

КІБЕРНЕТИКА та КОМП'ЮТЕРНІ ТЕХНОЛОГІЇ

Розглянуто задачу віртуального базування, яка є характерною для випадку числового програмного керування (ЧПК) оброблення при виготовленні деталей з заготовок близької геометричної форми. Показано, що для таких випадків замість закріплення заготовки в теоретично заданому положенні ефективною є корекція ЧПК програми під її поточне положення. Розглянуто існуючі алгоритми розв'язання задачі віртуального базування, яка зводиться до встановлення параметрів розміщення САД моделі деталі всередині хмари точок, отриманих скануванням поверхні заготовки. Показано, що їх основним недоліком є використання критеріїв не чутливих до взаємного положення САД моделі деталі та сканованих точок. Як альтернативу запропоновано використовувати алгоритм, що базується на використанні методу рНі-функцій, який повністю усуває вказаний недолік. Описано стратегію розв'язання задачі віртуального базування на базі запропонованого алгоритму та наведено приклади розв'язання кількох тестових задач, які підтверджують його ефективність.

Ключові слова: багатокутна область, метод рНі-функцій, віртуальне базування, ЧПК оброблення.

УДК 621.7/.9:004.946

DOI:10.34229/2707-451X.22.4.3

С.І. ПЛАНКОВСЬКИЙ, Є.В. ЦЕГЕЛЬНИК, О.В. ПАНКРАТОВ,
Т.Є. РОМАНОВА, С.В. МАКСИМОВ, В.В. КОМБАРОВ

МАТЕМАТИЧНА МОДЕЛЬ ТА МЕТОД РОЗВ'ЯЗАННЯ ЗАДАЧІ ВІРТУАЛЬНОГО БАЗУВАННЯ

Вступ. Останнім часом все більшого поширення набувають технологічні процеси, в яких готові деталі одержують із заготовок, виготовлених з мінімальними припусками на оброблення [1] за допомогою точного лиття, штампування або використання адитивних процесів (3D друку) [2]. Використання таких заготовок дозволяє суттєво знизити витрату матеріалу та тривалість операцій розмірного оброблення. Однак при цьому виникає проблема базування деталей особливо у випадках, коли вони мають великі габарити та вагу або малу жорсткість.

Традиційно при розмірному обробленні базування проводиться за допомогою спеціальних пристосувань, які не тільки закріплюють заготовку, а й забезпечують певну її орієнтацію у системі координат верстата. Однак у багатьох випадках, особливо при виготовленні великогабаритних деталей складної форми, проектування та виготовлення таких пристосувань потребує більшого часу та коштів, ніж виготовлення самої деталі, а сам процес базування вимагає істотних витрат часу, а для деталей великої ваги пов'язаний із суттєвими труднощами. Альтернатива – це застосування адаптивного розмірного оброблення [3] з віртуальним базуванням. У цьому випадку заготовка встановлюється на столі верстата за допомогою спрощених пристосувань, а її орієнтація в системі координат верстата і форма визначається за допомогою вбудованих вимірювальних систем.

Після цього вирішується задача віртуального розміщення САД (Computer Aided Design) моделі деталі всередині заготовки, зазвичай виходячи з умови забезпечення рівномірного припуску на оброблення, визначається розташування системи координат деталі щодо системи координат верстата і перераховується програма числового програмного керування (ЧПК) обробленням, яка зазвичай складається у системі координат деталі.

© С.І. Планковський, Є.В. Цегельник,
О.В. Панкратов, Т.Є. Романова,
С.В. Максимов, В.В. Комбаров, 2022

Таким чином, ідея адаптивного оброблення з віртуальним базуванням полягає у тому, що замість закріплення заготовки в теоретично заданому положенні використовується корекція ЧПК програми під її поточне положення. Такий підхід успішно використовується в аерокосмічній промисловості при обробленні великогабаритних тонкостінних деталей [4, 5], виробництві та відновленні лопаток газотурбінних двигунів [6 – 8], сопел ракетних рідинних двигунів [9, 10], а також при виробництві великогабаритних деталей складної форми – гребних гвинтів, лопатей гідравлічних та парових турбін тощо.

Визначення положення моделі CAD деталі всередині заготовки – найважливіший етап адаптивного оброблення. На практиці для сканування заготовки частіше всього застосовуються контактні або безконтактні (лазерні) системи. В результаті вимірювання поверхні заготовки задається область точок з кроком, що визначається частотою сканування. Мета розв'язання завдання є розміщення всередині заготовки моделі CAD деталі (рис. 1) таким чином, щоб:

- модель CAD ніде не перетинала поверхню, яку натягнуто на хмару точок;
- відстань від поверхні CAD моделі, до поверхні, яку натягнуто на хмару точок, була максимально рівномірною.

