

С.І. Літницький
НАБЛИЖЕНЕ ВИЗНАЧЕННЯ ОСІ КОНОІДА, ПРЕДСТАВЛЕНОГО ДИСКРЕТНИМ
КАРКАСОМ ТВІРНИХ

Розроблено алгоритм наближеного визначення осі коноїда, представленого дискретним каркасом твірних. В якості апроксимуючої осі коноїда шукали таку пряму, для якої відстань до заданих твірних є мінімальною. Проаналізована робота алгоритму на прикладі десяти різних коноїдів загального вигляду, представлених дискретним каркасом твірних.

Ключові слова: вісь коноїда, дискретний каркас, твірна.

Рис. 5. Літ. 1.

С.И. Литницкий
ПРИБЛИЖЕННОЕ ОПРЕДЕЛЕНИЕ ОСИ КОНОИДА, ПРЕДСТАВЛЕННОГО
ДИСКРЕТНЫМ КАРКАСОМ ОБРАЗУЮЩИХ

Разработан алгоритм приближенного определения оси коноида, представленного дискретным каркасом образующих. В качестве аппроксимирующей оси коноида искали такую прямую, для которой расстояние к заданным образующим является минимальным. Проанализирована работа алгоритма на примере десяти разных коноидов общего вида, представленных дискретным каркасом образующих.

Ключевые слова: ось коноида, дискретный каркас, образующая.

S.I. Litnitskiy
AXIS APPROXIMATE DEFINITION OF CONOID DESCRIBED BY THE SET OF STRAIGHT
LINES

The algorithm of axis approximate definition of conoid described by the set of straight lines is made. The approximate conoid axis is a lane. Conoid is created by straight lines. The axis area possible location was founded. After that the approximate conoid axis with the minimal length to straight lines was defined. The algorithm accuracy was analyzed by calculation length from approximate conoid axis to straight lines, angle and length between approximate axis and real one. For algorithm accuracy analyzing were used ten different conoids.

Keywords: conoid axis, discretely carcass of straight lines.

Вступ. На коноїді, представленому дискретним каркасом твірних, можна побудувати безліч плоских і просторових дискретно представлених напрямних і, виконавши їхню сплайн-інтерполяцію, отримати апроксимуючі континуально задані напрямні. Загущуючи в точках напрямної твірні, які будуть паралельні площині паралелізму коноїда, отримаємо лінійчасту дискретно представлену твірною поверхню. Ці твірні можуть і не проходити через пряму направляючу (вісь коноїда [1]), тому поверхня більше не буде коноїдом. Якщо ж з певним наближенням вдасться визначити вісь коноїда, то вона разом з криволінійною напрямною дозволить отримати апроксимуючий континуально заданий коноїд (рис. 1).

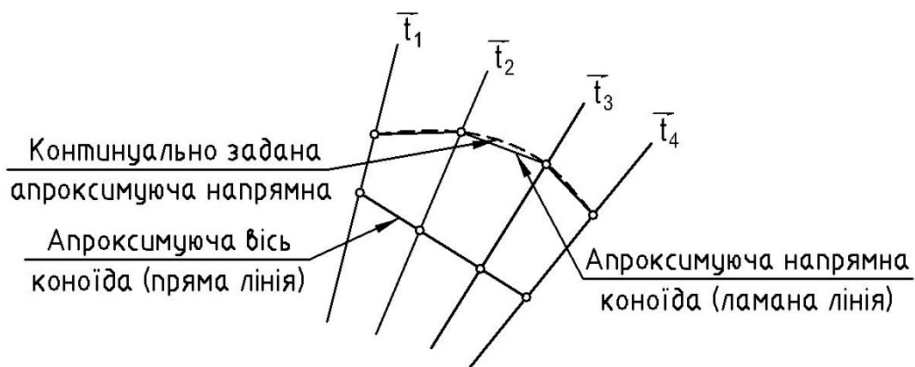


Рис. 1. До побудови апроксимуючого коноїда

Аналіз останніх досліджень. Наскільки відомо автору з літературних джерел, досі це питання в такому аспекті ніким не розглядалося.

