

ПРО ПРИШВИДШЕННЯ ОПТИМІЗАЦІЙНИХ МЕТОДІВ ДЛЯ ЗАДАЧІ СИНТЕЗУ БАГАТОШАРОВИХ ОПТИЧНИХ ПОКРИТТІВ

Ключові слова: багатошарові оптичні покриття, спектральні характеристики, синтез оптичних покриттів, аналітична похідна, узагальнений градієнт, метод золотого перерізу.

Ключевые слова: многослойные покрытия, спектральные характеристики, синтез оптических покрытий, аналитическая производная, обобщенный градиент, метод золотого сечения.

Вступ

Багатошарові оптичні покриття створюються шляхом напилення тонких діелектричних шарів, знаходять широке застосування в сучасних оптичних пристроях, пов'язаних з вивченням і формуванням оптичного випромінювання, а також в таких областях високих технологій, як лазерна техніка, оптоелектроніка, телекомунікації [1–5]. Першочерговим завданням при конструюванні таких структур залишається забезпечення максимальної селективності та експлуатаційної надійності таких структур під час використання мінімальної кількості тонкоплівкових шарів у структурах [6]. Такі задачі описано у дисертаційній роботі [7]. Розглядаються можливості, які дозволяють пришвидшити роботу відомих оптимізаційних заходів для синтезу багатошарових оптичних покриттів.

Постановка задачі

Задача синтезу багатошарових оптичних покриттів полягає в знаходженні таких параметрів багатошарового оптичного покриття — показників заломлення та геометричних товщин шарів $\vec{d} = (d_1, d_2, \dots, d_N)$ (N — кількість шарів), щоб функціонал, який введено для оцінки якості коефіцієнтів пропускання, був мінімальним на заданому спектральному діапазоні $[\lambda_1, \lambda_2]$:

$$F^* = F(\vec{n}^*, \vec{d}^*) = \min_{\vec{n}, \vec{d}} F(\vec{n}, \vec{d}) \quad (1)$$

при обмеженнях

$$n_{i\min} \leq n_i \leq n_{i\max}, \quad i = 1, 2, \dots, N, \quad (2)$$

$$d_{i\min} \leq d_i \leq d_{i\max}, \quad i = 1, 2, \dots, N, \quad (3)$$

де F^* — мінімальне значення функції якості покриття.

Функціями якості покриття в задачі (1)–(3) можуть бути:

$$F_1(\vec{n}, \vec{d}) = \frac{1}{L} \sum_{i=1}^L w_i (T(\vec{n}, \vec{d}, \lambda_i) - T_{ideal}(\lambda_i))^2, \quad (4)$$

$$F_2(\vec{n}, \vec{d}) = \frac{1}{L} \sum_{i=1}^L w_i |T(\vec{n}, \vec{d}, \lambda_i) - T_{ideal}(\lambda_i)|, \quad (5)$$

$$F_3(\vec{n}, \vec{d}) = \max_{i=1, \dots, L} w_i |T(\vec{n}, \vec{d}, \lambda_i) - T_{ideal}(\lambda_i)|, \quad (6)$$

де w_i — вагові коефіцієнти, що визначають внесок у цільову функцію при довжині хвилі λ_i , L — число точок сітки спектрального інтервалу від λ_1 до λ_2 , $T(\vec{n}, \vec{d}, \lambda_i)$ — значення коефіцієнта пропускання для параметрів (\vec{n}, \vec{d}) та довжини хвилі λ_i , $T_{ideal}(\lambda_i)$ — значення коефіцієнта пропускання, який потрібно досягнути при довжині хвилі λ_i .

Значення коефіцієнта пропускання енергії електромагнітної хвилі довжини λ через багат шарову оптичну структуру при падінні світла на поверхню під кутом θ_0 обчислюється за допомогою коефіцієнтів характеристичної матриці $M(\vec{n}, \vec{d}, \lambda)$ таким чином:

$$T(\vec{n}, \vec{d}, \lambda, \theta_0) = \frac{4}{2 + \frac{P_0}{P_s} M_{11}^2 + \frac{P_s}{P_0} M_{22}^2 + P_0 P_s M_{12}^2 + \frac{1}{P_0 P_s} M_{21}^2}, \quad (7)$$

де $p_0 = n_0 \cos \theta_0$ і $p_s = n_s \cos \theta_s$ — для ТЕ-хвилі (s — поляризація); $p_0 = \frac{n_0}{\cos \theta_0}$

і $p_s = \frac{n_s}{\cos \theta_s}$ — для ТМ-хвилі (p — поляризація); θ_0 — кут, під яким промінь падає на багат шарову структуру; θ_s — кут, під яким промінь виходить із багат шарової структури; n_0, n_s — показники заломлення зовнішнього середовища та підкладки відповідно.

Характеристична матриця N -шарової структури рівна добутку матриць кожного із шарів:

$$M(\vec{n}, \vec{d}, \lambda, \theta_0) = M(n_N, d_N, \lambda, \theta_N) \cdot M(n_{N-1}, d_{N-1}, \lambda, \theta_{N-1}) \cdots \cdots M(n_2, d_2, \lambda, \theta_2) \cdot M(n_1, d_1, \lambda, \theta_1), \quad (8)$$

де характеристична матриця одного шару рівна

$$M(n, d, \lambda, \theta) = \begin{vmatrix} \cos \delta(n, d, \lambda, \theta) & -\frac{i}{n} \cdot \sin \delta(n, d, \lambda, \theta) \\ -i \cdot n \cdot \sin \delta(n, d, \lambda, \theta) & \cos \delta(n, d, \lambda, \theta) \end{vmatrix},$$

$\delta(n, d, \lambda, \theta) = \frac{2 \cdot \pi \cdot n \cdot d \cdot \cos \theta}{\lambda}$ — фазова товщина шару, $i = \sqrt{-1}$, θ — кут між променем і нормаллю до площини падіння.

