 = 2$, $\beta_2 = 4$, $\beta_3 = 10$ відповідно обчислюватимемо за формулами (11), маючи на увазі, що $f(\alpha_1) = f(1,5) = -10,625$; $f(\alpha_2) = f(5) = -15$; $f(\alpha_3) = f(9,5) = 20,625$;

$$\varphi_1(x) = (x-\alpha_2)(x-\alpha_3) = (x-5)(x-9,5);$$

$$\varphi_2(x) = (x-\alpha_1)(x-\alpha_3) = (x-1,5)(x-9,5);$$

$$\varphi_3(x) = (x-\alpha_1)(x-\alpha_2) = (x-1,5)(x-5);$$

$$\varphi_1'(\alpha_2) = -4,5; \varphi_1'(\alpha_3) = 4,5;$$

$$\varphi_2'(\alpha_1) = -8; \varphi_2'(\alpha_3) = 8;$$

$$\varphi_3'(\alpha_1) = -3,5; \varphi_3'(\alpha_2) = 3,5.$$

Ітераційні формули мають вигляд

$$x_1^{(n+1)} = \alpha_1 - \frac{f(\alpha_2)}{\left(x_1^{(n)} - \alpha_2\right)(\alpha_2 - \alpha_3)} - \frac{f(\alpha_3)}{\left(x_1^{(n)} - \alpha_3\right)(\alpha_3 - \alpha_2)},$$

$$x_2^{(n+1)} = \alpha_2 - \frac{f(\alpha_1)}{\left(x_2^{(n)} - \alpha_1\right)(\alpha_1 - \alpha_3)} - \frac{f(\alpha_3)}{\left(x_2^{(n)} - \alpha_3\right)(\alpha_3 - \alpha_1)},$$

$$x_3^{(n+1)} = \alpha_3 - \frac{f(\alpha_1)}{\left(x_3^{(n)} - \alpha_1\right)(\alpha_1 - \alpha_2)} - \frac{f(\alpha_2)}{\left(x_3^{(n)} - \alpha_2\right)(\alpha_2 - \alpha_1)}.$$

Отже,

$$x_1^{(n+1)} = 1,5 - \frac{-15}{\left(x_1^{(n)}-5\right)(-4,5)} - \frac{20,625}{\left(x_1^{(n)}-9,5\right)4,5},$$

$$x_2^{(n+1)} = 5 - \frac{-10,625}{\left(x_1^{(n)}-1,5\right)(-8)} - \frac{-20,625}{\left(x_1^{(n)}-9,5\right)8},$$

$$x_3^{(n+1)} = 9,5 - \frac{-10,625}{\left(x_1^{(n)}-1,5\right)(-3,5)} - \frac{-15}{\left(x_1^{(n)}-5\right)3,5}.$$

Для коренів $\beta_1 = 2$, $\beta_2 = 4$, $\beta_3 = 10$ матимемо такі гіллясті дроби:

$$\beta_1 = 1,5 - \cfrac{\cfrac{-15}{-4,5}}{-5+1,5-\cfrac{\cfrac{-15}{-4,5}}{-5+1,5-...}-\cfrac{\cfrac{-20,625}{4,5}}{-9,5+1,5-...}} - \cfrac{\cfrac{-20,625}{4,5}}{-9,5+1,5-\cfrac{\cfrac{-15}{-4,5}}{-5+1,5-...}-\cfrac{\cfrac{-20,625}{4,5}}{-9,5+1,5-...}},$$

$$\beta_2 = 5 - \cfrac{\cfrac{-10,625}{-8}}{-1,5+5-\cfrac{\cfrac{-10,625}{-8}}{-1,5+5-...}-\cfrac{\cfrac{-20,625}{4,5}}{-9,5+5-...}} - \cfrac{\cfrac{-20,625}{4,5}}{-9,5+5-\cfrac{\cfrac{-10,625}{-8}}{-1,5+5-...}-\cfrac{\cfrac{-15}{3,5}}{-9,5+5-...}},$$

$$\beta_3 = 9,5 - \cfrac{\cfrac{-10,625}{-8}}{-1,5+9,5-\cfrac{\cfrac{-10,625}{-8}}{-1,5+9,5-...}-\cfrac{\cfrac{-15}{3,5}}{-5+9,5-...}} - \cfrac{\cfrac{-15}{3,5}}{-5+9,5-\cfrac{\cfrac{-10,625}{-8}}{-1,5+9,5-...}-\cfrac{\cfrac{-15}{3,5}}{-5+9,5-...}}.$$

Наведемо результати обчислення перших трьох ітерацій. Маємо

$$x_1^{(1)} = 1,8794693; x_2^{(1)} = 4,0476191; x_3^{(1)} = 10,072916;$$

$$x_1^{(2)} = 1,9667489; x_2^{(2)} = 4,0058361; x_3^{(2)} = 9,990722;$$

$$x_1^{(3)} = 1,9905253; x_2^{(3)} = 4,0007394; x_3^{(3)} = 10,001203.$$

Можливе погіршення результатів обчислень зі збільшенням ітераційного номера n. Під час реалізації ітераційного процесу (11) за прийнятного вибору α_1, α_2, α_3 і гарантованої теоретичної збіжності ітерацій зі швидкістю геометричної прогресії зі знаменником $q = \frac{\varepsilon}{h}\left(1+\frac{\varepsilon}{4}\right) = \frac{5}{7}$, похибки заокруглень можуть призводити до обчислювальної нестійкості ітерацій. Одним із можливих способів уникнути перетворення теоретично збіжного процесу (11) в практично збіжний можна вважати використання рівності (4) на кожному ітераційному кроці. Вимагатимемо, щоб рівність (4) справджувалася при $\alpha_1 = x_1^{(n)}$, $\alpha_2 = x_2^{(n)}$, $\alpha_3 = x_3^{(n)}$. Тобто, при кожному n або бодай для тих n, коли "надто грубо" порушується рівність

