

4. Градієнтні методи багатомірної оптимізації

4.1. Методи першого порядку

4.1.1. Градієнтний метод

Як відомо (див. розділ 1.5), градієнт (антиградієнт) скалярної функції $f_0(\vec{u})$ в деякій точці \vec{u}_k спрямований у бік найшвидшого зростання (найшвидшого зменшення) функції.

Вибираючи як напрямок пошуку екстремуму градієнт або антиградієнт (залежно від постановки задачі), алгоритм градієнтного методу можна записати у вигляді:

$$\vec{u}_{k+1} = \vec{u}_k - h_k \cdot \nabla f_0(\vec{u}_k), \quad (4.1)$$

де h_k - кроковий множник (параметр кроку).

Останній запис означає, що із точки $\vec{u}_k = \{u_1, u_2, \dots, u_n\}|_{\vec{u}=\vec{u}_k}$ у напрямку екстремуму робиться пошуковий крок величиною: $\Delta \vec{u}_k = \left\{ h \frac{\partial f_0(\vec{u})}{\partial u_1}, h \frac{\partial f_0(\vec{u})}{\partial u_2}, \dots, h \frac{\partial f_0(\vec{u})}{\partial u_n} \right\}|_{\vec{u}=\vec{u}_k}$ по кожній із незалежних змінних $u_i, i=1, 2, \dots, n$, тобто одночасно змінюються значення всіх незалежних змінних (рис. 4.1). У знайденій точці \vec{u}_{k+1} обчислюється значення цільової функції $f_0(\vec{u}_{k+1})$ і виконується перевірка достатніх умов оптимальності.

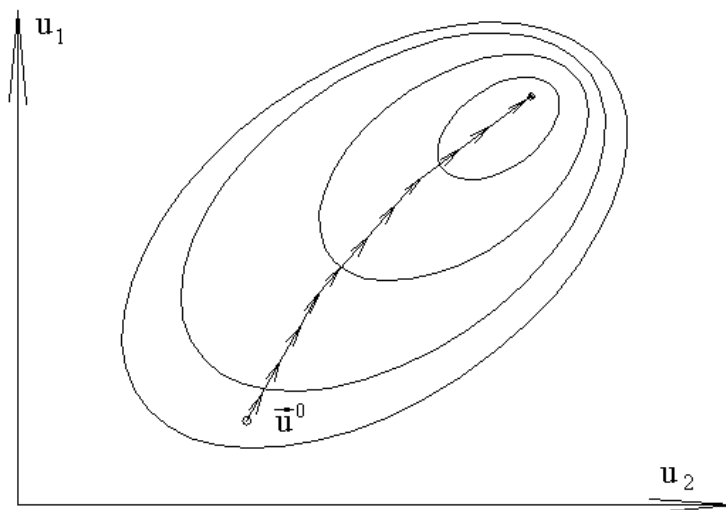


Рис. 4.1. Пошук екстремуму градієнтним методом

На практиці, як правило, пошук припиняють при виконанні умови:

$$\|\nabla f_0(\vec{u})\|_{\vec{u}=\vec{u}_k} = \sqrt{\sum_{i=1}^n \left[\left(\frac{\partial f_0(\vec{u})}{\partial u_i} \right) \Big|_{\vec{u}=\vec{u}_k} \right]^2} \leq \varepsilon,$$

де символ $\|\cdot\|$ позначає норму градієнта цільової функції $f_0(\vec{u})$; ε – наперед задане, достатньо мале число.

Як видно з (4.1) величина пошукового кроку залежить від градієнта цільової функції $\nabla f_0(\vec{u}_k)$. Це обставина істотно впливає на процес пошуку, тому що поблизу екстремуму $\nabla f_0(\vec{u}_k) \rightarrow 0$, і, отже, зменшується величина пробного кроку, що зменшує швидкість збіжності (швидкість досягнення екстремуму) алгоритму (4.1) до екстремуму.

Прийнятна величина пробного кроку встановлюється за допомогою крокового множника h_k . Необхідно відзначити, що занадто мале значення h_k знижує швидкість збіжності, а занадто велике значення h_k може призвести до того, що екстремум не буде

досягнутий взагалі. Останнє пояснюється тим, що при великому h_k відбувається перехід точки \vec{u}_{k+1} через точку екстремуму, і надалі зміну напрямку пошуку практично на протилежний, тобто повернення в точку \vec{u}_k . Таким чином, у районі екстремуму виникають пошукові коливання, що не припиняються.

Для вибору величини крокового множника може бути використаний наступний алгоритм:

$$h_{k+1} = \begin{cases} 2h_k, & \text{якщо } \gamma_k < \gamma_{\min}, \\ h_k, & \text{якщо } \gamma_{\min} \leq \gamma_k \leq \gamma_{\max}, \\ \left(\frac{1}{3} \div \frac{2}{5}\right)h_k, & \text{якщо } \gamma_k > \gamma_{\max}, \end{cases} \quad (4.2)$$

де γ_k - кут між сусідніми градієнтами (рис. 4.2), а γ_{\min} й γ_{\max} – допустимі граничні значення.

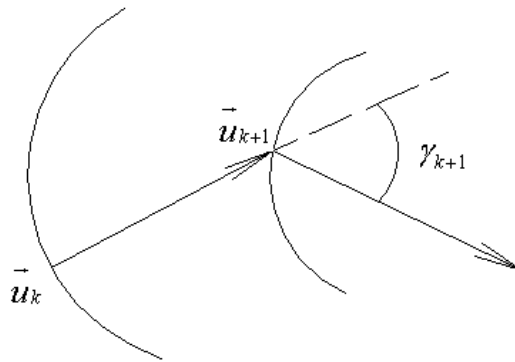


Рис. 4.2. До алгоритму вибору кроку градієнтного метода

Кут γ_{k+1} може бути обчислений в такий спосіб:

$$\cos \gamma_{k+1} = \frac{\sum_{i=1}^m \left(\frac{\partial f_0(\vec{u})}{\partial u_i} \Big|_{\vec{u}=\vec{u}_k} \cdot \frac{\partial f_0(\vec{u})}{\partial u_i} \Big|_{\vec{u}=\vec{u}_{k+1}} \right)}{\sqrt{\sum_{i=1}^m \left(\frac{\partial f_0(\vec{u})}{\partial u_i} \Big|_{\vec{u}=\vec{u}_{k+1}} \right)^2} \cdot \sqrt{\sum_{i=1}^m \left(\frac{\partial f_0(\vec{u})}{\partial u_i} \Big|_{\vec{u}=\vec{u}_k} \right)^2}} \quad (4.3)$$