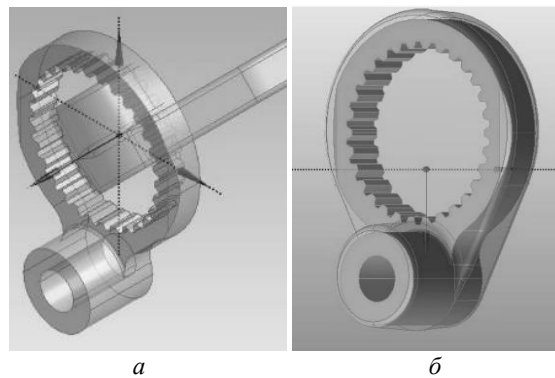


РИС. 1. Схема віртуального базування: *а* – початкове розташування CAD моделі деталі та заготовки; *б* – заготовка та CAD модель деталі після віртуального базування [11]

Розв'язок задачі розміщення практично у всіх дослідженнях, присвячених питанням віртуального базування, ведеться в два етапи. На першому етапі виконується попереднє розміщення CAD моделі всередині заготовки. Стартове положення CAD моделі в переважній кількості робіт задається з умови суміщення центрів ваги тонких оболонок, що співпадають з поверхнями заготовки і CAD моделі деталі [12 – 14]. Варіант стартового розміщення, запропонований у роботі [15] у випадку, коли припуск на оброблення є приблизно однаковим по всій поверхні деталі, є більш точним. В цьому випадку пропонується суміщувати не тільки центри ваги оболонок, але й їхні головні центральні осі інерції.

На другому етапі розв'язання задачі віртуального базування здійснюється точне розміщення шляхом трансляції та повороту CAD моделі відносно її початкового положення. Для цього в переважній більшості робіт використовується ітераційний алгоритм найближчих точок ICP (Iterative Closest Point), запропонований в роботі [16]. Критерієм оптимального взаємного розміщення геометричних об'єктів у цьому методі є мінімізація суми квадратів відстаней між їхніми елементами (для задачі віртуального базування відстаней між точками зі сканованої хмари та кривими і поверхнями CAD моделі).

Відомі модифікації алгоритму ICP, які використовують глобальні контрольні точки, розраховані як середні значення за координатами [17], або особливі точки, які отримано на основі зобра-

жень кутів повороту (Bearing Angle Image) [18], застосування яких при обчисленнях не вимагає завдання початкових наближень для матриці повороту і вектора зміщення, тим самим дозволяючи уникнути попадання у локальний мінімум, потребує меншої кількості ітерацій на суміщення хмар точок і виконується швидше у порівнянні з базовим алгоритмом.

Значимо, що обсяг обчислень за алгоритмом ICP залежить від часу визначення найближчих точок і часу знаходження параметрів перетворення, тому в гіршому випадку вона становить $O(N_s \times N_t)$ для N точок з хмар S і T відповідно, тобто складає $O(N^2)$ при однаковій кількості точок у суміщуваних хмарах.

Така особливість викликала появу модифікацій алгоритму, пов'язаних зі зменшенням кількості точок для суміщення з відповідною хмарию точок чи поверхнею. Найчастіше використовуваним підходом є впорядкування точок цільової хмари за допомогою геометричної інтерпретації бінарного дерева [19]. Застосування такого підходу дозволяє за рахунок сегментації точок за індексами суттєво скоротити кількість варіантів при пошуку найближчої точки з трансформованої хмари так, що кількість варіантів, що підлягає розгляду становить $O(N_s \times \log N_t)$.

Крім алгоритмів ICP, було запропоновано кілька інших методів. У роботі [20] досліджувався заснований на порівняннях отриманих поверхонь з номінальними даними або для прямого порівняння різних сканованих поверхонь з використанням алгоритму, заснованому на розширеній гаусовській кривизні, і метод порівняння характеристик, заснованих на сукупній нормальній орієнтації. У роботі [21] задача суміщення тривимірних об'єктів розглядалась з точки зору розв'язків системи нелінійних поліноміальних рівнянь, яку може бути розв'язано за допомогою алгоритму, де як критерій оцінювання використовувалась гаусівська та середня кривизна. Тим не менш на сьогоднішній день алгоритм ICP – це базовий стандарт для задач суміщення геометричних об'єктів.