Кути між променем і нормаллю до площини падіння для кожного із шарів пов'язані законом Снеліуса і легко визначаються зі співвідношення

$$n_0 \sin \theta_0 = n_1 \sin \theta_1 = \cdots = n_j \sin \theta_j = \cdots = n_N \sin \theta_N = n_s \sin \theta_s.$$

Способи обчислення узагальненого градієнта

Розглянемо скінченно-різницевий метод обчислення градієнта функції $F(\vec{n}, \vec{d})$. Нехай \vec{e}_i — N -вимірний вектор, i -та компонента якого рівна одиниці, усі інші компоненти рівні нулю. Задаємо досить малі величини Δn_i та Δd_i , $i = 1, \dots, N$. Тоді компоненти градієнта функції $F(\vec{n}, \vec{d})$ в точці (\vec{n}_k, \vec{d}_k) будуть обчислюватись за допомогою формул

$$\nabla_i F(\vec{n}_k, \vec{d}_k) = \frac{F(\vec{n}_k + \Delta n_i \vec{e}_i, \vec{d}_k) - F(\vec{n}_k, \vec{d}_k)}{\Delta n_i}, \quad i = 1, \dots, N,$$

$$\nabla_{N+i} F(\vec{n}_k, \vec{d}_k) = \frac{F(\vec{n}_k, \vec{d}_k + \Delta d_i \vec{e}_i) - F(\vec{n}_k, \vec{d}_k)}{\Delta d_i}, \quad i = 1, \dots, N.$$

Твердження 1. Для обчислення градієнта функції $F(\vec{n}, \vec{d})$ скінченно-різницевий метод вимагає $N_0^* = (2N + 1) \cdot (N - 1) \cdot L$ операцій множення характеристичних матриць.

Доведення. Для знаходження значення $F(\vec{n}_k, \vec{d}_k)$ потрібно $(N - 1)$ операцій множення характеристичних матриць. Для кожного з $2N$ параметрів, для яких потрібно обчислити значення $F(\vec{n}_k + \Delta n_i \vec{e}_i, \vec{d}_k)$ або $F(\vec{n}_k, \vec{d}_k + \Delta d_i \vec{e}_i)$, потрібно також $(N - 1)$ таких операцій. Ці операції необхідно зробити для кожної з L вибраних значень довжин хвиль. Відповідно кількість операцій буде рівна

$$N_0^* = ((N - 1) + 2N(N - 1)) \cdot L = (2N + 1) \cdot (N - 1) \cdot L.$$

Твердження доведено.

Аналітичний спосіб обчислення градієнта функції $F(\vec{n}, \vec{d})$ позбавлений недоліків, властивих скінченно-різницевому способу. Безпосереднє диференціювання функції $F_1(\vec{n}, \vec{d})$ дає формули для знаходження компонента її градієнта в точці (\vec{n}_k, \vec{d}_k) :

$$\nabla_i F_1(\vec{n}_k, \vec{d}_k) = 2 \frac{1}{L} \sum_{j=1}^L (T(\vec{n}_k, \vec{d}_k, \lambda_j) - T_j) \frac{\partial T(\vec{n}_k, \vec{d}_k, \lambda_j)}{\partial n_i}, \quad i = 1, \dots, N,$$

$$\nabla_{N+i} F_1(\vec{n}_k, \vec{d}_k) = 2 \frac{1}{L} \sum_{j=1}^L (T(\vec{n}_k, \vec{d}_k, \lambda_j) - T_j) \frac{\partial T(\vec{n}_k, \vec{d}_k, \lambda_j)}{\partial d_i}, \quad i = 1, \dots, N,$$

та формули для знаходження узагальнених градієнтів:

$$\nabla_i F_2(\vec{n}_k, \vec{d}_k) = \frac{1}{L} \sum_{j=1}^L \text{sign}(T(\vec{n}_k, \vec{d}_k, \lambda_j) - T_j) \frac{\partial T(\vec{n}_k, \vec{d}_k, \lambda_j)}{\partial n_i}, \quad i = 1, \dots, N,$$

$$\nabla_{N+i} F_2(\vec{n}_k, \vec{d}_k) = \frac{1}{L} \sum_{j=1}^L \text{sign}(T(\vec{n}_k, \vec{d}_k, \lambda_j) - T_j) \frac{\partial T(\vec{n}_k, \vec{d}_k, \lambda_j)}{\partial d_i}, \quad i = 1, \dots, N,$$

$$\nabla_i F_3(\bar{n}_k, \bar{d}_k) = \text{sign}(T(\bar{n}, \bar{d}, \lambda_{j^*}) - T_{j^*}) \frac{\partial T(\bar{n}_k, \bar{d}_k, \lambda_{j^*})}{\partial n_i}, i = 1, \dots, N,$$

$$\nabla_{N+i} F_3(\bar{n}_k, \bar{d}_k) = \text{sign}(T(\bar{n}, \bar{d}, \lambda_{j^*}) - T_{j^*}) \frac{\partial T(\bar{n}_k, \bar{d}_k, \lambda_{j^*})}{\partial d_i}, i = 1, \dots, N,$$

де

$$j^* = \underset{j=1, \dots, L}{\text{argmax}} |T(\bar{n}, \bar{d}, \lambda_j) - T_j|.$$

У випадку цільової функції (4) це буде градієнт, а у випадку цільової функції (5), (6) — узагальнений градієнт. Узагальнений градієнт використовується при розв'язанні задач синтезу багаточарових оптичних покриттів за допомогою методів негладкої оптимізації [8, 9].