Формулювання цілей статті. В роботі поставлено мету – розробити алгоритм наближеного визначення осі коноїда, представленого дискретним каркасом твірних.

Основна частина. Для наближеного визначення осі використовувалася така властивість коноїдів: усі твірні поверхні перетинають її вісь [1]. Тому в даному алгоритмі будемо шукати таку пряму, для якої сума відстаней до твірних буде мінімальною.

Нехай заданий впорядкований каркас твірних коноїда. Тоді алгоритм наближеного визначення осі коноїда складається з таких дій.

1. Перевіряємо, чи можна вважати задані твірні такими, що належать прямому коноїду. Для цього будемо спільні перпендикуляри між сусідніми твірними. Якщо усі перпендикуляри утворюють одну пряму, то досліджувана поверхня є прямим коноїдом, для якого ця пряма і є його віссю. Якщо описана вище умова не виконується, то переходимо до наступних пунктів алгоритму.

2. Шукаємо область розташування осі коноїда. Для цього призначаємо у просторі довільну початкову точку. Проводимо через неї площину, перпендикулярну до площини паралелізму. Визначаємо координати точок, в яких твірні перетнуть площину. З'єднуємо суміжні точки прямими. Обчислюємо суму кутів, які утворюють суміжні прямі (рис. 2). Після цього площина, побудована в довільній початковій точці, переміщується паралельно сама собі в обох напрямках з певним кроком (рис. 3). Крок задається залежно від змістовної суті поставленої задачі. В кожному випадку розташування площин визначається сума вище згаданих кутів. Якщо побудувати графік залежності значення суми цих кутів від накопиченої відстані між площинами, то, як показали обчислювальні експерименти, на ньому буде один або три локальних екстремуми, тобто значення суми кутів після монотонного спадання почне зростати або після зростання – спадати. Аналогічні дії були виконані для континуально заданих коноїдів, з відомими рівняннями. При цьому було виявлено, що локальні екстремуми, виникають у тому випадку, коли площина наближається до точок, в яких вісь коноїда перетинає задані твірні. Тому процес паралельного переміщення площин припиняється тоді, коли отримані координати одного з трьох локальних екстремумів. Фіксуємо координати площини, які відповідають цьому екстремуму. Ця площина перетинає вісь коноїда або знаходиться неподалік від неї. У випадку трьох локальних екстремумів, перший і третій екстремум є відповідно початком і кінцем області, в якій розташована вісь коноїда, тому найближче до осі коноїда знаходиться площина, яка відповідає другому за рахунком екстремуму (рис. 4, а).

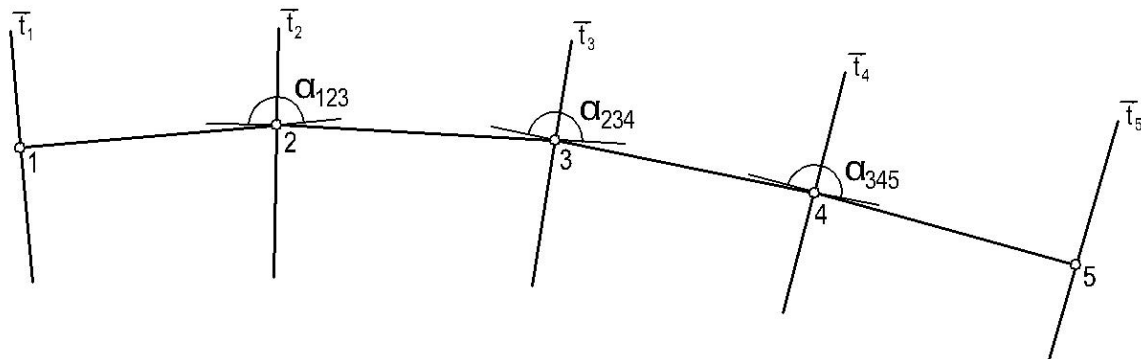


Рис. 2. До обчислення суми кутів, які утворюють сусідні відрізки

На рис. 4, а показаний приклад графіку з трьома локальними екстремумами (отриманий для поверхні, яка описується параметричними рівняннями $x = v - \frac{v^3}{3}$,