Недоліком ICP методів стосовно до задачі віртуального базування є застосування критерію мінімізації середнього значення квадрату відстані між хмарию точок вимірювання та проектною поверхнею. При такому критерії алгоритм не чутливий до того, де саме щодо CAD поверхні знаходяться точки з трансформованої хмари – усередині чи ззовні CAD моделі.

Вочевидь ситуацію не може змінити використання для визначення відстані інших норм, нечутливих до взаємного розташування точок, наприклад, манхеттенської, яка застосовувалась в деяких роботах, присвячених модифікації алгоритму ICP з метою прискорення збіжності [22]. Це саме стосується інших подібних робіт, наприклад, [23, 24] де було введено модифікацію функцію відстані від точки до поверхні і досліджувались властивості комплексної похибки поверхні.

З точки зору задачі віртуального базування може бути перспективним застосування методу множників Лагранжа, використаний у роботі [25] при розгляді задачі базування поверхні вільної форми. Крім того слід виділити роботи [26, 27] в яких запропоновано мінімізувати функціонал $U = \max[\rho_i] + U_P$, де ρ_i – відстань між точкою з хмари, отриманої вимірюванням до поверхні CAD моделі; U_P – штрафна функція.

При відповідному виборі коефіцієнтів штрафної функції вона набуває властивостей, які дозволяють «виштовхувати» точки з хмари назовні області CAD моделі. В роботах [26, 27] продемонстровано ефективність використання такого підходу в задачах віртуального базування деталей складної форми.

Водночас у відомих роботах не сформовано обґрунтований підхід до формування штрафних функцій, які використовуються при записуванні цільової функції. Таким чином, задача віртуального базування об'єктів складної форми на основі сумісного використання CAD моделі і хмари точок, яку отримано при вимірюванні заготовки, як і раніше залишається складним актуальним завданням. Розроблення нових підходів до розв'язання цієї проблеми доцільно розпочати на прик-

ладі двовимірних задач. Окрім зменшення обчислювальних витрат такий підхід може бути безпосередньо застосований для розв'язання задач віртуального базування при контурному фрезеруванні [28].

Як показано в роботах [28, 29] для комп'ютерного моделювання задачі контурного фрезерування деталей на верстатах з ЧПК може успішно використовуватись представлення геометричних моделей у вигляді ламаних ліній. Для цього плоскі геометричні контури заготовки та деталі задаються двовимірними цифровими масивами точок із заданим кроком. Положення заготовки на столі верстата зазвичай визначається за допомогою вбудованих вимірювальних головок які здатні в автоматизованому режимі визначати координати точок контуру заготовки у системі координат верстата з похибкою до 0,001 мм. На рис. 2 показано контури деталі D і заготовки B , які представлені цифровими масивами, що містять координати двох наборів точок $D = \{d_1, d_2, \dots, d_n\}$ і $B = \{b_1, b_2, \dots, b_m\}$. Задачею віртуального базування в цьому випадку є пошук параметрів розміщення, які забезпечують мінімізацію максимального припуску на оброблення (тобто відстані між контурами деталі та заготовки).

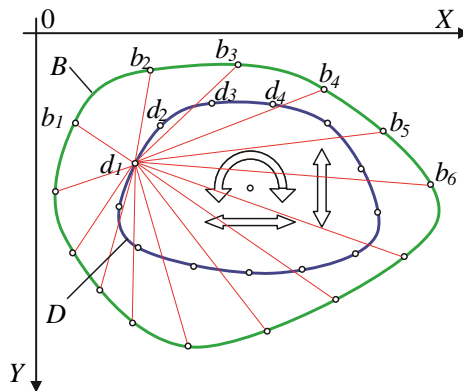


РИС. 2. Приклад віртуального базування при контурному фрезеруванні: D – контур деталі, B – контур заготовки [28]

Значимо, що навіть у такій спрощеній постановці при застосуванні алгоритму, який базується на максимізації відстані Хаусдорфу, у роботі [28] відзначались випадки перетинання контурів деталі і заготовки. Для запобігання таким похибкам віртуальне базування у [28] передбачало остаточне прийняття рішення щодо завершення процедури віртуального базування інженером-технологом. Таким чином, задача віртуального базування при ЧПК обробленні потребує розроблення нових алгоритмів, які б давали змогу здійснювати її без похибок та в автоматичному режимі.

У статті розглядається підхід, який використовує метод ϕ -функцій [30 – 32] для розв'язання задачі віртуального базування.

Постановка задачі.