Отже, для аналітичного обчислення $\nabla F_1(\bar{n}_k, \bar{d}_k)$, $\nabla F_2(\bar{n}_k, \bar{d}_k)$ та $\nabla F_3(\bar{n}_k, \bar{d}_k)$ достатньо для всіх $i = 1, \dots, N$ уміти знаходити аналітичні вирази для похідних

$$\frac{\partial T(\bar{n}_k, \bar{d}_k, \lambda_j)}{\partial n_i} \text{ та } \frac{\partial T(\bar{n}_k, \bar{d}_k, \lambda_j)}{\partial d_i} \text{ при фіксованому значенні } \lambda. \text{ Використовуючи ці}$$

похідні (7), легко знайти для кожного i ($1 \leq i \leq N$) за таким правилом [10]:

$$\frac{\partial T(\bar{n}, \bar{d}, \lambda)}{\partial n_i} = \frac{8 \left(\frac{n_0}{n_s} m_{11} \frac{\partial m_{11}}{\partial n_i} + \frac{n_s}{n_0} m_{22} \frac{\partial m_{22}}{\partial n_i} + n_0 n_s m_{12} \frac{\partial m_{12}}{\partial n_i} + \frac{1}{n_0 n_s} m_{21} \frac{\partial m_{21}}{\partial n_i} \right)}{\left(2 + \frac{n_0}{n_s} m_{11}^2 + \frac{n_s}{n_0} m_{22}^2 + n_0 n_s m_{12}^2 + \frac{1}{n_0 n_s} m_{21}^2 \right)^2},$$

$$\frac{\partial T(\bar{n}, \bar{d}, \lambda)}{\partial d_i} = \frac{8 \left(\frac{n_0}{n_s} m_{11} \frac{\partial m_{11}}{\partial d_i} + \frac{n_s}{n_0} m_{22} \frac{\partial m_{22}}{\partial d_i} + n_0 n_s m_{12} \frac{\partial m_{12}}{\partial d_i} + \frac{1}{n_0 n_s} m_{21} \frac{\partial m_{21}}{\partial d_i} \right)}{\left(2 + \frac{n_0}{n_s} m_{11}^2 + \frac{n_s}{n_0} m_{22}^2 + n_0 n_s m_{12}^2 + \frac{1}{n_0 n_s} m_{21}^2 \right)^2},$$

де $\frac{\partial m_{11}}{\partial n_i}$, $\frac{\partial m_{12}}{\partial n_i}$, $\frac{\partial m_{21}}{\partial n_i}$, $\frac{\partial m_{22}}{\partial n_i}$, $\frac{\partial m_{11}}{\partial d_i}$, $\frac{\partial m_{12}}{\partial d_i}$, $\frac{\partial m_{21}}{\partial d_i}$, $\frac{\partial m_{22}}{\partial d_i}$ — коефіцієнти наступних

(2 × 2)-матриць:

$$\frac{\partial M(\bar{n}, \bar{d}, \lambda)}{\partial n_i} = \begin{vmatrix} \frac{\partial m_{11}}{\partial n_i} & i \frac{\partial m_{12}}{\partial n_i} \\ i \frac{\partial m_{21}}{\partial n_i} & \frac{\partial m_{22}}{\partial n_i} \end{vmatrix};$$

$$\frac{\partial M(\bar{n}, \bar{d}, \lambda)}{\partial d_i} = \begin{vmatrix} \frac{\partial m_{11}}{\partial d_i} & i \frac{\partial m_{12}}{\partial d_i} \\ i \frac{\partial m_{21}}{\partial d_i} & \frac{\partial m_{22}}{\partial d_i} \end{vmatrix}.$$

Матриці $\frac{\partial M(\vec{n}, \vec{d}, \lambda)}{\partial n_i}$ та $\frac{\partial M(\vec{n}, \vec{d}, \lambda)}{\partial d_i}$ легко розрахувати, використовуючи (8):

$$\begin{aligned} \frac{\partial M(\vec{n}, \vec{d}, \lambda)}{\partial n_1} &= \frac{\partial M(\vec{n}_1, \vec{d}_1, \lambda)}{\partial n_1} \prod_{k=2}^N M(n_k, d_k, \lambda), \\ \frac{\partial M(\vec{n}, \vec{d}, \lambda)}{\partial n_i} &= \prod_{k=1}^{i-1} M(n_k, d_k, \lambda) \frac{\partial M(n_i, d_i, \lambda)}{\partial n_i} \prod_{k=i+1}^N M(n_k, d_k, \lambda), \quad i = 2, \dots, N-1, \\ \frac{\partial M(\vec{n}, \vec{d}, \lambda)}{\partial n_N} &= \prod_{k=1}^{N-1} M(n_k, d_k, \lambda) \frac{\partial M(n_N, d_N, \lambda)}{\partial n_N}, \\ \frac{\partial M(\vec{n}, \vec{d}, \lambda)}{\partial d_1} &= \frac{\partial M(n_1, d_1, \lambda)}{\partial d_1} \prod_{k=2}^N M(n_k, d_k, \lambda), \\ \frac{\partial M(\vec{n}, \vec{d}, \lambda)}{\partial d_i} &= \prod_{k=1}^{i-1} M(n_k, d_k, \lambda) \frac{\partial M(n_i, d_i, \lambda)}{\partial d_i} \prod_{k=i+1}^N M(n_k, d_k, \lambda), \quad i = 2, \dots, N-1, \\ \frac{\partial M(\vec{n}, \vec{d}, \lambda)}{\partial d_N} &= \prod_{k=1}^{N-1} M(n_k, d_k, \lambda) \frac{\partial M(n_N, d_N, \lambda)}{\partial d_N}, \end{aligned} \quad (9)$$

де матриця (2×2) — матриці $\frac{\partial M(\vec{n}, \vec{d}, \lambda)}{\partial n_i}$ та $\frac{\partial M(\vec{n}, \vec{d}, \lambda)}{\partial d_i}$, для кожного $i = 1, \dots, N$ матимуть наступний вигляд:

$$\begin{aligned} \frac{\partial M(n_i, d_i, \lambda)}{\partial n_i} &= \begin{vmatrix} \frac{\partial}{\partial n_k} \left(\cos \left(\frac{2\pi n_k d_k}{\lambda_j} \right) \right) & -i \frac{\partial}{\partial n_k} \left(\frac{1}{n_k} \sin \left(\frac{2\pi n_k d_k}{\lambda_j} \right) \right) \\ -i \frac{\partial}{\partial n_k} \left(n_k \sin \left(\frac{2\pi n_k d_k}{\lambda_j} \right) \right) & \frac{\partial}{\partial n_k} \left(\cos \left(\frac{2\pi n_k d_k}{\lambda_j} \right) \right) \end{vmatrix} = \\ &= \begin{vmatrix} -\frac{2\pi d_i}{\lambda_j} \sin \left(\frac{2\pi n_i d_i}{\lambda_j} \right) & -\frac{i}{n_i} \frac{2\pi d_i}{\lambda_j} \cos \left(\frac{2\pi n_i d_i}{\lambda_j} \right) \\ -i n_i \frac{2\pi d_i}{\lambda_j} \cos \left(\frac{2\pi n_i d_i}{\lambda_j} \right) & -\frac{2\pi d_i}{\lambda_j} \sin \left(\frac{2\pi n_i d_i}{\lambda_j} \right) \end{vmatrix}, \\ \frac{\partial M(n_i, d_i, \lambda)}{\partial d_i} &= \begin{vmatrix} \frac{\partial}{\partial d_k} \left(\cos \left(\frac{2\pi n_k d_k}{\lambda_j} \right) \right) & -i \frac{\partial}{\partial d_k} \left(\frac{1}{n_k} \sin \left(\frac{2\pi n_k d_k}{\lambda_j} \right) \right) \\ -i \frac{\partial}{\partial d_k} \left(n_k \sin \left(\frac{2\pi n_k d_k}{\lambda_j} \right) \right) & \frac{\partial}{\partial d_k} \left(\cos \left(\frac{2\pi n_k d_k}{\lambda_j} \right) \right) \end{vmatrix} = \end{aligned}$$

$$= \begin{vmatrix} -\frac{2\pi n_i}{\lambda_j} \sin\left(\frac{2\pi n_i d_i}{\lambda_j}\right) & -\frac{i}{n_i} \frac{2\pi n_i}{\lambda_j} \cos\left(\frac{2\pi n_i d_i}{\lambda_j}\right) \\ -in_i \frac{2\pi n_i}{\lambda_j} \cos\left(\frac{2\pi n_i d_i}{\lambda_j}\right) & -\frac{2\pi n_i}{\lambda_j} \sin\left(\frac{2\pi n_i d_i}{\lambda_j}\right) \end{vmatrix}.$$

Аналітичний спосіб дозволяє обчислити точне (у межах комп'ютерної арифметики) значення градієнта функції $F(\vec{n}, \vec{d})$ в точці (\vec{n}_k, \vec{d}_k) , за затратами арифметичних операцій він такий, як і в скінченно-різницевому способі. Аналітичний спосіб доцільніше використовувати, ніж скінченно-різницевий, коли потрібна висока точність знаходження оптимальних параметрів (\vec{n}^*, \vec{d}^*) .

Твердження 2. Для обчислення градієнта функції $F(\vec{n}, \vec{d})$ аналітичним способом кількість операцій множення характеристичних матриць рівна $N_1^* = 2N(N-1)L$.

Пришвидшення оптимізаційних методів

Знаходження значення функції якості та її градієнта (узагальненого градієнта) є основними операціями при розв'язанні задачі синтезу оптичних покриттів. Відповідно для пришвидшення роботи методів багатовимірного пошуку пропонується використати такі вдосконалення.

1. Використати аналітичну похідну при використанні градієнтних методів. Операції множення характеристичних матриць найбільш ресурсозатратні при знаходженні похідних. Перевагою аналітичного підходу є не лише менша на $(N-1)$ кількість таких операцій, а насамперед — більша точність і, як наслідок, менша кількість ітерацій.

2. Пришвидшене множення матриць. Пришвиджена реалізація [11], при якій забирається повторюване множення матриць для різних $i = 1, \dots, N$, властива прямолінійній реалізації. Суть прискореної реалізації полягає в наступному. Обчислюємо такі префікс-матриці:

$$M_i^+ = \prod_{k=1}^i M(n_k, d_k, \lambda_j), i = 1, \dots, N-1.$$

Обчислюємо такі суфікс-матриці:

$$M_i^- = \prod_{k=1}^N M(n_k, d_k, \lambda_j), i = 2, \dots, N.$$

Для цього досить $2(N-2)$ операцій помноження матриць розміром (2×2) .

Далі для знаходження матриць $\frac{\partial M(\vec{n}, \vec{d}, \lambda)}{\partial n_i}$ та $\frac{\partial M(\vec{n}, \vec{d}, \lambda)}{\partial d_i}$ для всіх $i = 1, \dots, N$

замість (9) використовуємо такі формули:

$$\frac{\partial M(\vec{n}, \vec{d}, \lambda_j)}{\partial n_1} = \frac{\partial M(n_1, d_1, \lambda_j)}{\partial n_1} \cdot M_2^-,$$

$$\begin{aligned}
\frac{\partial M(\vec{n}, \vec{d}, \lambda_j)}{\partial n_i} &= M_{i-1}^+ \cdot \frac{\partial M(n_i, d_i, \lambda_j)}{\partial n_i} \cdot M_{i+1}^-, \quad i = 2, \dots, N-1, \\
\frac{\partial M(\vec{n}, \vec{d}, \lambda_j)}{\partial n_N} &= M_{N-1}^- \cdot \frac{\partial M(n_N, d_N, \lambda_j)}{\partial n_N}, \\
\frac{\partial M(\vec{n}, \vec{d}, \lambda_j)}{\partial d_1} &= \frac{\partial M(n_1, d_1, \lambda_j)}{\partial d_1} \cdot M_2^-, \\
\frac{\partial M(\vec{n}, \vec{d}, \lambda_j)}{\partial d_i} &= M_{i-1}^+ \cdot \frac{\partial M(n_i, d_i, \lambda_j)}{\partial d_i} \cdot M_{i+1}^-, \quad i = 2, \dots, N-1, \\
\frac{\partial M(\vec{n}, \vec{d}, \lambda_j)}{\partial d_N} &= M_{N-1}^- \cdot \frac{\partial M(n_N, d_N, \lambda_j)}{\partial d_N}, \tag{10}
\end{aligned}$$

яка є також формулою (9), але записана в більш компактному для практичного застосування вигляді.

Твердження 3. Для обчислення градієнта функції $F(\vec{n}, \vec{d})$ пришвидшеним аналітичним способом потрібно $N_2^* = 2(3N - 4)L$ множень характеристичних матриць.