$y = 0,2\left(v - \frac{v^3}{3}\right) + u\left(v^2 - 0,2\left(v - \frac{v^3}{3}\right)\right)$, $z = 0,2\left(v - \frac{v^3}{3}\right) + u\left(\frac{3}{2}v + 1 - 0,2\left(v - \frac{v^3}{3}\right)\right)$), а на рис. 3, б – з

одним локальним екстремумом (отриманий для поверхні, яка описується параметричними рівняннями

$$x = v - \frac{v^3}{3}, \quad y = 0,7\left(v - \frac{v^3}{3}\right) + u\left(v^2 - 0,7\left(v - \frac{v^3}{3}\right)\right),$$

$$z = 0,3\left(v - \frac{v^3}{3}\right) + u\left(\frac{3}{2}v + 1 - 0,3\left(v - \frac{v^3}{3}\right)\right)).$$

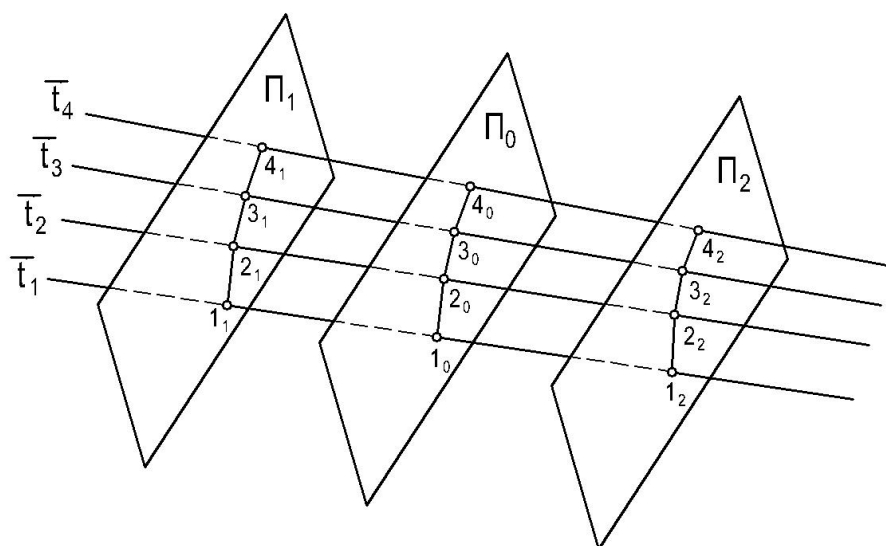


Рис. 3. До паралельного перенесення площини

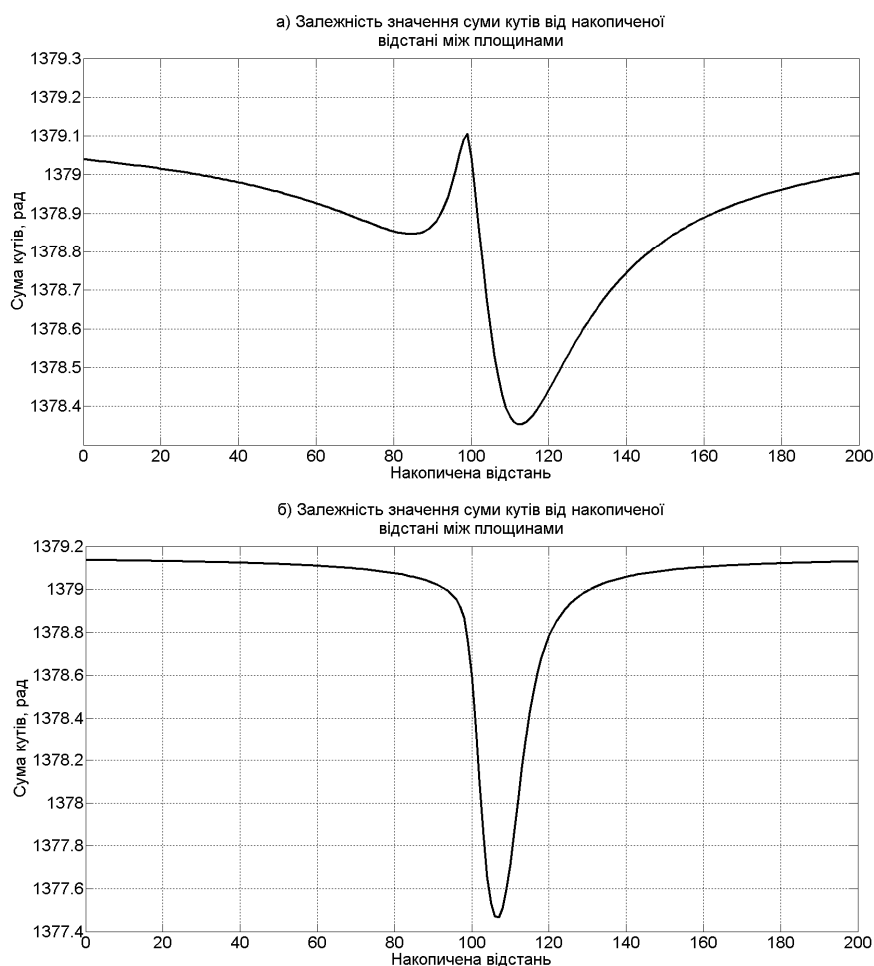


Рис. 4. Графіки залежності значення суми кутів від накопиченої відстані між площинами: а – з трьома локальними екстремумами, б – з одним

3. Визначаємо перше наближення координат осі коноїда. Алгоритм визначення цієї прямої складається з таких дій.

3.1. Визначаємо координати точок, в яких зафіксована площина перетинає твірні. З'єднуємо дві крайні точки прямою. Обчислюємо функцію $f_1(1)$, яка є сумою відстаней цієї прямої від твірних.