Багатокутна область. Нехай $B \subset \mathbf{R}^2$ – фіксована обмежена опукла багатокутна область, яку задано вершинами $u_k = (x'_k, y'_k)$, $k = 1, \dots, n$, і $B = \{(x, y) : \varphi_k(x, y) \geq 0, k = 1, \dots, n\}$, де $\varphi_k(x, y) = \alpha_k x - \beta_k y + \gamma_k$, $\alpha_k^2 + \beta_k^2 = 1$, для $\alpha_k = (y'_{k+1} - y'_k) / \sqrt{(x'_{k+1} - x'_k)^2 + (y'_{k+1} - y'_k)^2}$, $\beta_k = (x'_{k+1} - x'_k) / \sqrt{(x'_{k+1} - x'_k)^2 + (y'_{k+1} - y'_k)^2}$, $\gamma_k = -\alpha_k x'_k - \beta_k y'_k$, $k = 1, \dots, n$, за умови $u_{n+1} = u_1$.

Об'єкт. Нехай $D \subset \mathbf{R}^2$ – опуклий багатокутник, заданий вершинами $\tilde{v}_i = (\tilde{x}_i, \tilde{y}_i)$, $i = 1, \dots, m$.

Розміщення об'єкта D у просторі \mathbf{R}^2 визначається змінним вектором його параметрів розміщення (x_d, y_d, θ) . Трансляція D на вектор $v_d = (x_d, y_d) \in \mathbf{R}^2$ і обертання D на кут $\theta \in [0, 2\pi)$ визначається як

$$D(v_d, \theta) = \{t \in \mathbf{R}^2 : t = v_d + M(\theta)\tilde{t}, \forall \tilde{t} \in D(0, 0, 0)\},$$

де $D(0, 0, 0)$ позначає не трансльований і неповернутий багатокутник D і $M(\theta) = \begin{pmatrix} \cos \theta & -\sin \theta \\ \sin \theta & \cos \theta \end{pmatrix}$.

Таким чином кожна точка $\tilde{t} = (x, y) \in D(0, 0, 0)$ у локальній системі координат об'єкта D перетворюється на точку $t = (x, y)$ наступним чином:

$$x = x \cdot \cos \theta - y \cdot \sin \theta + x_d, \quad y = x \cdot \sin \theta + y \cdot \cos \theta + y_d.$$

Задача віртуального базування. Розмістити заданий об'єкт D повністю всередині фіксованої багатокутної області B з максимізацією евклідової відстані між об'єктом D і межею B .

Позначимо евклідову відстань між об'єктом D і межею B як ρ , тобто $\rho = \text{dist}(D, B^*) = \min_{d \in D, b \in B^*} \|d - b\|$, $B^* = \mathbf{R}^2 \setminus B$. Тому задача розміщення спрямована на пошук вектора (x_d, y_d, θ) забезпечивши максимальне значення ρ .

Математична модель. В термінах *phi*-функцій [30 – 32] сформулюємо задачу віртуального базування як задачу нелінійного програмування

$$\max \rho \text{ за умови } (v_d, \theta, \rho) \in W, \tag{1}$$

де

$$W = \{(v_d, \theta, \rho) \in \mathbf{R}^4 : \Phi^{B^*D}(v_d, \theta) \geq \rho\}, \tag{2}$$

$$\rho \geq 0. \tag{3}$$

В моделі (1)–(3) $(v_d, \theta, \rho) = (x_d, y_d, \theta, \rho)$ – вектор змінних, $\Phi^{B^*D}(v_d, \theta)$ – нормалізована *phi*-функція для об'єктів $D(v_d, \theta)$ і B^* визначається у такій формі:

$$\Phi^{B^*D}(v_d, \theta) = \min_{k=1, \dots, n, i=1, \dots, m} \varphi_k(x_{di}, y_{di}), \tag{4}$$

де

$$\varphi_k(x_{di}, y_{di}) = \alpha_k x_{di} - \beta_k y_{di} + \gamma_k,$$

$$x_{di} = x_{di} \cdot \cos \theta - y_{di} \cdot \sin \theta + x_d, \quad y_{di} = x_{di} \cdot \sin \theta + y_{di} \cdot \cos \theta + y_d.$$

Модель (1)–(3) з урахуванням (4) можна переписати в еквівалентній формі

$$\max \rho \tag{5}$$

за умови

$$\alpha_k(x_{di} \cdot \cos \theta - y_{di} \cdot \sin \theta + x_d) - \beta_k(x_{di} \cdot \sin \theta + y_{di} \cdot \cos \theta + y_d) + \gamma_k - \rho \geq 0, \tag{6}$$

для $k=1, \dots, n, i=1, \dots, m$,

$$\rho \geq 0. \quad (7)$$

Тому змінну ρ можна розглядати як $\rho = \min_{i=1, \dots, m, k=1, \dots, n} \varphi_k(x_{di}, y_{di})$.