Доведення. Підрахуємо кількість операцій для однієї довжини хвилі. Для знаходження допоміжних матриць M_i^+ ($i = 1, \dots, N-1$) та M_i^- ($i = 2, \dots, N$) потрібно всього $2(N-2)$ операцій, тобто $(N-2)$ для кожної. Для реалізації формули (10) потрібно $2(2 + 2(N-2)) = 4(N-1)$ множень характеристичних матриць.

Тому для обчислення матриць $\frac{\partial M(\vec{n}, \vec{d}, \lambda)}{\partial n_i}$ та $\frac{\partial M(\vec{n}, \vec{d}, \lambda)}{\partial d_i}$ для всіх $i = 1, \dots, N$ прискорена реалізація потребує

$$N_2^* = (2(N-2) + 4(N-1))L = 2(3N-4)L$$

операцій множення матриць.

Твердження доведено.

Твердження 4. Пришвидшений варіант аналітичного способу обчислення градієнта функції $F(\vec{n}, \vec{d})$ виграє у прямолінійної реалізації, і цей виграш буде характеризуватися величиною

$$q(N) = \frac{3}{N} - \frac{1}{N(N-1)},$$

де N — кількість шарів.

Доведення. Знайдемо відповідне співвідношення кількостей операцій множення характеристичних матриць:

$$q(N) = \frac{N_2^*}{N_1^*} = \frac{2(3N-4)L}{2N(N-1)L} = \frac{3(N-1)-1}{N(N-1)} = \frac{3}{N} - \frac{1}{N(N-1)}.$$

Твердження доведено.

Аналогічне твердження також можна сформулювати і для обчислення градієнта функції $F(\vec{n}, \vec{d})$ скінченно-різницевим методом (див. твердження 1).

Для обчислення градієнта функції $F(\vec{n}, \vec{d})$ пришвидшеним скінченно-різницевим методом кількість операцій множень характеристичних матриць буде

$$N_3^* = (2(3N - 4) + (N - 1))L.$$

Розглянемо перевагу пришвидшеного множення матриць. При $N = 2$ маємо $q(2) = 1$, а при $N = 3$ маємо $q(3) = \frac{5}{6}$. Однак при $N = 60$ маємо суттєвий виграш:

$$q(60) = \frac{3 \cdot 60 - 4}{60 \cdot 59} = \frac{176}{3540} \approx \frac{1}{20}.$$

Це означає, що при роботі зі шістдесятишаровим оптичним покриттям можна зекономити у двадцять разів на обчисленнях, пов'язаних із множенням матриць. Однак потрібно відзначити, що при розв'язанні оптимізаційної задачі (1)–(3) ми не отримаємо «чистого» виграшу за часом у $q(N)$ разів. Річ у тім, що розрахунок характеристичних матриць для кожного шару потребує використання трудомістких щодо обчислення математичних функцій $\sin(x)$ та $\cos(x)$, які можуть перевершувати складні операції, пов'язані з множенням матриць. Тому застосуємо прийом, який дозволить зменшити час виконання цих операцій.

3. Табуляція тригонометричних функцій ($\sin(x)$, $\cos(x)$). У роботі значення косинусів та синусів потрібно знаходити від функції

$$\delta(n, d, \lambda) = \frac{2\pi nd}{\lambda}.$$

Якщо значення $n \in [n_{\min}, n_{\max}]$, $d \in [d_{\min}, d_{\max}]$ та $\lambda \in [\lambda_{\min}, \lambda_{\max}]$, то діапазон значень цієї функції буде $\left[\frac{2\pi n_{\min} d_{\min}}{\lambda_{\max}}, \frac{2\pi n_{\max} d_{\max}}{\lambda_{\min}} \right]$. Розділимо його не менше, ніж на 10^7 частин, і протабулюємо значення синусів та косинусів. Перевіримо на різних комп'ютерах, наскільки збільшилась швидкодія при 10^9 операцій множення характеристичних матриць (табл. 1). Для цього була написана програма-автоматизатор, яка виконувала цю дію 100 разів, і визначався середній час.

Таблиця 1

№	Характеристики комп'ютера	Час виконання без табуляції, с	Час виконання з використанням табуляції, с
1	Intel(R) Core(TM) i3-3220 3.30 ГГц ОЗУ 8 ГБайт	742,22	7,98
2	AMD Ryzen 3 3250U with Radeon Graphics 2.60 ГГц ОЗУ 8 ГБайт	853,74	10,04
3	Intel(R) Core(TM) i3-3250 3.50 ГГц ОЗУ 4 ГБайт	1008,85	6,94
4	Intel(R) Pentium(R) CPU 2.2 ГГц ОЗУ 4 ГБайт	4253,16	31,74

З табл. 1 видно, що швидкість виконання суттєво збільшилась. Для комп'ютера з процесором AMD Ryzen 3 3250U with Radeon Graphics 2.60 ГГц ОЗУ 8 ГБайт час виконання зменшився приблизно у 85 разів, а для комп'ютера з процесором Intel(R) Core(TM) i3-3250 3.50 ГГц ОЗУ 4 ГБайт процес множення характеристичних матриць прискорився понад 140 разів. Відзначимо, що під час зміни кількості операцій множення характеристичних матриць час виконання без табуляції та з табуляцією збереже вказане співвідношення.

Твердження 5. Наявність табуляцій значень тригонометричних функцій при виконанні операцій множення характеристичних матриць у задачі (1)–(3) зменшує час виконання у десятки разів.

4. Використання методу золотого перерізу при визначенні кроку. Крок α_k^* в ітераційному процесі визначається із розв'язку задачі

$$F(x^k + \alpha_k^* p^k) = \min_{\alpha^k} F(x^k + \alpha^k p^k).$$

Це задача одновимірної оптимізації.

Порівняємо ефективність методів одновимірної оптимізації — золотого перерізу та тернарний пошук. Для цього розв'яжемо окрему задачу: знайти розміщення точки, від якої відстань до прямих була б якомога меншою. Під відстанню до прямих будемо розуміти відстань до найбільш віддаленої прямої. Прямі задані двома точками.