З рис. 4.2 випливає, що якщо γ_{k+1} по величині малий, тобто напрямки $\nabla f_0(\vec{u}_k)$ і $\nabla f_0(\vec{u}_{k+1})$ практично збігаються, то можна збільшити пошуковий крок, припускаючи, що екстремум перебуває далеко. Якщо ж γ_{k+1} – великий, тобто градієнти $\nabla f_0(\vec{u}_k)$ і $\nabla f_0(\vec{u}_{k+1})$ спрямовані практично назустріч один одному, то необхідно зменшити пошуковий крок, тому що очевидно, що точка \vec{u}_{k+1} перейшла за точку екстремуму. Ці прості міркування і покладені в основу алгоритму (4.1).

4.1.2. Метод найшвидшого спуску.

Градієнтний метод (4.1), на кожній ітерації якого величина крокового множника h_k вибирається з умови мінімуму функції $f_0(\vec{u})$ в напрямку руху, тобто $f_0(\vec{u}_k - h_k \cdot \nabla f_0(\vec{u}_k)) = \min_{h \geq 0} f_0(\vec{u}_k - h \cdot \nabla f_0(\vec{u}_k))$, називається *методом найшвидшого спуску* (методом Коші).

Відповідно до методу найшвидшого спуску, на кожній ітерації необхідно за допомогою описаних вище методів вирішувати задачу одномірної мінімізації функції $f_0(\vec{u})$ в напрямку антиградієнту. Пошук здійснюється доти, поки не перестане

зменшуватися значення $f_0(\vec{u})$ (рис. 4.3). Після цього визначається значення $\nabla f_0(\vec{u}_{k+1})$ і процедура пошуку триває. Якщо $\|\nabla f_0(\vec{u}_{k+1})\| \leq \varepsilon$, то пошук припиняється.

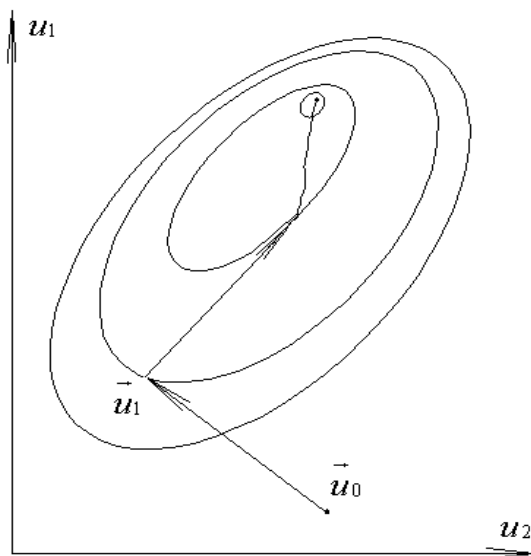


Рис. 4.3. Траекторія руху за методом Коші

Особливістю методу найшвидшого спуску є те, що для квадратичних цільових функцій, напрямки спуску на двох послідовних ітераціях взаємно ортогональні (перпендикулярні).