Загальна стратегія розв'язання. Алгоритм розв'язання задачі *віртуального базування* складається з трьох основних етапів.

Етап 1. Побудова множини допустимих стартових точок задачі (5)–(7).

Етап 2. Пошук локально-оптимального максимуму задачі (5)–(7) за допомогою ПРОРТ для кожної стартової точки, яку знайдено на етапі 1.

Етап 3. Вибір найкращого локального оптимального розв'язку з знайдених на етапі 2.

Розглянемо детально евристичний алгоритм для пошуку допустимої точки задачі (5)–(7).

Крок 1. Генеруємо точку $v_d^0 = (x_d^0, y_d^0) \in B$ випадковим чином.

Крок 2. Генеруємо кут повороту $\theta^0 \in [0, 2\pi]$ випадковим чином.

Крок 3. Розв'язуємо таку задачу нелінійного програмування, починаючи з точки (x_d^0, y_d^0, θ^0) :

$$\max \mu \quad (8)$$

за умови

$$\alpha_k(\mu(x_{di} \cdot \cos \theta + y_{di} \cdot \sin \theta) + x_d) - \beta_k(\mu(-x_{di} \cdot \sin \theta + y_{di} \cdot \cos \theta) + y_d) + \gamma_k \geq 0, \quad (9)$$

для $k=1, \dots, n, i=1, \dots, m$,

$$0 \leq \mu \leq 1. \quad (10)$$

У моделі (8)–(10) (v_d, θ, μ) – вектор змінних.

Крок 4. Якщо $\mu^* < 1$ є розв'язком задачі (8)–(10), тоді переходимо до кроку 1, інакше ($\mu^* = 1$)

беремо точку (v_d^*, θ^*) як допустиму стартову точку задачі (5)–(7).

Обчислювальні експерименти. У даному розділі розглядаються результати обчислень, які знайдено за вищенаведеним алгоритмом. Усі числові експерименти проводили на комп'ютері AMD FX(tm)-6100, 3,30 ГГц, мова програмування C++, Windows 10. Для локальної оптимізації у задачі нелінійного програмування використано відкритий код ПРОРТ [33].

Приклад 1.

Область B: $\{(x'_k, y'_k), k=1, \dots, n=10\} = \{(48, 146), (46, 155), (44, 162), (40, 164), (33, 163), (26, 160), (24, 155), (25, 150), (32, 145), (39, 144)\}$.

Об'єкт D: $\{(x_{di}, y_{di}), i=1, \dots, m=10\} = \{(43.2, 131.4), (41.4, 139.5), (39.6, 145.8), (36.0, 147.6), (29.7, 146.7), (23.4, 144.0), (21.6, 139.5), (22.5, 135.0), (28.8, 130.5), (35.1, 129.6)\}$.

Локально-оптимальний розв'язок, знайдений за нашим алгоритмом, $\rho^* = 0.9469$, $(x_d^*, y_d^*, \theta^*) = (3.645778, 15.392939, 0.0)$ отримано за 0.094 с (3 ітерації). Відповідне локально-оптимальне розміщення показано на рис. 3,а.

Приклад 2.

Область B: $\{(x'_k, y'_k), k=1, \dots, n=10\} = \{(48, 146), (46, 155), (44, 162), (40, 164), (33, 163), (26, 160), (24, 155), (25, 150), (32, 145), (39, 144)\}$.

Об'єкт D : $\{(x_{di}, y_{di}), i=1, \dots, m=10\} = \{(40.8, -124.1), (39.1, -131.75), (37.4, -137.7), (34.0, -139.4), (28.05, -138.55), (22.1, -136.0), (20.4, -131.75), (21.25, -127.5), (27.2, -123.25), (33.15, -122.4)\}$.

Локально-оптимальний розв'язок, знайдений за нашим алгоритмом, $\rho^* = 0.429455$, $(x_d^*, y_d^*, \theta^*) = (-60.768149, 62.430262, -4.189957)$ отримано за 0.188 с (7 ітерацій). Відповідне локально-оптимальне розміщення показано на рис. 3,б.

Приклад 3.

Область B : $\{(x'_k, y'_k), k=1, \dots, n=17\} = \{(87, 152), (83, 153), (78, 154), (72, 154), (68, 153), (66, 152), (63, 149), (62, 147), (61.5, 145), (61, 137), (63, 133), (65, 130), (70, 129), (76, 130), (80, 132), (84, 139), (86, 146)\}$

Об'єкт D : $\{(x_{di}, y_{di}), i=1, \dots, m=10\} = \{(48, 146), (46, 155), (44, 162), (40, 164), (33, 163), (26, 160), (24, 155), (25, 150), (32, 145), (39, 144)\}$.