$$f_1 = \sum \frac{\left| (x_i - a_1) \cdot \begin{vmatrix} b_2 & b_3 \\ n_{yi} & n_{zi} \end{vmatrix} + (y_i - a_2) \cdot \begin{vmatrix} b_3 & b_1 \\ n_{zi} & n_{xi} \end{vmatrix} + (z_i - a_3) \cdot \begin{vmatrix} b_1 & b_2 \\ n_{xi} & n_{yi} \end{vmatrix} \right|}{\sqrt{\begin{vmatrix} b_2 & b_3 \\ n_{yi} & n_{zi} \end{vmatrix}^2 + \begin{vmatrix} b_3 & b_1 \\ n_{zi} & n_{xi} \end{vmatrix}^2 + \begin{vmatrix} b_1 & b_2 \\ n_{xi} & n_{yi} \end{vmatrix}^2}}, \quad i = \overline{1, k}, \quad (1)$$

де a_1, a_2, a_3 – координати точки, інцидентної осі коноїда;

b_1, b_2, b_3 – проекції вектора осі коноїда;

x, y, z, n_x, n_y, n_z – координати точки, інцидентної-їй твірній, та координати проекцій вектора i -ої твірної;

k – кількість твірних.

Крайня точка розділяє свою твірну на дві частини і, відповідно, задає на твірній два напрями. В одному з напрямів з певним кроком переміщаємо цю точку вздовж твірної і обчислюємо функцію f_1 (рис. 5). Крок, з яким переміщається точка, задається залежно від поставлених задач. Цю операцію повторюємо доти, доки зменшується значення функції f_1 . Фіксуємо її мінімальне значення і координати прямої, які їй відповідають. Аналогічні дії виконуємо, переміщаючи точку у протилежному напрямі. З двох зафіксованих значень функції f_1 вибираємо менше. Надалі точку вздовж крайньої твірної будемо переміщати в напрямі, який відповідає цьому меншому значенню.