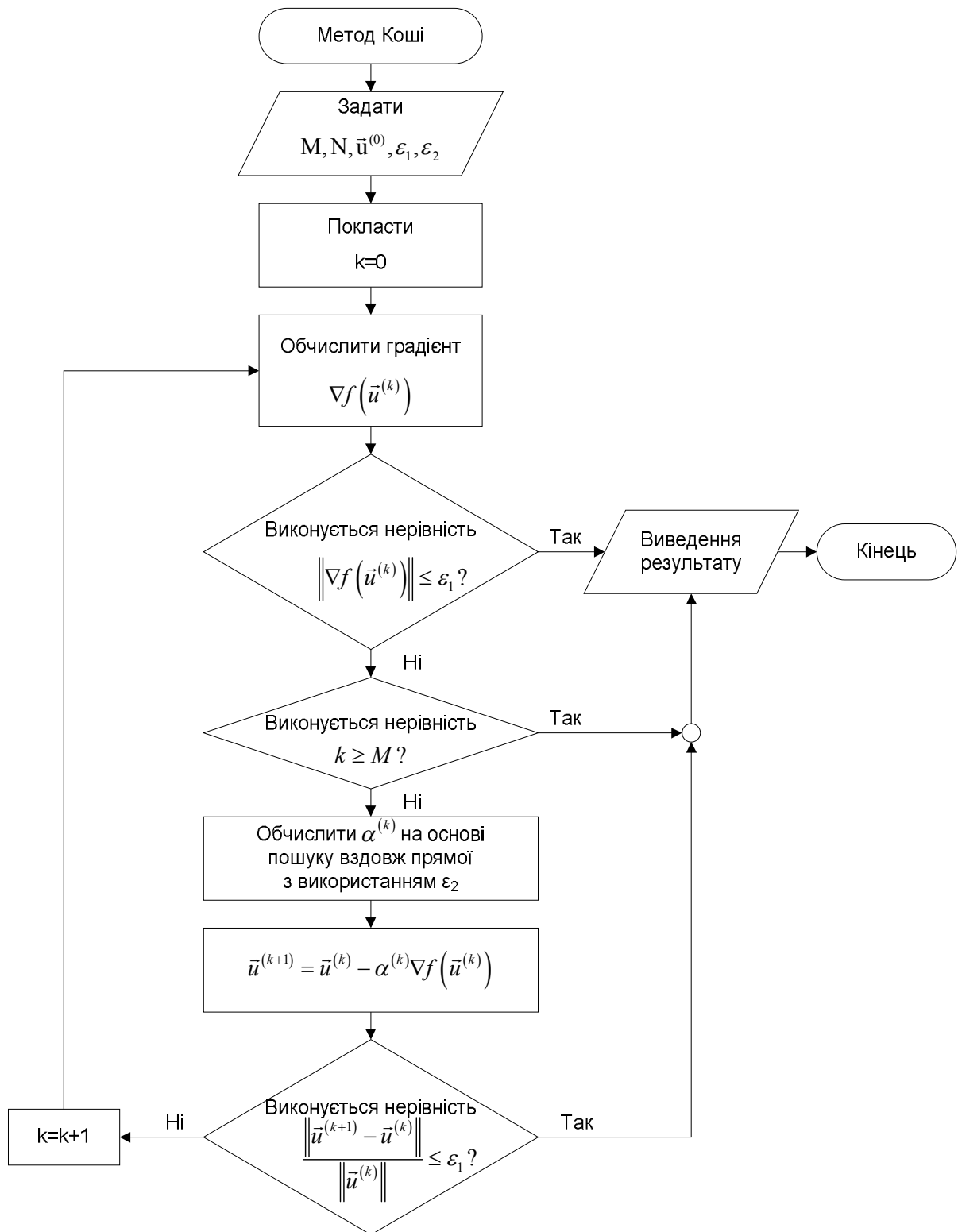


Рис. 4.4. Структурна схема методу Коші

Приклад 4.1. Метод найшвидшого спуску

Розглянемо функцію

$$f(x) = 8x_1^2 + 4x_1x_2 + 5x_2^2$$

і скористаємось методом Коші для вирішення задачі її мінімізації.

Перш за все, обчислимо компоненти градієнта

$$\frac{\partial f}{\partial x_1} = 16x_1 + 4x_2, \quad \frac{\partial f}{\partial x_2} = 10x_2 + 4x_1.$$

Для того, щоб застосувати метод найшвидшого спуску, задамо початкове наближення

$$x^{(0)} = [10, 10]^T$$

і побудуємо нове наближення

$$x^{(1)} = x^{(0)} - \alpha^{(0)} \nabla f(x^{(0)}).$$

Оберемо $\alpha^{(0)}$ таким чином, щоб $f(x^{(1)}) \rightarrow \min$; $\alpha^{(0)} = 0,056$. Отже, $x^{(1)} = [-1,20, 2,16]^T$. Далі знайдемо точку

$$x^{(2)} = x^{(1)} - \alpha^{(1)} \nabla f(x^{(1)}),$$

обчисливши градієнт в точці $x^{(1)}$ і провівши пошук вздовж прямої.

Таблиця 4.1

Результати обчислення за методом Коші

k	$x_1^{(k)}$	$x_2^{(k)}$	$f(x^{(k)})$
1	-1,2403	2,1181	24,2300
2	0,1441	0,1447	0,3540
3	-0,0181	0,0309	0,0052
4	0,0021	0,0021	0,0000

В табл. 4.1 представлено дані, отримані при проведенні ітерацій на основі одномірного пошуку за методом квадратичної інтерполяції. Послідовність отриманих точок зображено на рис. 4.5.

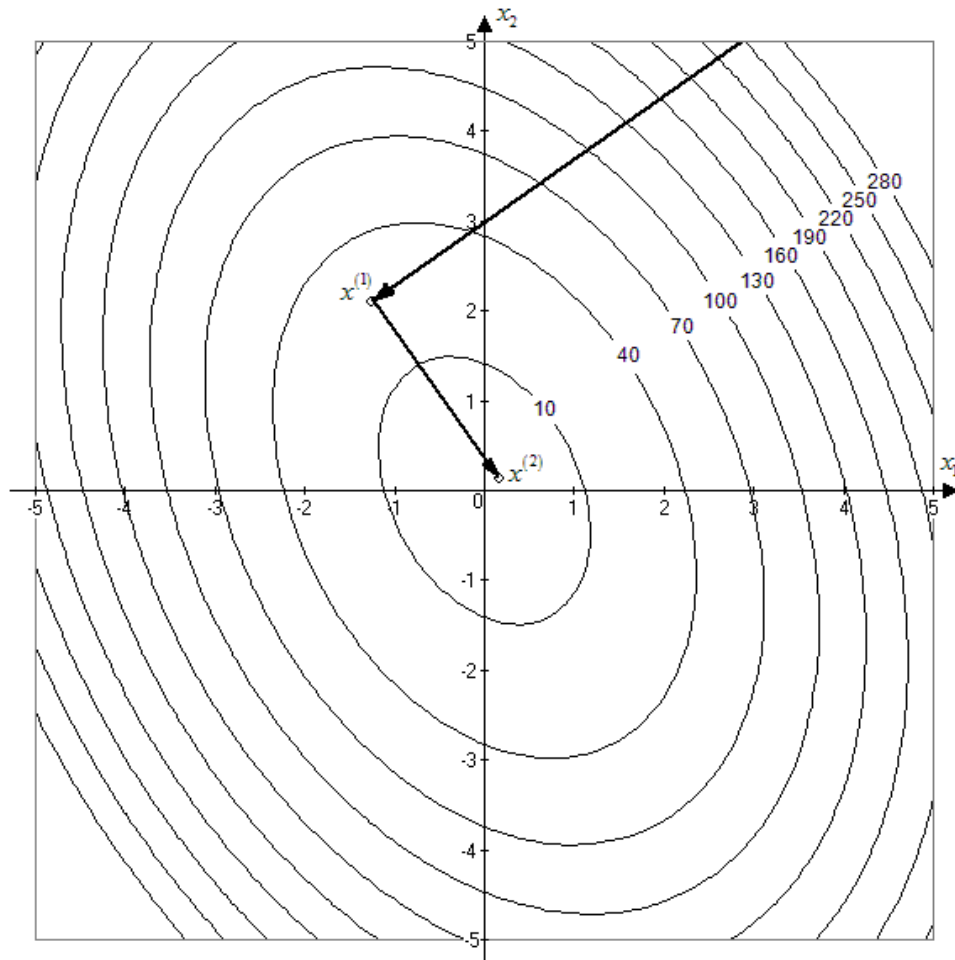


Рис. 4.5. Ітерації за методом Коші з використанням квадратичної інтерполяції

4.1.3. Метод крутого спуску

В методі крутого спуску в напрямку градієнта виконується не один крок, а декілька до моменту досягнення локального екстремуму.