Локально-оптимальний розв'язок, знайдений за нашим алгоритмом, $\rho^* = 0.882078$, $(x_d^*, y_d^*, \theta^*) = (190.297898, 35.869992, -11.498256)$ отримано за 2.265 с (7 ітерацій). Відповідне локально-оптимальне розміщення показано на рис. 3,в.

Приклад 4.

Область B : $\{(x'_k, y'_k), k=1, \dots, n=17\} = \{(87, 152), (83, 153), (78, 154), (72, 154), (68, 153), (66, 152), (63, 149), (62, 147), (61.5, 145), (61, 137), (63, 133), (65, 130), (70, 129), (76, 130), (80, 132), (84, 139), (86, 146)\}$

Об'єкт D : $\{(x_{di}, y_{di}), i=1, \dots, m=10\} = \{(48., -146.), (46., -155.), (44., -162.), (40., -164.), (33., -163.), (26., -160.), (24., -155.), (25., -150.), (32., -145.), (39., -144.)\}$

Локально-оптимальний розв'язок, знайдений за нашим алгоритмом, $\rho^* = 0.661594$, $(x_d^*, y_d^*, \theta^*) = (177.293139, 23.874649, -2.654040)$ отримано за 0.265 с (5 ітерацій). Відповідне локально-оптимальне розміщення показано на рис. 3,г.

Приклад 5.

Область B : $\{(x'_k, y'_k), k=1, \dots, n=10\} = \{(48, 146), (46, 155), (44, 162), (40, 164), (33, 163), (26, 160), (24, 155), (25, 150), (32, 145), (39, 144)\}$.

Об'єкт D : $\{(x_{di}, y_{di}), i=1, \dots, m=17\} = \{(60.9, 106.4), (58.1, 107.1), (54.6, 107.8), (50.4, 107.8), (47.6, 107.1), (46.2, 106.4), (44.1, 104.3), (43.4, 102.9), (43.05, 101.5), (42.7, 95.9), (44.1, 93.1), (45.5, 91.0), (49.0, 90.3), (53.2, 91.0), (56.0, 92.4), (58.8, 97.3), (60.2, 102.2)\}$.

Локально-оптимальний розв'язок, знайдений за нашим алгоритмом, $\rho^* = 1.081789$, $(x_d^*, y_d^*, \theta^*) = (-75.330592, 163.346680, -32.596205)$ отримано за 0.279 с (7 ітерацій). Відповідне локально-оптимальне розміщення показано на рис. 3,д.

Приклад 6.

Область B : $\{(x'_k, y'_k), k=1, \dots, n=10\} = \{(48, 146), (46, 155), (44, 162), (40, 164), (33, 163), (26, 160), (24, 155), (25, 150), (32, 145), (39, 144)\}$.

Об'єкт D : $\{(x_{di}, y_{di}), i=1, \dots, m=17\} = \{(60.9, -106.4), (58.1, -107.1), (54.6, -107.8), (50.4, -107.8), (47.6, -107.1), (46.2, -106.4), (44.1, -104.3), (43.4, -102.9), (43.05, -101.5), (42.7, -95.9), (44.1, -93.1), (45.5, -91.0), (49.0, -90.3), (53.2, -91.0), (56.0, -92.4), (58.8, -97.3), (60.2, -102.2)\}$.

Локальний оптимальний розв'язок, знайдений за нашим алгоритмом, $\rho^* = 0.671111$, $(x_d^*, y_d^*, \theta^*) = (105.414125, 65.179839, 3.323901)$ отримано за 0.094 с (1 ітерація). Відповідне локально-оптимальне розміщення показано на рис. 3, *e*.