$$x_1^{(n)} + x_2^{(n)} + ... + x_m^{(n)} = -a_{m-1} \quad (m=3) \qquad (27)$$

отримані ітерації змінюватимемо, взагалі кажучи, довільним способом ("у межах доцільності") задля дотримання рівності (27). Схожа ситуація може траплятися й під час застосування інших ітераційних методів. Це стосується, зокрема, до поширених у багатьох прикладних науках методів декомпозиції, що використовують агрегаційно-ітеративні принципи, досліджені, наприклад, у [11] (див [11, Розділ XIII]). Одним з оправдань одночасного обчислення ітерацій (11) до всіх коренів полінома $f(x)$ у прикладі 2 є доцільність використань співвідношень (27) для забезпечення практичної збіжності цих ітерацій.

За суттю, це означає, що у цьому прикладі можливість поліпшити ситуацію щодо збіжності ітераційного процесу (11) грунтується на виборі $x^{(0)} = \{\alpha_1, \alpha_2, \alpha_3\}$ таким способом, щоб числа $\alpha_1, \alpha_2, \alpha_3$ були "достатньо" близькими до коренів $\beta_1 = 2, \beta_2 = 4, \beta_3 = 10$. Тому замість того, щоб підправляти ітерації $x^{(3)} = \left\{x_1^{(3)}, x_2^{(3)}, x_3^{(3)}\right\}$ можна скористатися з довільно вибраних значень $\alpha_1, \alpha_2, \alpha_3$, які задовольняють рівність (4) і відрізняються від вибраних раніше у цьому прикладі тим, що "краще" наближаються до коренів $\beta_1 = 2, \beta_2 = 4, \beta_3 = 10$. Очевидно, що можна прийняти $\alpha_1 = 1,8; \alpha_2 = 4,5; \alpha_3 = 9,7$, а також $\varepsilon = 0,5; h = \frac{9}{4}; q = \frac{1}{4}$. У такому разі матимемо

$$x_1^{(1)} = 1,9691514; x_2^{(1)} = 4,0103277; x_3^{(1)} = 10,020521;$$

$$x_1^{(2)} = 1,9948661; x_2^{(2)} = 4,0007146; x_3^{(2)} = 9,998686;$$

$$x_1^{(3)} = 1,9991353; x_2^{(3)} = 4,0000308; x_3^{(3)} = 10,000084.$$

При цьому $x_1^{(3)}+x_2^{(3)}+x_3^{(3)} = 15,999250$. Обидві ситуації у цьому прикладі у першому випадку при $q = \frac{5}{7}$, а у другому – при $q = \frac{1}{4}$ підтверджується доцільність використовувати рівність (27) як контрольну при "туртовому" застосуванні алгоритму (11), водночас ітеруючи всі корені полінома.

Приклад 3. Застосуємо алгоритм (11), (12) до полінома з комплексними коренями $f(x) = (x^2+1)(x^2+4) = (x+i)(x-i)(x+2i)(x-2i)$. Позначимо $\beta_1 = -i$, $\beta_2 = i$, $\beta_3 = -2i$, $\beta_4 = 2i$. Приймемо відповідно $\alpha_1 = -0,9i$, $\alpha_2 = 0,9i$, $\alpha_3 = -2,1i$, $\alpha_4 = 2,1i$.

Обчислимо $f(\alpha_1) = 0,6061$; $f(\alpha_2) = 0,6061$; $f(\alpha_3) = 1,3981$; $f(\alpha_4) = 1,3981$;

$$\varphi_1(x) = (x-\alpha_2)(x-\alpha_3)(x-\alpha_4) = (x-0,9i)(x^2+4,41);$$

$$\varphi_2(x) = (x-\alpha_1)(x-\alpha_3)(x-\alpha_4) = (x+0,9i)(x^2+4,41);$$

$$\varphi_3(x) = (x-\alpha_1)(x-\alpha_2)(x-\alpha_4) = (x^2+0,81)(x-2,1i);$$

$$\varphi_4(x) = (x-\alpha_1)(x-\alpha_2)(x-\alpha_3) = (x^2+0.81)(x^2+2,1i).$$

$$\varphi_1'(\alpha_2) = (\alpha_2-\alpha_3)(\alpha_2-\alpha_4) = 3,6;$$
$$\varphi_1'(\alpha_3) = (\alpha_3-\alpha_2)(\alpha_3-\alpha_4) = -12,6;$$

$$\varphi_1'(\alpha_4) = (\alpha_4-\alpha_2)(\alpha_4-\alpha_3) = -8,82;$$
$$\varphi_2'(\alpha_1) = (\alpha_1-\alpha_3)(\alpha_1-\alpha_4) = 3,6;$$