Спочатку задамо структуру, яка буде описувати лінію в програмі

```
struct line {
    double a, b, c;
    line (double x1, double y1, double x2, double y2) {
        double aa = (y2 - y1);
        double bb = (x1 - x2);
        double cc = x2 * (y1 - y2) + y2 * (x2 - x1);
        double norm = sqrt(aa * aa + bb * bb);
        this->a = aa / norm;
        this->b = bb / norm;
        this->c = cc / norm;
    }
    double get(double x, double y) {
        return abs(a * x + b * y + c);
    }
};
```

В описаній структурі вже зразу наведено метод, який повертає відстань від точки до прямої. Як відомо, відстань від точки (x_0, y_0) до i -ї прямої

$a_i \cdot x + b_i \cdot y + c_i = 0$ рівна

$$\text{dist}(x_0, y_0, a_i, b_i, c_i) = \frac{|a_i \cdot x_0 + b_i \cdot y_0 + c_i|}{\sqrt{a_i^2 + b_i^2}}.$$

Поділивши на $\sqrt{a_i^2 + b_i^2}$ значення коефіцієнтів прямої $a'_i = \frac{a_i}{\sqrt{a_i^2 + b_i^2}},$

$b'_i = \frac{b_i}{\sqrt{a_i^2 + b_i^2}}, c'_i = \frac{c_i}{\sqrt{a_i^2 + b_i^2}},$ отримаємо, що відстань

$$\text{dist}(x_0, y_0, a_i, b_i, c_i) = |a'_i x_0 + b'_i y_0 + c'_i|.$$

Структура, яка описує пряму, готова до використання, і швидкість знаходження відстані від точки до прямої зростає після уникнення постійного визначення кореня квадратного та поділу на нього.

Введемо дані та сформуємо всю інформацію про прямі у векторі

```
vector<line> lines;
int n; cin >> n;
for (int i = 0; i < n; i++) {
    int x1, y1, x2, y2;
    cin >> x1 >> y1 >> x2 >> y2;
    line s(x1, y1, x2, y2);
    lines.push_back(s);
}
```

Відстань від точки до прямих знайдемо за допомогою функції:

```
double dist(double x, double y) {
    double ans = 0.0;
    for (auto i : lines) {
        ans = max(ans, i.get(x, y));
    }
    return ans;
}
```

З огляду на введені вище позначення, цільова функція має вигляд

$$f(x, y) = \max_{i=1,n} dist(x, y, a_i, b_i, c_i) \rightarrow \min_{x \in R, y \in R}$$

Дослідимо, як буде зображена ця цільова функція в просторі. Типовий вигляд цільової функції зображено на рис. 1.

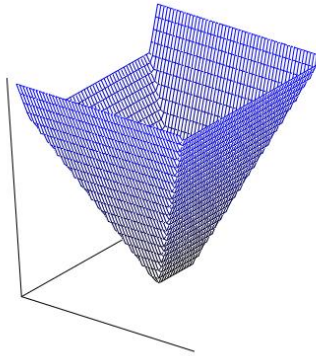


Рис. 1

Якщо записати функцію відстані від точки до прямих при фіксації одного з параметрів x чи y , то можна звернути увагу, що вона унімодальна, тобто неперервна і при зміні незафіксованого параметра спочатку спадає, а потім зростає (рис. 2). Тому застосуємо для знаходження мінімального значення цільової функції метод золотого перерізу.

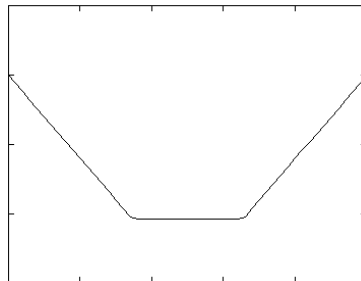


Рис. 2

Зуважимо, що цей метод можна застосовувати по черзі — спочатку до одного, а потім — до іншого параметра. Але такий підхід потребує великої кількості ітерацій, поки наблизиться до шуканого розв'язку. Тому використаємо те, що цільова функція при кожному фіксованому значенні змінної x унімодальна. Будемо всередині кожного кроку для змінної x знаходити оптимальне значення цільової функції зразу ж і щодо змінної y (рис. 3). Це суттєво пришвидшить пошук розв'язку. Для визначення шуканого значення щодо змінної x щоразу запускається метод золотого перерізу за змінною y , і досягнуте значення є індикатором вибору значення звуження проміжку за змінною x .

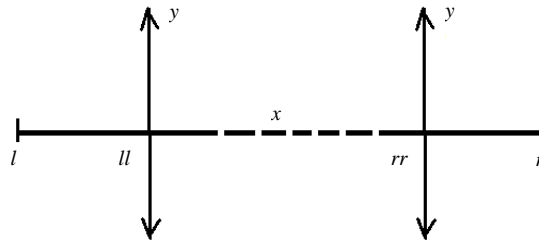


Рис. 3

Відповідна реалізація наведена нижче:

```
double Golden_Section_y (double x) {
    double l = -1e9, r = 1e9;
    while (r - l > eps) {
        double ll = l + 0.38 * (r - l);
        double rr = l + 0.62 * (r - l);
        if (dist(x, ll) < dist(x, rr))
            r = rr;
        else
            l = ll;
    }
    return l;
}

double Golden_Section_x () {
    double l = -1e9, r = 1e9;
    while (r - l > eps) {
        double ll = l + 0.38 * (r - l);
        double rr = l + 0.62 * (r - l);
        if (dist(ll, Golden_Section_y(ll)) < dist(rr, Golden_Section_y(rr)))
            r = rr;
        else
            l = ll;
    }
    return l;
}
```

Часова складність алгоритму рівна $O(n \cdot \log^2 C)$, де C — параметр, який залежить від діапазону пошуку та точності. Квадрат операцій виникає через те, що один метод золотого перерізу використовується в іншому.

Порівняємо метод золотого перерізу з іншим — тернарним пошуком. Відзначимо, що при заміні коефіцієнтів у методі золотого перерізу 0,38 на 1/3, а 0,62 на 2/3 отримаємо метод тернарного пошуку. У системі Codeforces проведено замір часу знаходження розв'язків для обох цих методів, коли розмірність задачі більша і йтиме від 10000 до 100000 з кроком 10000.