3.2. Паралельно самій собі переміщаємо зафіксовану площину в одному з напрямків і виконуємо дії, аналогічні діям підпункту 3.1, але крайню точку переміщаємо вздовж своєї твірної лише у вибраному у попередньому підпункті напрямку. Порівнюємо отримане мінімальне значення функції f_1 з мінімальним значенням, отриманим у попередньому підпункті. Якщо воно менше, то надалі площину будемо паралельно переміщати у вибраному в цьому підпункті напрямку. Якщо ж більше, то – у протилежному напрямку.

3.3. Визначаємо перше наближення координат осі коноїда, обчислюючи функцію f_1 . При цьому паралельно переміщаємо з певним кроком зафіксовану площину у напрямку, вибраному у підпункті 3.2, і переміщаємо крайню точку вздовж своєї твірної у напрямку, вибраному у підпункті 3.1. Крок, з яким переміщаються площина і точка відповідно, задаються залежно від змістовної суті поставленої задачі. Цю операцію повторюємо доти, доки зменшується значення функції f_1 . Фіксуємо координати прямої, які відповідають остаточному мініимальному значенню функції f_1 . Призначаємо ці координати першим наближенням осі коноїда.

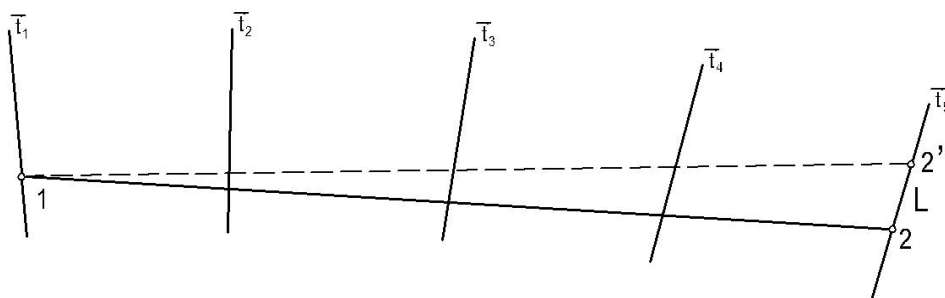


Рис. 5. Переміщення точки вздовж крайньої твірної

4. Визначаємо остаточні координати осі коноїда. Мінімізуємо функцію $f_2(2)$, яка є сумою відстаней осі коноїда від твірних (в якості початкового наближення до осі коноїда, призначаємо координати, отримані в підпункті 3.3)

$$f_2(a, b) = \sum \frac{\left| (x_i - a_{x1}) \cdot \begin{vmatrix} b_2 & b_3 \\ n_{yi} & n_{zi} \end{vmatrix} + (y_i - a_{y1}) \cdot \begin{vmatrix} b_3 & b_1 \\ n_{zi} & n_{xi} \end{vmatrix} + (z_i - a_{z1}) \cdot \begin{vmatrix} b_1 & b_2 \\ n_{xi} & n_{yi} \end{vmatrix} \right|}{\sqrt{\begin{vmatrix} b_2 & b_3 \\ n_{yi} & n_{zi} \end{vmatrix}^2 + \begin{vmatrix} b_3 & b_1 \\ n_{zi} & n_{xi} \end{vmatrix}^2 + \begin{vmatrix} b_1 & b_2 \\ n_{xi} & n_{yi} \end{vmatrix}^2}}, i = \overline{1; k}, \quad (2)$$