$$\varphi_2'(\alpha_3) = (\alpha_3-\alpha_1)(\alpha_3-\alpha_4) = -8,82;$$
$$\varphi_2'(\alpha_4) = (\alpha_4-\alpha_1)(\alpha_4-\alpha_3) = -12,6;$$

$$\varphi_3'(\alpha_1) = (\alpha_1-\alpha_2)(\alpha_1-\alpha_4) = -5,4;$$
$$\varphi_3'(\alpha_2) = (\alpha_2-\alpha_1)(\alpha_2-\alpha_4) = 2,16;$$

$$\varphi_3'(\alpha_4) = (\alpha_4-\alpha_1)(\alpha_4-\alpha_2) = -3,6;$$
$$\varphi_4'(\alpha_1) = (\alpha_1-\alpha_2)(\alpha_1-\alpha_3) = 2,16;$$

$$\varphi_4'(\alpha_2) = (\alpha_2-\alpha_1)(\alpha_2-\alpha_3) = -5,4;$$
$$\varphi_4'(\alpha_3) = (\alpha_3-\alpha_1)(\alpha_3-\alpha_2) = -3,6.$$

Отримуємо такі значення для перших двох ітерацій:

$$x_1^{(1)} = -0,9935339i; x_2^{(1)} = 0,9935339i;$$
$$x_3^{(1)} = -2,0075331i; x_4^{(1)} = 2,0075331i;$$

$$x_1^{(2)} = -0,9995256i; x_2^{(2)} = 0,9995256i;$$
$$x_3^{(2)} = -2,0006170i; x_4^{(2)} = 2,0006170i.$$

6. Реалізація описаного алгоритму, поданого як у вигляді (11), так і у вигляді (13) з початковим наближенням (12), взагалі кажучи, формально не потребує обчислення ітерацій $x_i^{(n)}$ водночас для всіх $i = \overline{1,m}$, забезпечуючи лінійний характер збіжності індивідуально кожної з послідовностей $\left\{x_i^{(n)}\right\}$ до відповідного кореня β_i полінома $f(x)$. Однак, як засвідчує приклад 2, може бути доцільним використовувати ітерації (11) гуртом для всіх коренів полінома $f(x)$. Використання гуртового характеру ітераційного процесу дає змогу скористати з ідеї корегування ітерацій з використанням співвідношень (27).

Пропонуємо інший ітераційний алгоритм для апроксимації коренів β_i полінома $f(x)$, для побудови якого використаємо видозміну ітераційних формул (11) та (13). Формули (5), (6) замінимо формулами

$$\varphi^{(n)}(x) = \prod_{j=\overline{1,m}} (x - \beta_j^{(n)}), \tag{28}$$

$$\varphi_i^{(n)}(x) = \frac{\varphi^{(n)}(x)}{(x-\beta_i^{(n)})} = \prod_{j=\overline{1,m};j\neq i} (x - \beta_j^{(n)}), \tag{29}$$

будуючи послідовності $\{\beta_j^{(n)}\}$ $(i = \overline{1,m})$ за допомогою формул

$$\beta_i^{(n+1)} = \beta_i^{(n)} - \frac{f(\beta_i^{(n)})}{\varphi_i^{(n)}(\beta_i^{(n)})} \quad (i = \overline{1,m}) \tag{30}$$

та приймаючи

$$\beta_i^{(0)} = \alpha_i \quad (i = \overline{1,m}) \tag{31}$$

і припускаючи, що справджується умова (4). Нехай, як і раніше, поліном $f(x)$ має прості корені β_i, тобто,

$$f(x) = \prod_{j=\overline{1,m}} (x - \beta_j).$$

Формули (30) подамо у вигляді

$$\beta_i^{(n+1)} = \beta_i^{(n)} - \frac{\prod\limits_{j=\overline{1,m}} (\beta_i^{(n)} - \beta_j)}{\prod\limits_{j=\overline{1,m};j\neq i} (\beta_i^{(n)} - \beta_j^{(n)})}, \tag{32}$$

тобто,

$$\beta_i^{(n+1)} = \beta_i^{(n)} - (\beta_i^{(n)} - \beta_i) \prod_{j=\overline{1,m};j\neq i} \frac{(\beta_i^{(n)} - \beta_j)}{(\beta_i^{(n)} - \beta_j^{(n)})}. \tag{33}$$

Віднімемо β_i від обидвох частин рівності (33) і запишемо

$$\beta_i^{(n+1)} - \beta_i =$$
$$= (\beta_i^{(n)} - \beta_i)(1 - \prod_{j=\overline{1,m};j\neq i} \frac{(\beta_i^{(n)} - \beta_j^{(n)} + \beta_j^{(n)} - \beta_j)}{(\beta_i^{(n)} - \beta_j^{(n)})}).$$

Отже,

$$\beta_i^{(n+1)} - \beta_i = (\beta_i^{(n)} - \beta_i)(1- \\ - \prod_{j=\overline{1,m};j\neq i} (1 + \frac{\beta_j^{(n)} - \beta_j}{\beta_i^{(n)} - \beta_j^{(n)}})) \quad (i = \overline{1,m}). \tag{34}$$