Як видно з рис. 4, існує лінійна залежність часу виконання розглядуваних методів від розмірності задачі. Метод тернарного пошуку потребує приблизно на 40 % більше часових затрат, ніж метод золотого перерізу при знаходженні розв'язку розглядуваної задачі. Також важливим показником є те, що кількість

ітерацій методу золотого перерізу для визначення шуканого значення змінної x у діапазоні $[-10^9, 10^9]$ становитиме 74, а для методу тернарного пошуку — 87.

Графіки залежності часу виконання тернарного пошуку та методу золотого перерізу від розмірності задачі приведені на рис. 4.

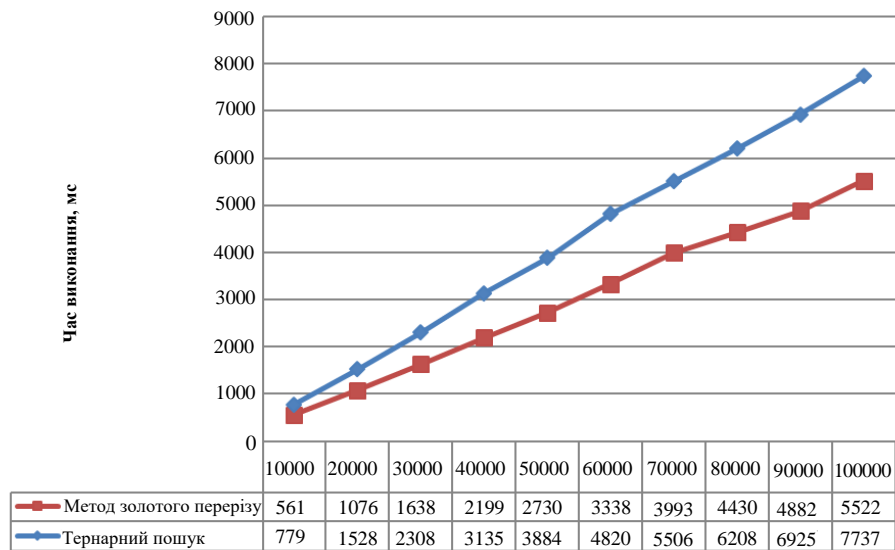


Рис. 4

5. Ефективна реалізація множення двох матриць. Для задач, пов'язаних з оптимізацією шаруватих структур в задачах теплопровідності та радіації, потрібно виконувати множення матриць великої розмірності. Тут може з'явитися ще один пункт пришвидшення роботи методів багатовимірного пошуку, який пов'язаний з особливістю реалізації операції множення матриць. Розглянемо дві реалізації цієї операції.

Таблиця 2

Загальновідомий варіант	Пришвидшений варіант
<pre>for (int i = 0; i < n; i++) for (int j = 0; j < n; j++) for (int k = 0; k < n; k++) c[i][j] += a[i][k] * b[k][j];</pre>	<pre>for (int i = 0; i < n; i++) for (int k = 0; k < n; k++) { long long x = a[i][k]; for (int j = 0; j < n; j++) c[i][j] += x * b[k][j]; }</pre>

У табл. 2 описано знаходження добутку матриць $C = A \cdot B$, усі матриці розмірностей $n \times n$. Проведемо для цих двох реалізацій обчислювальний експеримент, використавши тестувальну систему ejudge 3.7.9+.

Для проведення обчислювального експерименту підготовлено десять тестів для матриць розмірності 1000×1000 . Час проходження кожного тесту при загальновідомій реалізації перевищує 15 с. При пришвидшеній реалізації він суттєво менший, ніж при загальновідомій, і максимального витрачений час рівний 1,388 с. Відзначимо, що час зчитування вхідних даних і виводу відповіді становить приблизно 350 мс. Відповідно, якщо у загальновідомому варіанті поміняти другий та третій цикли місцями і зафіксувати у звичайній змінній значення елемента першої матриці, то швидкість виконання операції множення для матриць суттєво зросте. Це пришвидшення становить від 2 до 15 разів — залежно від характеристик комп'ютера. При збільшенні розмірності матриць перевага пришвидшеного варіанту зростатиме. Можливі й подальші покращення [12], які можуть дати перевагу ще на 20 %.

Висновок

Для пришвидшення роботи оптимізаційних методів для синтезу багат шарових оптичних покриттів запропоновано використати: пришвидшений спосіб для знаходження узагальненого градієнта цільової функції за допомогою використання префікс- та суфікс-масивів у аналітичному способі обчислення градієнта, табуляцію значень тригонометричних функцій, швидке множення матриць та використання ефективного методу при одновимірній оптимізації. Зокрема показано, що при виконанні 10^9 операцій множення характеристичних матриць використання табуляції значень $\sin(x)$ та $\cos(x)$ зменшує час виконання в десятки разів. Своєю математичною природою задачі синтезу оптичних шаруватих покриттів близькі до цілого ряду задач синтезу в інших областях: електродинаміці, радіофізиці, акустиці, оскільки хвильові процеси в них та їх основні характеристики описуються тими ж рівняннями, що і в оптиці. Тому математичні методи і підходи до розв'язання задач синтезу, пов'язаних з оптичними шаруватими покриттями, можна поширити і на задачі синтезу з інших розділів фізики та техніки.