Алгоритм метода має вигляд

$$\vec{u}_{k+1} = \vec{u}_k - h_k \cdot \frac{df(x^{(p)})}{dx_j} \quad (4.4)$$

де p – точка, в якій обчислюється градієнт.

Траєкторію руху за цим алгоритмом зображено на рис. 4.6.

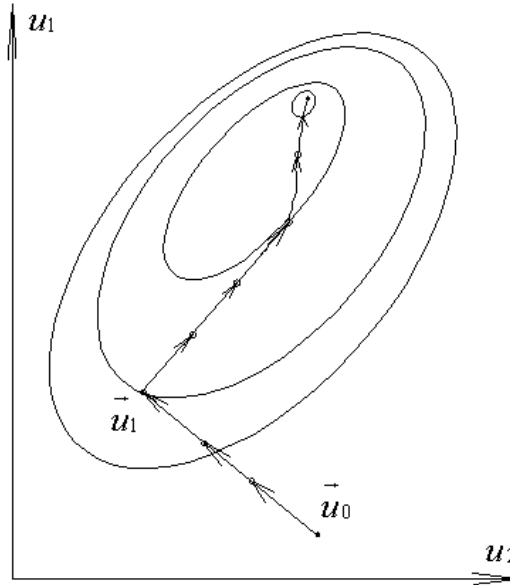


Рис. 4.6. Траєкторія руху за методом крутого спуску

4.1.4. Метод спряжених градієнтів (Флетчера-Рівса)

В розглянутих методах на кожному кроці ніяк не використовувалась інформація, отримана на попередніх кроках. Очевидно, використовуючи цю інформацію, можна розраховувати на прискорення його збіжності. Методи пошуку, в яких нове наближення формується на основі декількох попередніх, називаються багатокроковими. Одним з багатокрокових методів є двокроковий метод спряжених градієнтів. Ідея методу заключається в побудові наступного наближення за схемою:

$$\vec{u}_{k+1} = \vec{u}_k - h_k \nabla f(\vec{u}_k) + \beta_k (\vec{u}_k - \vec{u}_{k-1}), \quad (4.5)$$

де параметри h_k і β_k вибираються на кожному кроці оптимальним чином, тобто з умови

$$(h_k, \beta_k) = \arg \min_{h, \beta} f(\vec{u}_k - h \nabla f(\vec{u}_k) + \beta (\vec{u}_k - \vec{u}_{k-1})). \quad (4.6)$$

В загальному випадку знайти рішення даної задачі в явному вигляді не вдається, але для випадку квадратичної функції методу можна придати наступну форму:

$$\vec{u}_{k+1} = \vec{u}_k + h_k \vec{s}_k, \quad \vec{s}_k = -\nabla f(\vec{u}_k) + \beta_k \vec{s}_{k-1}, \quad (4.7)$$

$$\beta_k = \frac{|\nabla f(\vec{u}_k)|^2}{|\nabla f(\vec{u}_{k-1})|^2}, \quad \beta_0 = 0, \quad h_k = \arg \min_{h \geq 0} f(\vec{u}_k + h \vec{s}_k).$$

Характерною особливістю методу є той факт, що він приводить до знаходження точки мінімуму квадратичної цільової функції за число ітерацій, що не перевищує розмірність задачі.

Не дивлячись на те, що останній алгоритм отримано для квадратичної цільової функції, метод може успішно застосовуватись і для неквадратичних функцій. Але тоді метод не буде кінцевим, тому при цьому видається доцільним через кожні n кроків

проводити оновлення методу, тобто приймати $\beta = 0$ при $k = 0, n, 2n, \dots$. Метод Флетчера-Рівса є за формою методом першого порядку і тому простим в реалізації, вигідно відрізняється від звичайних градієнтних методів тим, що наділений по суті всіма перевагами методів другого порядку (в тому числі квадратичною збіжністю).

Приклад 4.2. Метод Флетчера-Рівса

Знайти точку мінімуму функції

$$f(x) = 4x_1^2 + 3x_2^2 - 4x_1x_2 + x_1,$$

якщо $\bar{x}^{(1)} = [0, 0]^T$.

Крок 1. Визначимо градієнт цільової функції $\nabla f(x) = [8x_1 - 4x_2 + 1, 6x_2 - 4x_1]^T$, тоді напрямком пошуку $\bar{s}^{(1)} = -\nabla f(\bar{x}^{(1)}) = -[1, 0]^T$.

Крок 2. Пошук вздовж прямої:

$$\bar{x}^{(2)} = \bar{x}^{(1)} - \alpha^{(1)} \nabla f(\bar{x}^{(1)}) \rightarrow \alpha_1 = \frac{1}{8},$$

$$\bar{x}^{(2)} = [0, 0]^T - \frac{1}{8} [1, 0]^T = \left[-\frac{1}{8}, 0\right]^T.$$

Крок 3. $k = 2$

$$\bar{s}^{(2)} = -\left[0, \frac{1}{2}\right]^T - \left[\frac{1}{4}, 1\right] [1, 0]^T = \left[-\frac{1}{4}, \frac{1}{2}\right]^T.$$

Крок 4. Пошук вздовж прямої:

$$\bar{x}^{(3)} = \bar{x}^{(2)} + \alpha^{(2)} \bar{s}^{(2)} \rightarrow \alpha^{(2)} = \frac{1}{4},$$

$$x^{(3)} = \left[-\frac{1}{8}, 0\right]^T - \frac{1}{4} \left[\frac{1}{4}, \frac{1}{2}\right]^T = \left[-\frac{3}{16}, -\frac{1}{8}\right]^T,$$

$$\nabla f(\bar{x}^{(3)}) = [0, 0]^T.$$

Таким чином, отримане рішення збігається з точним рішенням. Це пояснюється тим, що цільова функція квадратична, а похибки округлення відсутні.

4.2. Методи другого порядку

4.2.1. Метод Ньютона

Градієнтний метод заснований на припущенні про можливість локальної лінійної апроксимації цільової функції $f_0(\vec{u})$.

Якщо цільова функція двічі диференційована, то можна скористатися квадратичною апроксимацією цільової функції.