Рівності (34) використаємо для оцінки збіжності ітераційного процесу (30), (31). Маємо на увазі, що на кожному кроці цього процесу знаходимо $\beta_1^{(n+1)}$, $\beta_2^{(n+1)}$, ..., $\beta_m^{(n+1)}$, тобто, співвідношення (30) і (31) розглядаємо як систему рівностей для $i = \overline{1,m}$. Позначимо

$$\varepsilon_n = \max_{j=\overline{1,m}} \left|\beta_j^{(n)} - \beta_j\right| \tag{35}$$

для кожного окремого натурального n. Припустимо також, що

$$\left|\beta_i^{(n)} - \beta_j\right| \geq h \quad (i,j = \overline{1,m};\ i \neq j). \tag{36}$$

За цієї ситуації з (34) випливають оцінки

$$\left|\beta_i^{(n+1)} - \beta_i\right| \leq \varepsilon_n[(1 + \frac{\varepsilon_n}{h})^{m-1} - 1] \quad (i = \overline{1,m}). \tag{37}$$

Отже,

$$\varepsilon_{n+1} \leq \varepsilon_n M, \quad M > (1 + \frac{\varepsilon_n}{h})^{m-1} - 1. \tag{38}$$

Можна прийняти також, що

$$M \geq e^{\frac{m-1}{h}} - 1. \tag{39}$$

Оцінка (39) є тим грубшою, чим нижчим є степінь m полінома $f(x)$. Очевидно, що з (37) випливає також оцінка

$$\left|\beta_i^{(n+1)} - \beta_i\right| \leq \varepsilon_n^2 M_0 \quad (i = \overline{1,m}), \tag{40}$$

у якій число M_0 можна очевидним способом оцінити зверху. Оцінка (40) характеризує квадратичну швидкість збіжності кожної з послідовностей $\left\{\beta_i^{(n)}\right\}$ до відповідного кореня β_i полінома $f(x)$. Оформимо у вигляді окремого твердження отриманий результат.

Теорема 2. *Якщо $M < 1$, де M означено за допомогою співвідношень (35) – (38), то ітераційний процес (31),(32) при кожному $i = \overline{1,m}$ збігається відповідно до коренів β_i полінома (2). При цьому мають місце оцінки (38) і (39), а також оцінки (40), які характеризують квадратичну швидкість збіжності кожної з послідовностей $\left\{\beta_i^{(n)}\right\}$ $(i = \overline{1,m})$.*

7. При $m = 2$ рівності (34) мають вигляд

$$\beta_1^{(n+1)} - \beta_1 = (\beta_1^{(n)} - \beta_1)(\beta_2^{(n)} - \beta_2)\frac{1}{\beta_2^{(n)} - \beta_1^{(n)}},$$

$$\beta_2^{(n+1)} - \beta_2 = (\beta_1^{(n)} - \beta_1)(\beta_2^{(n)} - \beta_2)\frac{1}{\beta_1^{(n)} - \beta_2^{(n)}}.$$

Позначивши $\left|\beta_2^{(n)} - \beta_1^{(n)}\right| = h_n$ і припустивши, що $h_n \geq h$, матимемо очевидні оцінки

$$\left|\beta_1^{(n+1)} - \beta_1\right| \leq \varepsilon_n^2 \frac{1}{h}; \quad \left|\beta_2^{(n+1)} - \beta_2\right| \leq \varepsilon_n^2 \frac{1}{h}.$$

Для $m = 3$ рівності (34) запишемо у вигляді

$$\beta_1^{(n+1)} - \beta_1 = (\beta_1^{(n)} - \beta_1)(1 - (1+ \\ +\frac{\beta_2^{(n)} - \beta_2}{\beta_1^{(n)} - \beta_2^{(n)}})(1 + \frac{\beta_3^{(n)} - \beta_3}{\beta_1^{(n)} - \beta_3^{(n)}})),$$

$$\beta_2^{(n+1)} - \beta_2 = (\beta_2^{(n)} - \beta_2)(1 - (1+ \\ +\frac{\beta_1^{(n)} - \beta_1}{\beta_2^{(n)} - \beta_1^{(n)}})(1 + \frac{\beta_3^{(n)} - \beta_3}{\beta_2^{(n)} - \beta_3^{(n)}})),$$

$$\beta_3^{(n+1)} - \beta_3 = (\beta_3^{(n)} - \beta_3)(1 - (1+ \\ +\frac{\beta_1^{(n)} - \beta_1}{\beta_3^{(n)} - \beta_1^{(n)}})(1 + \frac{\beta_2^{(n)} - \beta_2}{\beta_3^{(n)} - \beta_2^{(n)}})).$$

Звідси, очевидно, що матимемо оцінку, яка характеризує квадратичну швидкість збіжності при $m = 3$:

$$\varepsilon_{n+1} \leq \varepsilon_n^2 \frac{1}{h}(2 + \frac{\varepsilon}{h}).$$

Аналогічно для $m = 4$ матимемо

$$\varepsilon_{n+1} \leq \varepsilon_n^2 \frac{1}{h}(3 + 3\frac{\varepsilon}{h} + \frac{\varepsilon^2}{h^2}).$$

Подібним способом можна отримати відповідні оцінки для $m > 4$, засвідчуючи надлінійну швидкість збіжності ітераційного алгоритму (22), (31), якщо на кожному кроці процесу реалізувати алгоритм водночас для всіх $i = \overline{1,m}$. Наведена оцінка для ε_{n+1} при $m = 2$ та $m = 3, 4$ ілюструє наростання її громіздкості у разі збільшення m.