де x, y, z, n_x, n_y, n_z – координати точки, інцидентної-ї твірній, та координати проекцій вектора i -ої твірної;

$$a_{x1}(v_1) = a_{x01} + \left(n_{x1} / \sqrt{n_{x1}^2 + n_{y1}^2 + n_{z1}^2} \right) \cdot v_1, a_{y1}(v_1) = a_{y01} + \left(n_{y1} / \sqrt{n_{x1}^2 + n_{y1}^2 + n_{z1}^2} \right) \cdot v_1,$$

$$a_{z1}(v_1) = a_{z01} + \left(n_{z1} / \sqrt{n_{x1}^2 + n_{y1}^2 + n_{z1}^2} \right) \cdot v_1 \text{ – координати першої точки, інцидентної осі коноїда;}$$

$$a_{x2}(v_2) = a_{x02} + \left(n_{xk} / \sqrt{n_{xk}^2 + n_{yk}^2 + n_{zk}^2} \right) \cdot v_2, a_{y2}(v_2) = a_{y02} + \left(n_{yk} / \sqrt{n_{xk}^2 + n_{yk}^2 + n_{zk}^2} \right) \cdot v_2,$$

$$a_{z2}(v_2) = a_{z02} + \left(n_{zk} / \sqrt{n_{xk}^2 + n_{yk}^2 + n_{zk}^2} \right) \cdot v_2 \text{ – координати другої точки, інцидентної осі коноїда;}$$

$a_{x01}, a_{y01}, a_{z01}, a_{x02}, a_{y02}, a_{z02}$ – початкове наближення координат точок, інцидентних осі коноїда;

v_1, v_2 – відстань між початковими і новими координатами точок, інцидентних осі коноїда;

$b_1 = a_{x1} - a_{x2}, b_2 = a_{y1} - a_{y2}, b_3 = a_{z1} - a_{z2}$ – проекції вектора осі коноїда;

k – кількість твірних.

Точність розробленого алгоритму була перевірена на десяти різних коноїдах. Спершу на площині $z = 0$ задавалась сітка з трикутними комірками. Після цього в точках сітки обчислювалася апліката континуально заданої поверхні, тобто формувалася дискретно представлена поверхня. Після цього в кожній точці відсіка задавалась твірна, тобто формувався дискретний каркас з 441-ої впорядкованої твірної. Якщо за одиницю довжини прийняти 1 м, то розміри сторін сітки складали 3×3 м, а крок точок дорівнював 0,2 м. Оскільки рівняння поверхонь є відомими, то була можливість порівняти дійсні координати осей коноїда з наближеними.

Для оцінки точності наближеного визначення осі коноїда обчислювалася відстань і кутове відхилення між апроксимуючою та дійсною осями. Також обчислювалася сума відстаней апроксимуючої осі коноїда від твірних цієї поверхні.

Для перевірки роботи алгоритму вибиралися непрямі коноїди, для яких площина паралелізму була перпендикулярна площині $z = 0$. Рівняння коноїдів задавалися таким чином, щоб осі коноїдів проходили під різними кутами до площини паралелізму.

Для коноїда, описаного параметричними рівняннями $x = v - \frac{v^3}{3},$

$$y = 0,2 \left(v - \frac{v^3}{3} \right) + u \left(v^2 - 0,2 \left(v - \frac{v^3}{3} \right) \right), z = 0,2 \left(v - \frac{v^3}{3} \right) + u \left(\frac{3}{2} v + 1 - 0,2 \left(v - \frac{v^3}{3} \right) \right),$$

сума відстаней від твірних до апроксимуючої осі коноїда дорівнювала $2,9452 \cdot 10^{-6}$ м. Відстань між апроксимуючою і дійсною осями склала $1,9051 \cdot 10^{-7}$ м, а кутове відхилення – $6,2603 \cdot 10^{-7}$ рад.

Висновки. В статті розроблено алгоритм наближеного визначення осі коноїда, представленого дискретним каркасом твірних. Проаналізована точність розробленого алгоритму на прикладі різних коноїдів. Критерії відхилення, для досліджуваних поверхонь, дорівнювали близькому до комп'ютерного нуля числу. Надалі планується розробити алгоритми наближеного визначення визначників інших типів дискретно представлених поверхонь.

1. Кривошапко С.Н. Аналитические поверхности: материалы по геометрии 500 поверхностей и информация к расчету на прочность тонких оболочек / С. Н. Кривошапко, В. Н. Иванов, С. М. Халаби. – М.: Наука, 2006. – 544 с.

Стаття надійшла 14.04.2015.