Розкладемо цільову функцію в ряд Тейлора:

$$f_0(\vec{u}_{k+1}) = f_0(\vec{u}_k) + \nabla f_0(\vec{u}_k)^T (\vec{u}_{k+1} - \vec{u}_k) + \frac{1}{2} (\vec{u}_{k+1} - \vec{u}_k)^T \cdot \nabla^2 f_0(\vec{u}_k) (\vec{u}_{k+1} - \vec{u}_k) \dots \quad (4.8)$$

Якщо знехтувати у цьому ряді всіма членами, починаючи із другого порядку, то необхідна умова екстремуму в точці \vec{u}_{k+1} буде мати вигляд:

$$\nabla f_0(\vec{u}_k) + \nabla^2 f_0(\vec{u}_k) (\vec{u}_{k+1} - \vec{u}_k) = 0 \quad (4.9)$$

Звідки:

$$\vec{u}_{k+1} = \vec{u}_k - \left[\nabla^2 f_0(\vec{u}_k) \right]^{-1} \nabla f_0(\vec{u}_k) \quad (4.10)$$

Останній вираз є алгоритмом методу Ньютона. У тому випадку, якщо \vec{u}_{k+1} буде лише приблизним розв'язком задачі оптимізації, необхідно прийняти \vec{u}_{k+1} в якості нової початкової точки і знову скористатися алгоритмом (4.10). Таким чином, метод Ньютона можна розглядати як послідовний пошук точок мінімуму квадратичних апроксимацій цільової функції $f_0(\vec{u})$.

Метод Ньютона має швидку збіжність поблизу екстремуму, де помилка, викликана нехтуванням членами вищого порядку в розкладанні $f_0(\vec{u})$ у ряд Тейлора, незначна. Для квадратичних цільових функцій метод Ньютона дозволяє отримати значення екстремуму за один крок.

Слід відмітити, що метод Ньютона може бути застосований для оптимізації функції однієї змінної $f(u)$, що двічі диференційована. Тоді алгоритм (4.10) набуває наступного вигляду:

$$u_{k+1} = u_k - \left(\frac{d^2 f}{du^2} \right)^{-1} \left(\frac{df}{du} \right). \quad (4.11)$$

Приклад 4.3. Метод Ньютона

Мінімізувати за допомогою методу Ньютона функцію

$$f(x) = 8x_1^2 + 4x_1x_2 + 5x_2^2,$$

$$\vec{x}^{(0)} = [10, 10]^T$$

Визначимо градієнт і матрицю Гессе для цієї функції:

$$\nabla f(x) = [16x_1 + 4x_2, 10x_2 + 4x_1]^T,$$

$$\nabla^2 f(x) = \begin{bmatrix} 16 & 4 \\ 4 & 10 \end{bmatrix}.$$

З формули (4.10) отримуємо:

$$\begin{aligned} \vec{x}^{(1)} &= [10, 10]^T - \frac{1}{144} \begin{bmatrix} 10 & -4 \\ -4 & 16 \end{bmatrix} [200, 140]^T = \\ &= [10, 10]^T - \frac{1}{144} [1440, 1440]^T, \end{aligned}$$

тобто

$$\vec{x}^{(1)} = [0, 0]^T,$$

що збігається з точним рішенням.

4.2.2. Модифікований метод Ньютона

Досвід показує, що при дослідженні неквадратичних функцій метод Ньютона не відзначається високою надійністю. Насправді, коли початкова точка пошуку знаходиться на значній відстані від точки екстремуму, крок за методом Ньютона виявляється занадто великим, що може призвести до відсутності сходження. Метод можна доволі просто модифікувати таким чином, щоб забезпечити зменшення цільової функції від ітерації до ітерації та здійснювати пошук вздовж прямої, як в методі найшвидшого спуску. Алгоритм пошуку набуває наступного вигляду:

$$\vec{u}_{k+1} = \vec{u}_k - \alpha_k \left[\nabla^2 f_0(\vec{u}_k) \right]^{-1} \nabla f_0(\vec{u}_k). \quad (4.12)$$

Вибір α_k здійснюється таким чином, щоб

$$f_0(\vec{u}_{k+1}) \rightarrow \min;$$

це гарантує виконання нерівності

$$f_0(\vec{u}_{k+1}) \leq f_0(\vec{u}_k).$$

Такий метод носить назву модифікованого методу Ньютона і у випадках, коли обчислення точних значень перших і других похідних не викликає суттєвих труднощів, видається надійним і ефективним. Однак при використанні цього методу на кожному кроці виникає необхідність побудови і рішення лінійного рівняння, що містить елементи матриці Гессе $\nabla^2 f_0(\vec{u})$.

4.2.3. Метод Девідона-Флетчера-Пауела (ДФП).

Одним із недоліків методу Ньютона є необхідність обчислення зворотної матриці других похідних цільової функції по всіх незалежних змінних.

Існують методи, що дозволяють не обчислювати безпосередньо матрицю Гессе або зворотну до неї $[\nabla^2 f_0(\bar{u}_k)]^{-1}$, а апроксимувати її, використовуючи для цього знайдені градієнти в попередніх точках. У результаті такої апроксимації на деякій k -тій ітерації буде отримана матриця H_k , що відповідає $[\nabla^2 f_0(\bar{u}_k)]^{-1}$. Завдяки цьому, метод ДФП можна віднести до методів другого порядку, хоча звичайно методи, що використовують для апроксимації зворотної матриці Гессе градієнти в попередніх точках, виділяють у самостійний клас так званих квазіньютонівських методів.

Метод ДФП є досить ефективним, тому що поєднує достоїнства методу найшвидшого спуску (швидка зміна цільової функції на перших ітераціях) і методу Ньютона (добра збіжність поблизу екстремуму).

Алгоритм методу ДФП має вигляд:

$$\bar{u}_{k+1} = \bar{u}_k - \gamma_k H_k \nabla f_0(\bar{u}_k), \quad (4.13)$$

$$H_{k+1} = H_k - \frac{H_k \bar{y}_k \bar{y}_k^T H_k}{\bar{y}_k^T H_k \bar{y}_k} + \frac{\bar{\delta}_k \bar{\delta}_k^T}{\bar{\delta}_k^T \bar{y}_k}, \quad H_1 = I, \quad (4.14)$$

де $\bar{y}_k = \nabla f_0(\bar{u}_{k+1}) - \nabla f_0(\bar{u}_k)$, $\bar{\delta}_k = \bar{u}_{k+1} - \bar{u}_k$, I – одинична матриця, γ_k – кроковий множник, який визначається з умови мінімізації функції в напрямку $H_k \nabla f_0(\bar{u}_k)$.