8. Практичну ефективність алгоритму (30), (31) порівняно з алгоритмом (12), (13) проілюструємо, зіставляючи результати обчислень з отриманими числовими результатами у прикладах 1 – 3.

Приклад 4. Для полінома $f(x) = x^2 - 6x + 5 = (x-1)(x-5)$ з прикладу 1, для якого приймемо $\beta_1^{(0)} = \alpha_1 = 0.5$, $\beta_2^{(0)} = \alpha_2 = 5.5$, можна вважати, що $\varepsilon_0 = 0.5$; $h = 4.5$. Обчислимо за формулами (30):

$$\beta_1^{(1)} = \beta_1^{(0)} - \frac{(\beta_1^{(0)})^2 - 6\beta_1^{(0)} + 5}{\beta_1^{(0)} - \beta_2^{(0)}} = \\ = 0.5 - \frac{5.5^2 - 60.5 + 5}{0.5 - 4.5} = 1.0625,$$

$$\beta_2^{(1)} = \beta_2^{(0)} - \frac{(\beta_2^{(0)})^2 - 6\beta_2^{(0)} + 5}{\beta_2^{(0)} - \beta_1^{(0)}} = \\ = 4.5 - \frac{4.5^2 - 64.5 + 5}{4.5 - 0.5} = 4.9375,$$

$$\beta_1^{(2)} = \beta_1^{(1)} - \frac{(\beta_1^{(1)})^2 - 6\beta_1^{(1)} + 5}{\beta_1^{(1)} - \beta_2^{(1)}} = \\ = 1.0625 - \frac{1.0625^2 - 61.0625 + 5}{1.0625 - 4.9375} = 0.9998992,$$

$$\beta_2^{(2)} = \beta_2^{(1)} - \frac{(\beta_2^{(1)})^2 - 6\beta_2^{(1)} + 5}{\beta_2^{(1)} - \beta_1^{(1)}} = \\ = 4.9375 - \frac{4.9375^2 - 64.9375 + 5}{-3.875} = 5.001008,$$

$$\beta_1^{(3)} = \beta_1^{(2)} - \frac{(\beta_1^{(2)})^2 - 6\beta_1^{(2)} + 5}{\beta_1^{(2)} - \beta_2^{(2)}} = \\ = 0.998992 - \frac{0.998992^2 - 60.998992 + 5}{0.998992 - 5.001008} = 0.9999997,$$

$$\beta_2^{(3)} = \beta_2^{(1)} - \frac{(\beta_2^{(2)})^2 - 6\beta_2^{(2)} + 5}{\beta_2^{(2)} - \beta_2^{(1)}} = \\ = 0.001008 - \frac{5.001008^2 - 65.001008 + 5}{5.001008 - 0.998992} = 5.0000003.$$

Результати цих обчислень підтверджують, що послідовності $\left\{\beta_i^{(n)}\right\}$ збігаються швидше за послідовності $\left\{x_i^{(n)}\right\}$, обчислені для прикладу 1.

Приклад 5. Алгоритм (30) для полінома $f(x) = (x-2)(x-4)(x-10)$ з прикладу 2 з початковими наближеннями $\beta_1^{(0)} = \alpha_1 = 1.5$; $\beta_2^{(0)} = \alpha_2 = 5$; $\beta_3^{(0)} = \alpha_3 = 9.5$ дає такі результати

$$\beta_1^{(1)} = \beta_1^{(0)} - \frac{f(\beta_1^{(0)})}{\varphi_2^{(0)}(\beta_0^{(2)})} = \\ = \beta_1^{(0)} - \frac{(\beta_2^{(0)} - \beta_1)(\beta_1^{(0)} - \beta_2)(\beta_3^{(0)} - \beta_3)}{(\beta_2^{(0)} - \beta_1^{(0)})(\beta_2^{(0)} - \beta_3^{(0)})} = \\ = 5 - \frac{(5-2)5-4)(5-10)}{(5-1.5)(5-9.5)} = 4.0476191;$$

$$\beta_3^{(1)} = \beta_3^{(0)} - \frac{f(\beta_3^{(0)})}{\varphi_3^{(0)}(\beta_3^{(0)})} =$$
$$= \beta_3^{(0)} - \frac{(\beta_3^{(0)} - \beta_1)(\beta_3^{(0)} - \beta_2)(\beta_3^{(0)} - \beta_3)}{(\beta_3^{(0)} - \beta_1^{(0)})(\beta_3^{(0)} - \beta_2^{(0)})} =$$
$$= 9.5 - \frac{(9.5-2)(2.5-4)(9.5-10)}{(9.5-1.5)(9.5-5)} = 10.072916;$$

$$\beta_1^{(2)} = \beta_1^{(1)} - \frac{f(\beta_1^{(1)})}{\varphi_2^{(1)}(\beta_1^{(1)})} =$$
$$= \beta_1^{(1)} - \frac{(\beta_1^{(1)} - \beta_1)(\beta_1^{(1)} - \beta_2)(\beta_1^{(1)} - \beta_3)}{(\beta_1^{(1)} - \beta_2^{(1)})(\beta_1^{(1)} - \beta_3^{(1)})} =$$
$$= 1.8794642 - (-0.1168393) = 1.9963035;$$