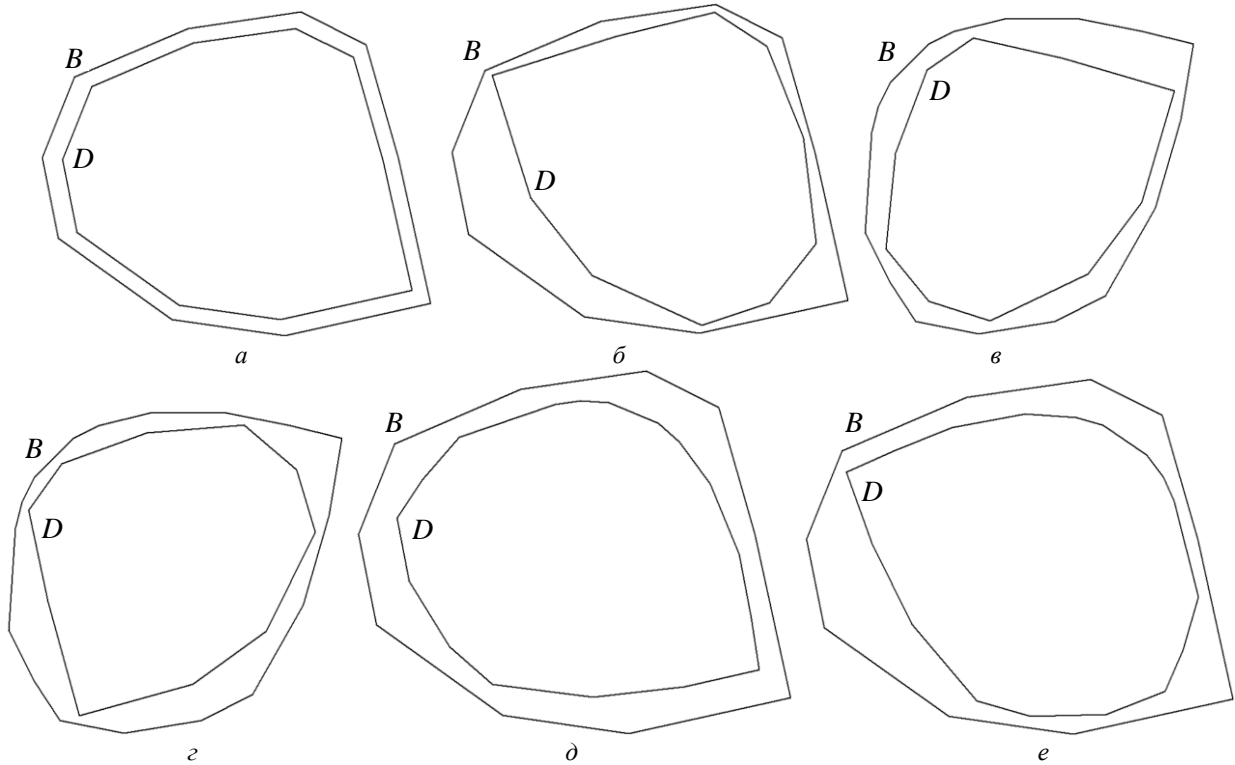


РИС. 3. Локально-оптимальне розміщення: *a* – приклад 1; *б* – приклад 2; *в* – приклад 3; *г* – приклад 4; *д* – приклад 5; *e* – приклад 6

Результати. Розглянуто задачу віртуального базування, яка характерна для випадку ЧПК оброблення при виготовленні деталей з заготовок “близької” геометричної форми. Задача зводиться до встановлення параметрів розміщення CAD моделі деталі всередині хмари точок, які отримано скануванням поверхні заготовки. Показано, що використання існуючих алгоритмів розв’язання задачі віртуального базування, які у переважній кількості базуються на методі ІСР, може призводити до помилкових розв’язків. Причиною цього є використання критеріїв не чутливих до перетинання поверхонь CAD моделі деталі та заготовки.

Запропоновано альтернативний підхід для розв’язання задачі віртуального базування на основі методу *phi*-функцій, який унеможливорює виникнення вказаних помилок. Тестування запропонованого підходу проведено для двовимірного випадку. Він характерний для віртуального базування при контурному фрезеруванні. Для побудови алгоритму віртуального базування розглянуто нову постановку задачі розміщення багатокутних об’єктів у багатокутній області. Побудовано математичну модель як неперервну задачу нелінійного програмування та розроблено евристичний алгоритм. Наведено результати числових експериментів та графічні ілюстрації отриманих результатів, які підтверджують ефективність використання запропонованого алгоритму.

Висновки. Отримані результати показують, що використання методу *phi*-функцій – перспективні для розв’язання задач віртуального базування. Його основна перевага над існуючими алгори-

тмами – унеможливлення виникнення помилкових розв'язків з перетинанням поверхонь заготовки та деталі. Такі похибки при проведенні базування без участі оператора неминуче призведуть до виникнення виробничого браку.

Розроблено алгоритм розв'язання задачі віртуального базування у двовимірній постановці для випадку, коли деталь та заготовка – опуклі багатокутники. Для розглянутих тестових задач час розв'язання не перевищував 2.5 с, що цілком відповідає вимогам подальшого промислового використання запропонованого методу віртуального базування.