Видно, що при $k=1$ метод ДФП повністю збігається з методом найшвидшого спуску.

Надалі, за рахунок введення виправлень до H_k , метод ДФП буде наближатись до алгоритму (4.10).

Приклад 4.4. Метод ДФП

За допомогою метода ДФП знайти точку мінімуму функції

$$f(x) = 4x_1^2 + 3x_2^2 - 4x_1x_2 + x_1,$$

якщо $\bar{x}^{(1)} = [0, 0]^T$.

Крок 1. Нехай напрямком пошуку $\bar{s}^{(1)} = -\nabla f(x^{(1)}) = -[1, 0]^T$.

Крок 2. Пошук вздовж прямої:

$$\bar{x}^{(2)} = \bar{x}^{(1)} - \alpha^{(1)} \nabla f(\bar{x}^{(1)}) \rightarrow \alpha_1 = 1/8,$$

$$\bar{x}^{(2)} = [0, 0]^T - 1/8 [1, 0]^T = [-1/8, 0]^T.$$

Крок 3. $k=1$, $H^{(2)} = H^{(1)} - \frac{H^{(1)} \bar{y}^{(1)} (\bar{y}^{(1)})^T H^{(1)}}{(\bar{y}^{(1)})^T H^{(1)} \bar{y}^{(1)}} + \frac{\bar{\delta}_1 \bar{\delta}_1^T}{\bar{\delta}_1^T \bar{y}_1}$,

$$H^{(1)} = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} = I,$$

$$\bar{\delta}^{(1)} = [-1/8, 0]^T - [0, 0]^T = [-1/8, 0]^T,$$

$$\bar{y}^{(1)} = [0, 1/2]^T - [1, 0]^T = [-1, 1/2]^T,$$

$$\begin{aligned}
& -\frac{H^{(1)}\bar{y}^{(1)}(\bar{y}^{(1)})^T H^{(1)}}{(\bar{y}^{(1)})^T H^{(1)}\bar{y}^{(1)}} + \frac{\bar{\delta}_1 \bar{\delta}_1^T}{\bar{\delta}_1^T \bar{y}_1} = \frac{\begin{bmatrix} -1/8, 0 \end{bmatrix}^T}{\begin{bmatrix} -1/8, 0 \end{bmatrix}} - \frac{\begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} -1, 1/2 \end{bmatrix}^T \begin{bmatrix} -1, 1/2 \end{bmatrix} \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}}{\begin{bmatrix} -1, 1/2 \end{bmatrix} \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} -1, 1/2 \end{bmatrix}^T} = \\
& = \begin{bmatrix} 1/8 & 0 \\ 0 & 0 \end{bmatrix} - \begin{bmatrix} 4/5 & -2/5 \\ -2/5 & 1/5 \end{bmatrix}, \\
& H^{(2)} = \begin{bmatrix} 0,325 & 2/5 \\ 2/5 & 4/5 \end{bmatrix},
\end{aligned}$$

Новий напрямок пошуку $\bar{s}^{(2)} = -H^{(2)}\nabla f(x^{(2)}) = -\begin{bmatrix} 1/5, 2/5 \end{bmatrix}^T$.

Крок 4. Пошук вздовж прямої:

$$\begin{aligned}
\bar{x}^{(3)} &= \bar{x}^{(2)} - \alpha^{(2)}\bar{s}^{(2)} \rightarrow \alpha^{(2)} = 5/16, \\
\bar{x}^{(2)} &= \begin{bmatrix} -1/8, 0 \end{bmatrix}^T - 5/16 \begin{bmatrix} 1/5, 2/5 \end{bmatrix}^T = \begin{bmatrix} -3/16, -1/8 \end{bmatrix}^T, \\
\nabla f(x^{(3)}) &= [0, 0]^T.
\end{aligned}$$

Отже, пошук завершено, причому знайдена точка збігається з точним рішенням.

4.3. Застосування градієнтних методів в задачах адаптивного управління

Приклад 4.5. Задача адаптивної ідентифікації статичного об'єкта.

Змістовна постановка

Знайти рівняння, що описує залежність вихідної величини статичного об'єкта управління від вхідної, з використанням метода моделі, що самонастроюється (СНМ).

Математична постановка

Одномірний об'єкт описується рівнянням наступного вигляду:

$$y = a_1 u + a_2 u^2, \quad (4.15)$$

де y, u – вимірювані вихід і вхід об'єкта, a_1, a_2 – коефіцієнти, що треба визначити в результаті рішення задачі ідентифікації.

Модель об'єкта має таку ж структуру:

$$y_m = a_{1,m} u + a_{2,m} u^2 \quad (4.16)$$

де y_m – вихідна величина моделі, $a_{1,m}, a_{2,m}$ – невідомі параметри моделі.

За критерій близькості виходу об'єкта і моделі обираємо квадрат миттєвої помилки між виходом об'єкта і моделі:

$$I = \varepsilon^2 = (y_m - y)^2 \quad (4.17)$$

Математичний опис задачі, що полягає в обчисленні коефіцієнтів $a_{1,m}, a_{2,m}$ на кожному кроці ітераційної процедури оптимізації критерію оптимальності (4.17), має вигляд:

$$I = \varepsilon^2 = (y_m - y)^2 \rightarrow \min_{a_{1,m}, a_{2,m}}, \quad (4.18)$$

$$y = a_1 u + a_2 u^2,$$

$$y_m = a_{1,m} u + a_{2,m} u^2.$$

Початкові значення коефіцієнтів моделі $a_{1,m}^{(0)}, a_{2,m}^{(0)}$ визначаються за апріорними відомостями про об'єкт.

Структурна схема рішення задачі ідентифікації.

Згідно до методу СНП, структурна схема задачі ідентифікації має вигляд, представлений на рис. 4.7.