$$\beta_2^{(2)} = \beta_2^{(1)} - \frac{f(\beta_2^{(1)})}{\varphi_2^{(1)}(\beta_2^{(1)})} =$$
$$= \beta_2^{(1)} - \frac{(\beta_2^{(1)} - \beta_1)(\beta_2^{(1)} - \beta_2)(\beta_2^{(1)} - \beta_3)}{(\beta_2^{(1)} - \beta_1^{(1)})(\beta_2^{(1)} - \beta_3^{(1)})} =$$
$$= 4.0476191 - 0.0444275 = 4.0031916;$$

$$\beta_3^{(2)} = \beta_3^{(1)} - \frac{f(\beta_3^{(1)})}{\varphi_3^{(1)}(\beta_3^{(1)})} =$$
$$= \beta_3^{(1)} - \frac{(\beta_3^{(1)} - \beta_1)(\beta_3^{(1)} - \beta_2)(\beta_3^{(1)} - \beta_3)}{(\beta_3^{(1)} - \beta_1^{(1)})(\beta_3^{(1)} - \beta_2^{(1)})} =$$
$$= 10.072916 - 0.072411 = 10.000505;$$

$$\beta_1^{(2)} = \beta_1^{(2)} - \frac{f(\beta_1^{(2)})}{\varphi_1^{(2)}(\beta_1^{(1)})} =$$
$$= \beta_1^{(2)} - \frac{(\beta_1^{(2)} - \beta_1)(\beta_1^{(2)} - \beta_2)(\beta_1^{(2)} - \beta_3)}{(\beta_1^{(2)} - \beta_2^{(2)})(\beta_1^{(2)} - \beta_3^{(2)})} =$$
$$= 1.9963035 - (-0.0036901) = 1.9999936;$$

$$\beta_2^{(3)} = \beta_2^{(2)} - \frac{f(\beta_2^{(2)})}{\varphi_2^{(2)}(\beta_2^{(2)})} =$$
$$= \beta_2^{(2)} - \frac{(\beta_2^{(2)} - \beta_1)(\beta_2^{(2)} - \beta_2)(\beta_2^{(2)} - \beta_3)}{(\beta_2^{(2)} - \beta_1^{(2)})(\beta_2^{(2)} - \beta_3^{(2)})} =$$
$$= 4.0031916 - 0.0031854 = 4.0000062;$$

$$\beta_3^{(3)} = \beta_3^{(2)} - \frac{f(\beta_3^{(2)})}{\varphi_3^{(2)}(\beta_3^{(2)})} =$$
$$= \beta_3^{(2)} - \frac{(\beta_3^{(2)} - \beta_1)(\beta_3^{(2)} - \beta_2)(\beta_3^{(2)} - \beta_3)}{(\beta_3^{(2)} - \beta_1^{(2)})(\beta_3^{(2)} - \beta_3^{(2)})} =$$
$$= 10.000505 - 0.000505 = 10.000000.$$

9. Очевидно, що вимога щодо одиничної кратності коренів β_i полінома $f(x)$ не обмежує загальності у попередньому викладі. З іншого боку, наведений нижче приклад засвідчує, що досліджені алгоритми можуть виявитися придатними, а результати обчислень з їх використанням можуть давати прийнятні практично числові результати для полінома $f(x)$, який має кратні корені.

Приклад 6. Використаємо алгоритм (13) для уточнення коренів полінома $f(x) = (x-2)^2(x-1)$. Позначимо $\beta_1 = 1; \beta_2 = \beta_3 = 2$. Приймемо $\alpha_1 = 0.7; \alpha_2 = 1.8; \alpha_3 = 2.5$. Очевидно, що умова (4) справджується. Отже, $\varphi(x) = (x-0.7)(x-1.8)(x-2.5), \varphi_1(x) = (x-1.8)(x-2.5); \varphi_2(x) = (x-0.7)(x-2.5); \varphi_3(x) = (x-0.7)(x-1.8)$. Обчислення за формулами (30) дають, зокрема,
$x_1^{(1)} = 0.5660606; x_2^{(1)} = 1.8415584; x_3^{(1)} = 2.202381;$
$x_1^{(2)} = 0.9928110; x_2^{(2)} = 1.8732606; x_3^{(2)} = 2.1203424;$
$x_1^{(3)} = 0.9988054; x_2^{(3)} = 1.8923362; x_3^{(3)} = 2.0846828;$
$x_1^{(4)} = 0.9998009; x_2^{(4)} = 1.9226101; x_3^{(4)} = 2.0649511.$
Для алгоритму (33) можна використати рівносильну форму запису при $(i = \overline{1,m})$

$$\beta_i^{(n+1)} = \beta_i^{(n)} - \sum_{j=1}^{m} \frac{f(\beta_j^{(n)})}{(\beta_i^{(n)} - \beta_j^{(n)}) \prod\limits_{i=1}^{m} (\beta_j^{(n)} - \beta_j^{(n)})}. \quad (41)$$

При початковому наближенні (12) формули (41) можна використати як формули побудови для наближення ("гуртом") до всіх водночас коренів β_1 полінома $f(x)$ за допомогою гіллястих дробів В.Я. Скоробогатька, які збігаються до цих коренів з квадратичною швидкістю збіжності. Відмітимо, що у цьому випадку отримані гіллясті дроби не є, взагалі кажучи, періодичними.