О.В. Міца, П.І. Стецюк, О.М. Левчук, В.І. Пецько, І.Ф. Повхан

ПРО ПРИШВИДШЕННЯ ОПТИМІЗАЦІЙНИХ МЕТОДІВ ДЛЯ ЗАДАЧІ СИНТЕЗУ БАГАТОШАРОВИХ ОПТИЧНИХ ПОКРИТТІВ

Розглянуто п'ять способів для пришвидшення багатовимірного пошуку розв'язку задачі синтезу багат шарових оптичних покриттів за допомогою методів нульового та першого порядків. Перший спосіб — це використання аналітичної похідної для цільової функції якості багат шарового покриття. Він дозволяє точно (у межах комп'ютерної арифметики) обчислити значення градієнта гладкої цільової функції та узагальненого градієнта негладкої цільової функції. Перший спосіб потребує такої ж кількості арифметичних операцій, як і скінченно-різницьові способи обчислення градієнта та узагальненого градієнта. Другий спосіб — це використання пришвидшеного знаходження градієнта цільової функції за допомогою використання префікс- та суфікс-масивів у аналітичному способі обчислення градієнта. Цей прийом дозволяє зменшити кількість арифметичних операцій втричі для задач великої розмірності. Третій спосіб — це використання табуляції значень тригонометричних функцій для обчислення характеристичних матриць. Цей прийом зменшує час виконання операцій множення характеристичних матриць у десятки разів в залежності від характеристик комп'ютера. Для деяких архітектур комп'ютера ця перевага становить більше ніж 140 разів. Четвертий спосіб — це використання методу золотого перерізу для одновимірної оптимізації в задачах синтезу оптичних покриттів. Зокрема, при розв'язанні однієї часткової задачі показано, що метод тернарного пошуку потребує приблизно на 40 % більше часових затрат, ніж метод золотого перерізу. П'ятий спосіб — це використання ефективної реалізації множення двох матриць. Вона полягає у зміні порядку другого і третього циклів для загальновідомого методу множення двох матриць та фіксації у звичайній змінній значення елемента першої матриці. Це дозволяє суттєво прискорити виконання операції множення двох матриць. Для матриць розмірності 1000×1000 пришвидшення складає від 2 до 15 разів — залежно від характеристик комп'ютера.

Ключові слова: багат шарові оптичні покриття, спектральні характеристики, синтез оптичних покриттів, аналітична похідна, узагальнений градієнт, метод золотого перерізу.

ON THE ACCELERATION OF OPTIMIZATION METHODS FOR THE PROBLEM OF SYNTHESIS OF MULTILAYER OPTICAL COATINGS

Five ways to speed up the multidimensional search in order to solve the problem of synthesis of multilayer optical coatings by using the methods of zero and first orders have been considered. The first way is to use an analytical derivative for the target quality function of the multilayer coating. It allows us to calculate accurately (within the computer arithmetic) the value of the gradient of a smooth objective function and generalized gradient of a non-smooth objective one. The first way requires the same number of arithmetic operations as well as finite-difference methods of calculating the gradient and the generalized gradient. The second way is to use a speedy finding of the objective function gradient using the prefix- and suffix-arrays in the analytical method of calculating the gradient. This technique allows us to reduce the number of arithmetic operations thrice for large-scale problems. The third way is the use of tabulating the values of trigonometric functions to calculate the characteristic matrices. This technique reduces the execution time of multiplication operations of characteristic matrices ten times depending on the computer's specifications. For some computer architectures, this advantage is more than 140 times. The fourth method is the use of the golden section method for the one-dimensional optimization in the problems of synthesis of optical coatings. In particular, when solving one partial problem it is shown that the ternary search method requires approximately 40% more time than the golden section method. The fifth way is to use the effective implementation of multiplication of two matrices. It lies in changing the order of the second and third cycles for the well-known method of multiplying two matrices and fixing in a common variable value of the element of the first matrix. This allows us to speed up significantly the multiplication operation of two matrices. For matrices having 1000 x 1000 dimension the acceleration is from 2 to 15 times, depending on the computer's specifications.

Keywords: multilayer optical coatings, spectral characteristics, synthesis of optical coatings, an analytical derivative, the generalised gradient, the golden section method.

1. Willey R.R. Practical production of optical thin films. Willey Optical, Consultants: Charlevoix, MI, USA. 2008. 419 p.
2. Macleod H.A. Thin film optical filters. New York : McGraw-Hill. 1986. P. 260–261.
3. Kryuchyn A.A., Petrov V.V., Rubish V.M., Trunov M.L., Lytvyn P.M. and Kostyukevich S.A. Formation of nanoscale structures on chalcogenide films. *Phys. Status Solidi B*. 2017. DOI: 10.1002/pssb.201700405.
4. Liu Q, Gerling LG, Bernal-Texca F, Toudert J, Li T, Zhan X, Martorell J. Light harvesting at oblique incidence decoupled from transmission in organic solar cells exhibiting 9.8 % efficiency and 50 % visible light transparency. *Advanced Energy Materials*. 2020. **10**. 1904196. P. 1–9.
5. Dobrowolski J.A., Tikhonravov A.V., Tmbeckov M.K., Sullivan B.T., Verly P.G. Optimal normal-incidence antireflection coatings. *Appl. Opt.*, 1996. **35**. P. 644–658.
6. Furman, Sh.A., Tikhonravov A.V. Basics of optics of multilayer systems. Edition Frontieres, Gif-sur-Yvette. 1992. 242 p.
7. Міца О.В. Моделювання та оптимізація спектральних коефіцієнтів шаруватих оптичних систем з неоднорідними границями. Рукопис дисертації на здобуття наукового ступеня доктора технічних наук. Ін-т кібернетики ім. В.М. Глушкова НАН України. Київ, 2021. 298 с.
8. Сергиенко И.В., Стецюк П.И. О трех научных идеях Н.З. Шора. *Кибернетика и системный анализ*. 2012. № 1. С. 4–22.
9. Стецюк П.И. Теория и программные реализации r -алгоритмов Шора. *Кибернетика и системный анализ*. 2017. **53**, № 5. С. 43–57.
10. Стецюк П.И., Міца О.В. Про обчислення градієнта у задачі синтезу оптичних покриттів. *Теорія оптимальних рішень*. 2005. № 4. С. 127–133.
11. Стецюк П.И., Міца А.В. О задачах оптимизации параметров для многослойных оптических покрытий. *Кибернетика и системный анализ*. 2005. № 4. С. 107–115.
12. Ермолаев И. Умножение матриц: эффективная реализация шаг за шагом. [Електронний ресурс]. Доступно: <https://habr.com/ru/post/359272/>. Дата звернення: Лист. 28, 2020.

Отримано 28.09.2021