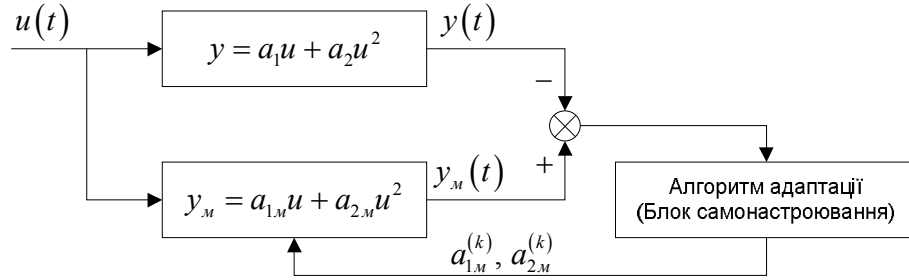


Рис. 4.7. Структурна схема системи ідентифікації статичного об'єкта

На вхід об'єкта і моделі подається вимірюваний сигнал $u(t)$. Помилка $\varepsilon(t)$ подається на блок обчислення $a_{1,m}, a_{2,m}$ згідно до алгоритму адаптації, для синтезу якого використовується градієнтний метод оптимізації.

Синтез градієнтного алгоритму ідентифікації

Запишемо градієнтний алгоритм оптимізації для задачі (4.18).

$$a_{i,m}^{(k+1)} = a_{i,m}^{(k)} - h \frac{\partial I}{\partial a_{i,m}}, \quad i = \overline{1,2}. \quad (4.19)$$

Критерій оптимальності з урахуванням виразів, що описують об'єкт і модель, приймає вигляд

$$I = (a_{1,m} + a_{2,m}u^2 - a_1u - a_2u^2)^2. \quad (4.20)$$

Визначимо частинні похідні у виразах (4.19).

$$\frac{\partial I}{\partial a_1} = 2 \cdot \varepsilon \cdot u; \quad \frac{\partial I}{\partial a_2} = 2 \cdot \varepsilon \cdot u^2. \quad (4.21)$$

Підставимо (4.21) у вираз (4.19). Алгоритм адаптивної ідентифікації приймає вигляд

$$\begin{aligned} a_{1,m}^{(k+1)} &= a_{1,m}^{(k)} - 2h\varepsilon u; \\ a_{2,m}^{(k+1)} &= a_{2,m}^{(k)} - 2h\varepsilon u^2. \end{aligned} \quad (4.22)$$

Приклад 4.6. Задача синтезу дискретної адаптивної системи автоматичного управління з еталонною моделлю (ДАСАУЕМ)

Структурна схема ДАСАУЕМ зображена на рис. 4.8.

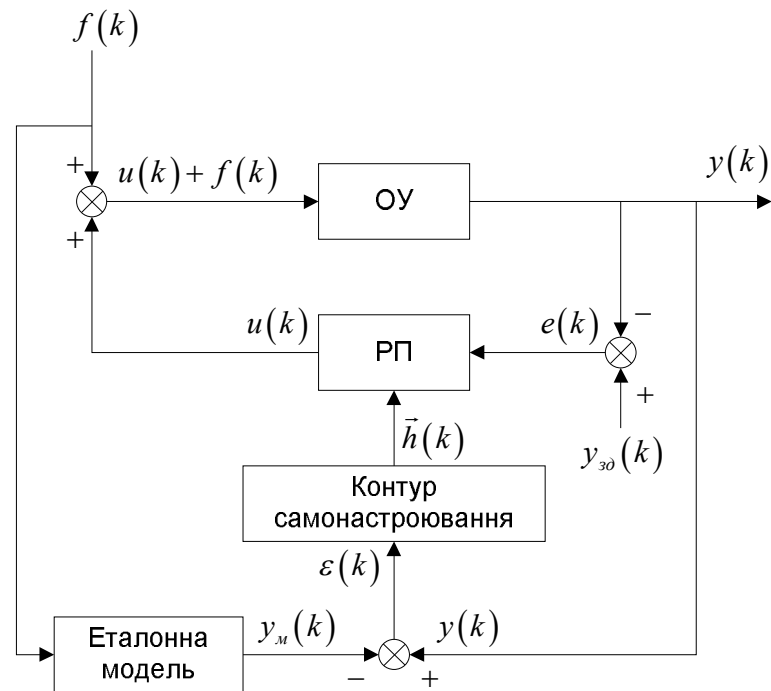


Рис. 4.8. Структурна схема ДАСАУЕМ

На рис. 4.8 прийнято наступні позначення:

ОУ – об’єкт управління зі змінними у часі параметрами;

РП – регулюючий пристрій (регулятор);

$y(k), y_m(k)$ – виходи об’єкта і моделі;

$e(k)$ – похибка регулювання;

$y_{зд}$ – завдання регулятора;

$\varepsilon(k) = y(k) - y_m(k)$ – помилка між виходом системи автоматичного регулювання і еталонної моделі;

$\vec{h}(k)$ – вектор параметрів настроювання регулятора.

Регулюючий пристрій має настройки, що відповідають деякому номінальному режиму функціонування об’єкта управління. При зміні параметрів об’єкта настройки регулятора стають неоптимальними і вимоги до показників якості системи регулювання не виконуються. Контур самонастроювання (КСН) на основі інформації про розбаланс $\varepsilon(k)$ коректує настройки регулятора РП таким чином, щоб $\varepsilon(k) \rightarrow 0$.

Змістовна постановка задачі синтезу ДАСАУЕМ

Знайти рівняння контуру самонастроювання, що забезпечує ефективну роботу системи регулювання при зміні параметрів об’єкта.

За допомогою еталонної моделі задається необхідний перехідний процес. при подачі збурення $f(k)$ виникають перехідні процеси в основній системі та в еталонній моделі. Якщо перехідні процеси відрізняються один від одного, то з’являється помилка $\varepsilon(k)$, що подається на КСН, вихідний сигнал якого змінює параметри регулятора до тих пір, поки $\varepsilon(k)$ не дорівнюватиме нулю, тоді $y(k) = y_m(k)$. Синтез КСН виконати з застосуванням градієнтного методу оптимізації критерію $I = \varepsilon^2(k+1)$.