10. Вважатимемо, що коефіцієнти a_i полінома $f(x)$ є елементами банахової алгебри E з одиницею. Задамо, як і раніше, попарно різні комплексні числа α_j $(j = \overline{1,m})$. Означимо множини M_i як сукупності таких елементів $x \in E$, для яких $\alpha_j = (j = \overline{1,m}; j \neq i)$ не є власними числами i, отже, існують обернені елементи $(x - \alpha_i I)^{-1}$. Крім того, припускаємо, що елементи $\alpha_i I$ не є розв'язками рівняння (1), що не порушує загальності викладення. За цих припущень рівняння (1) рівносильне з рівнянням

$$x = x - f(x)\varphi_i^{-1}(x) \quad (x \in M_i) \quad (i = \overline{1,m}), \quad (42)$$

а також з рівнянням

$$x = x_i^{(0)} + \sum_{j=\overline{1,m}; j\neq i} \kappa_{ij}(x - \alpha_i I)^{-1} \quad (i = \overline{1,m}). \quad (43)$$

Використовуємо позначення

$$\varphi_1(x) = \prod_{j=\overline{1,m}; j\neq i} \kappa_{ij}(x - \alpha_i I)^{-1} \quad (i = \overline{1,m}), \quad (44)$$

$$\kappa_{ij} = -\frac{1}{\varphi_i'(\alpha_i)} f(\alpha_i I) \quad (i,j = \overline{1,m}), \quad (45)$$

де $\varphi_i'(a_i)$ – означене за формулою (10) число, а також позначення

$$x_i^{(0)} = -\left(\alpha_{m+1} + I \sum_{j=\overline{1,m}; j\neq i} \alpha_i\right)(i,j = \overline{1,m}). \quad (46)$$

Ітераційний процес для $i = \overline{1,m}$ реалізуємо за формулами

$$x_i^{(n+1)} = x_i^{(0)} + \sum_{j=\overline{1,m}; j\neq i} \kappa_{ij}(x_i^{(n)} - \alpha_i I)^{-1} \quad (47)$$

з початковими наближеннями (46). Їх можна подати за вибору $x_i^{(0)}$ за формулами (46) також у вигляді

$$x_i^{(n+1)} = x_i^{(n)} - f(x_i^{(n)})\varphi_i^{-1}(x_i) \quad (i = \overline{1,m}). \quad (48)$$

Рівносильність ітераційних формул (46) і (47) випливає з рівносильності рівняння (42) при $x_i^{(0)}$, що задовольняють рівності (46), з рівнянням (43) для кожного фіксованого $i = \overline{1,m}$. У наступній теоремі вважатимемо, що індекс i зафіксований і для цього індексу розглядаємо ітераційний процес (46), (47). Через $\|x\|$ позначимо норму елемента $x \in E$.

Теорема 3. *Нехай* $\tilde{M}_i = \{x : ||x - \alpha_i|| \le q < 1; x \in E\}$ – *куля у банаховій алгебрі* E *і нехай: 1) якщо* $x \in \tilde{M}_i$, *то* $\|x - \alpha_i I\| \ge Q > 1$, *при* $j = \overline{1,m}$; $j \ne i$; *2) при* $x, y \in \tilde{M}_i$ *матимемо*

$$\left\| \sum_{j=\overline{1,m}; j\ne i} k_{ij}(x-\alpha_i I)^{-1}(y-\alpha_i I)^{-1} \right\| \le h_i < 1.$$

Тоді послідовність $x_i^{(n)}$ *при зафіксованому* i *збігається за нормою в* E *до єдиного в* $\tilde{M}_i$ *розв'язку* β_i *рівняння (1) не повільніше від геометричної прогресії зі знаменником* h_i.

$\square$ *Доведення.* Як і при доведенні теореми 1, досить застосувати принцип стиску, отримавши перед цим аналог рівності (11) в термінах банахової алгебри E. Якщо поліном $f(x)$ має вигляд (3), то аналогічні до наведеного результати отримуються очевидним способом. $\blacksquare$

Отримані результати стосуються ситуації, коли алгоритм (46), (47) розглядається індивідуально при фіксованому значенні i. Цей процес можна розглядати і як "гуртовий" алгоритм, коли виконуються ітерації (47) водночас для всіх $i = \overline{1,m}$. У цьому випадку так само, як і для ітераційного процесу (11) з умовою (4) можна використовувати замість формул (46) формули вигляду (27) як засіб для коригування ітераційного процесу. Алгоритм (46), (47) за структурою і за способом обґрунтування близький до відповідного алгоритму із [6].

Приклад 7. Розглянемо рівняння $x^2 - Ax = \theta$, де $A = \begin{pmatrix} 2,1000 & 0,3000 \\ 0,1000 & 2,2000 \end{pmatrix}$, $\theta = \begin{pmatrix} 0 & 0 \\ 0 & 0 \end{pmatrix}$. Нехай $\alpha = 2$. Отже,

$$x_0 = -A - \alpha I = \begin{pmatrix} 2,1000 & 0,3000 \\ 0,1000 & 2,2000 \end{pmatrix} - -2\begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix} = \begin{pmatrix} 0,1000 & 0,3000 \\ 0,1000 & 0,2000 \end{pmatrix},$$

$$f(\alpha I) = \begin{pmatrix} 2 & 0 \\ 0 & 2 \end{pmatrix} - -\begin{pmatrix} 2,1000 & 0,3000 \\ 0,1000 & 2,2000 \end{pmatrix}\begin{pmatrix} 2 & 0 \\ 0 & 2 \end{pmatrix} = =\begin{pmatrix} 0,2000 & 0,6000 \\ 0,2000 & 0,4000 \end{pmatrix}.$$

Одержуємо такі послідовні наближення:

$$x^{(1)} = \begin{pmatrix} -0,0624 & -0,1186 \\ -0,0023 & -0,0998 \end{pmatrix},$$

$$x^{(2)} = \begin{pmatrix} 0,0194 & 0,0168 \\ 0,0139 & 0,0114 \end{pmatrix},$$

$$x^{(3)} = \begin{pmatrix} -0,0009 & -0,1186 \\ -0,0023 & -0,0020 \end{pmatrix},$$

$$x^{(4)} = \begin{pmatrix} 0,0036 & -0,0001 \\ 0,0000 & 0,0000 \end{pmatrix},$$

що можна вважати за прийнятну ілюстрацію ефективності алгоритму (46), (47).

Наведені у цій замітці дослідження є продовженням досліджень із [6; 12; 13].

Література

[1] Hildebrand F.B. Introduction to Numerical Analysis. – Mc Crow-Hill Book Company, Inc, New York, 1956.

[2] Трауб Дж. Итерационные методы решения уравнений. – М.: Мир, 1985. – 263 с.

[3] Боднарчук П.І., Скоробогатько В.Я. Гіллясті ланцюгові дроби та їх застосування. – К.: Наук.думка, 1974. – 272 с.

[4] Скоробогатько В.Я. Теория ветвящихся цепных дробей и ее применение в вычислительной математике. – М.: Наука, 1983. – 312 с.

[5] Боднар Д.И. Ветвящиеся цепные дроби. – К.: Наук. думка, 1986. – 174 с.

[6] Шувар Б.А., Шуляр М.А. Про нулі полінома з коефіцієнтами із алгебри Банаха // Вісник Львівського політехн. інституту. Математ. Механіка. – Львів, 1977. – Т.119.

[7] Рисс Ф., Секефальви – Надь Б. Лекции по функциональному анализу. – М.: ИЛ, 1954.

[8] Канторович Л.В., Акилов Г.П. Функциональный анализ. – М.: Наука, 1977. – 741 с.

[9] Рудин У. Функциональный анализ. – М.: Мир, 1975. – 443 с.

[10] Сявавко М.С. Гіллясті ланцюгові дроби. – К.: Наук.думка, 1994. – 205 с.

[11] Шувар Б.А., Копач М.І., Ментинський С.М., Обшта А.Ф. Двосторонні наближені методи. – Івано-Франківськ: Вид-во Прикарпатського нац. ун-ту ім. В. Стефаника, 2007. – 515 с.

[12] Шувар Б.А., Обшта А.Ф. Наближена факторизація поліномів у банахових алгебрах // Дванадцята міжнародна наукова конференція ім. акад. М. Кравчука. Матеріали конференції. – К., 2008. – с. 870.

[13] Шувар Б.А., Обшта А.Ф. Наближена факторизація поліномів у банахових алгебрах і числових полях // 9-та наукова конференція ін-ту ІМФН Нац. ун-ту "Львівська політехніка". – Львів, 2008.

ИТЕРАЦИОННЫЕ АЛГОРИТМЫ И ВЕТВЯЩИЕСЯ ДРОБИ ДЛЯ ФАКТОРИЗАЦИИ МНОГОЧЛЕНОВ В ЧИСЛОВЫХ ПОЛЯХ И БАНАХОВЫХ АЛГЕБРАХ

А.Ф. Обшта, Б.А. Шувар

Национальный университет "Львивська политехника", ул. С. Бандеры, 12, Львов, 79013, Украина

Предложены итерационные алгоритмы аппроксимации корней многочленов в банаховых алгебрах, обладающие линейной сходимостью, и построены соответствующие аналоги ветвящихся дробей. На их основании для многочленов с числовыми коэффициентами построены итерационные алгоритмы с квадратичной сходимостью, не ассоциирующиеся с методами ньютоновского типа.

Ключевые слова: банаховы алгебры, итерационные алгоритмы, ветвящиеся дроби, линейная сходимость, квадратичная сходимость.

2000 MSC: 12D05/32A65

УДК: 517.948/517.946

THE ITERATIVE ALGORITHMS AND BRANCHING FRACTIONS FOR FACTORING POLYNOMIALS IN NUMERICAL FIELDS AND BANACH ALGEBRAS

A.F. Obshta, B.A. Shuvar

National University "Lvivska Politechnika" 12 S. Bandera Str., 79013, Lviv, Ukraine

The iterative algorithms for approximating roots of polynomials in Banach algebras, which have linear convergence are proposed and corresponding analogues branched fractions are constructed. On their basis for polynomials with numerical coefficients proposed iterative algorithm with quadratic convergence, which is not associated with Newton methods.

Key words: the Banach algebras, iterative algorithms, branching fractions, linear convergence, quadratic convergence.

2000 MSC: 12D05/32A65

УДК: 517.948/517.946