Математична постановка задачі

Об’єкт управління описується рівнянням

$$y(k+1) + a_1 y(k) + a_2 y(k-1) = bu(k) + f(k). \quad (4.23)$$

Рівняння регулятора має вигляд

$$u(k) = -h_1 y(k) - h_2 y(k-1), \quad (4.24)$$

якщо прийняти, що $y_{\text{до}} = 0$. Тоді рівняння замкнутої АСР має наступний вигляд:

$$y(k+1) + (a_1 + bh_1)y(k) + (a_2 + bh_2)y(k-1) = f(k). \quad (4.25)$$

Рівняння еталонної моделі приймемо тої ж структури, що й рівняння основної АСР:

$$y_m(k+1) + a_{1,m}y_m(k) + a_{2,m}y_m(k-1) = f(k). \quad (4.26)$$

Таким чином, задача синтезу ДАСАУЕМ приймає вигляд

$$\begin{aligned} I &= \varepsilon^2(k+1) \rightarrow \min_{h_1, h_2}; \\ y(k+1) + (a_1 + bh_1)y(k) + (a_2 + bh_2)y(k-1) &= f(k); \\ y_m(k+1) + a_{1,m}y_m(k) + a_{2,m}y_m(k-1) &= f(k); \\ \varepsilon(k) &= y(k) - y_m(k) = \\ &= (a_1 + bh_1)y(k) + (a_2 + bh_2)y(k-1) - a_{1,m}y_m(k) - a_{2,m}y_m(k-1). \end{aligned} \quad (4.27)$$

Рішення задачі синтезу ДАСАУЕМ

Застосуємо градієнтний алгоритм оптимізації для рішення задачі (4.27). Тоді

$$h_1^{(k+1)} = h_1^{(k)} - \lambda \frac{\partial I}{\partial h_1};$$

$$h_2^{(k+1)} = h_2^{(k)} - \lambda \frac{\partial I}{\partial h_2};$$

Враховуючи вирази для I та $\varepsilon(k+1)$ в (4.27), отримаємо

$$\frac{\partial I}{\partial h_1} = \frac{\partial \varepsilon^2(k+1)}{\partial h_1} = 2 \cdot \varepsilon(k+1) \cdot \frac{\partial \varepsilon(k+1)}{\partial h_1};$$

$$\frac{\partial I}{\partial h_2} = \frac{\partial \varepsilon^2(k+1)}{\partial h_2} = 2 \cdot \varepsilon(k+1) \cdot \frac{\partial \varepsilon(k+1)}{\partial h_2}.$$

В даному випадку

$$\frac{\partial \varepsilon(k+1)}{\partial h_1} = by(k); \quad \frac{\partial \varepsilon(k+1)}{\partial h_2} = by(k-1).$$

Звідси остаточно отримаємо алгоритм контуру самонастроювання

$$\begin{aligned} h_1^{(k+1)} &= h_1^{(k)} - 2\lambda \varepsilon(k+1)by(k) \\ h_2^{(k+1)} &= h_2^{(k)} - 2\lambda \varepsilon(k+1)by(k-1) \end{aligned} \quad (4.28)$$

4.4. Контрольні завдання

1. В задачі безумовної оптимізації виконати один крок пошуку екстремуму простим градієнтним методом і обчислити значення цільової функції в знайденій точці.

$$I(u) = u^2 + 2u + 2 \rightarrow \min_u.$$

Пошук починати з точки $u^{(0)} = 1$. Параметр кроку в алгоритмі пошуку взяти $h=0,25$.

2. В задачі безумовної оптимізації виконати один крок градієнтного спуску і обчислити значення цільової функції $I(u_1^{(1)}, u_2^{(1)})$.

$$I(u) = u_1^2 + u_2^2 \rightarrow \min_{u_1, u_2};$$

$$u_i^{(1)} = u_i^{(0)} - h \frac{\partial I(u_1^{(0)}, u_2^{(0)})}{\partial u_i};$$

Початкові умови: $u_1^{(0)} = 4$; $u_2^{(0)} = 4$.

Параметр кроку: $h = 1$

3. Задача безумовної оптимізації функції має вигляд:

$$I(u) = u^2 + 2u + 2 \rightarrow \min_u.$$

Обчислити оптимальний крок методом Ньютона в алгоритмі градієнтного спуску.

4. Задано функцію

$$f(x) = 100(x_2 - x_1^2)^2 + (1 - x_1)^2$$

і дві перших точки, отриманих в процесі пошуку мінімуму функції $f(x)$:

$$x^{(0)} = [-1, 2; 1]^T, \quad x^{(1)} = [-1, 3; 1, 07]^T.$$

Визначте напрямки пошуку з точки $x^{(1)}$, користуючись наступними методами:

а) методом найшвидшого спуску,

б) методом Ньютона.

5. Побудувати лінію рівня функції $f(u_1, u_2) = (u_1 - 2)^2 + 4(u_2 - 1)^2 = 10$. Обчислити та побудувати градієнт в одній із точок лінії рівня.

6. Знайти мінімум функції $f(x_1, x_2) = x_1^2 - x_1 + x_2^2 - x_2 + 1$.

7. Знайти точки стаціонарності функції $f(x_1, x_2) = x_1 - x_1^2 - x_1^3 - x_1 x_2$ і виявити, які з них є точками максимуму.

8. Записати матрицю Гессе функції задачі 7 в точках її стаціонарності.

9. При апроксимації експериментально отриманих значень $y_i^E(x_i)$ лінійною залежністю за методом найменших квадратів розв'язується наступна екстремальна задача вибору коефіцієнтів a_0 і a_1 :

$$f_0(a_0, a_1) = \sum_i (a_0 + a_1 x_i - y_i^E(x_i))^2 \rightarrow \min_{a_0, a_1};$$

а) записати умови оптимальності для цієї задачі;

б) записати вираз для матриці Гессе цільової функції і показати, що матриця позитивно визначена.

в) знайти коефіцієнти a_0 і a_1 , якщо результати експерименту задані таблицею:

x_i	1	2	3	4
y_i^E	2	3	5	8

10. Виконати один цикл пошуку максимуму функції

$$f_0(x_1, x_2) = x_1 + x_2 - (x_1 x_2)^2 - x_1^2 - 2x_2^2$$

з початкової точки $\vec{x}^{(0)} = (1, 0)$, використовуючи наступні методи:

а) метод Коші;

б) градієнтний метод зі сталим кроком $h = 1$.