Розвиток цієї роботи буде направлено на поширення запропонованого методу на випадки двовимірних задач в яких САД модель деталі має довільну форму і утворена шляхом булевих операцій над геометричними примітивами. В подальшому планується розглянути задачі віртуального базування тривимірних тіл.

Буде досліджено ефективність розроблених алгоритмів при розв'язанні конкретних практичних задач у виробничих умовах. Для цього планується дослідити ефективність взаємодії створеного методу віртуального базування з інструментальними засобами позиціонування сучасного обладнання з ЧПК – вимірювальними головками, лазерними та оптичними сканерами.

Подяки. Робота частково підтримана грантом № 02.2020/167 Національного фонду досліджень України та НДР 0121U109639 Міністерства науки і освіти України, the Volkswagen Foundation under grant No. 9C086.

Список літератури

1. Marini D., Cunningham D., Corney J.R. Near net shape manufacturing of metal: a review of approaches and their evolutions. *Proceedings of the Institution of Mechanical Engineers, Part B: Journal of Engineering Manufacture*. 2018. **232** (4). P. 650–669. <https://doi.org/10.1177/0954405417708220>
2. Riveiro A., del Val J., Comesaña R., et al. Laser additive manufacturing processes for near net shape components. In: Gupta K. (eds.) *Near Net Shape Manufacturing Processes*. MFMT. Springer, Cham, 2019. P. 105–141. https://doi.org/10.1007/978-3-030-10579-2_5
3. Kombarov V., Sorokin V., Tsegelnyk Y., et al. Numerical control of machining parts from aluminum alloys with sticking minimization. *International Journal of Mechatronics and Applied Mechanics*. 2021. **9**. P. 209–216. <https://doi.org/10.17683/IJOMAM/ISSUE9.30>
4. Bi Q., Huang N., Zhang S., et al. Adaptive machining for curved contour on deformed large skin based on on-machine measurement and isometric mapping. *International Journal of Machine Tools and Manufacture*. 2019. **136**. P. 34–44. <https://doi.org/10.1016/j.ijmachtools.2018.09.001>
5. Del Sol I., Rivero A., López de Lacalle L.N., Gamez A.J. Thin-wall machining of light alloys: A review of models and industrial approaches. *Materials*. 2019. **12** (12). Art. No. 2012. <https://doi.org/10.3390/ma12122012>
6. Wu D., Wang H., Peng J., et al. Analysis of machining deformation for adaptive CNC machining technology of near-net-shaped jet engine blade. *The International Journal of Advanced Manufacturing Technology*. 2019. **104** (9). P. 3383–3400. <https://doi.org/10.1007/s00170-019-03898-6>
7. Denkena B., Boess V., Nesper D., et al. Engine blade regeneration: A literature review on common technologies in terms of machining. *The International Journal of Advanced Manufacturing Technology*. 2015. **81** (5). P. 917–924. <https://doi.org/10.1007/s00170-015-7256-2>
8. Zhao Y., Sun J., Li J., et al. Remanufacturing of compressor blade by laser direct metal deposition and subsequent multi-axis milling. In: *International Congress on Applications of Lasers & Electro-Optics*. Laser Institute of America, 2015. **2015**. P. 748–753. <https://doi.org/10.2351/1.5063225>
9. Gradl P.R. Rapid fabrication techniques for liquid rocket channel wall nozzles. In: *52nd AIAA/SAE/ASEE Joint Propulsion Conference*. AIAA, 2016. No. 4771. <https://doi.org/10.2514/6.2016-4771>
10. Коваленко А.Н., Кирсанов Д.В., Миросиди Н.А., и др. Разработка новой технологии изготовления сопловых блоков без использования пайки. *Космическая техника. Ракетное вооружение*. 2018. **2**. С. 68–75. <https://doi.org/10.33136/stma2018.02.068>

11. Планковський С.І., Цегельник С.В., Минтюк В.Б., та ін. Метод віртуального базування деталей з формою, наближеною до форми заготовок. *Авіаційно-космічна техніка і технологія*. 2020. **4** (164). С. 74–82. <https://doi.org/10.32620/aktt.2020.4.09>
12. Shen B., Huang G., Mak K., Wang X. A best-fitting algorithm for optimal location of large-scale blanks with free-form surfaces. *Journal of Materials Processing Technology*. 2003. **139**. P. 310–314. [https://doi.org/10.1016/S0924-0136\(03\)00241-3](https://doi.org/10.1016/S0924-0136(03)00241-3)
13. Sun Y., Wang X., Guo D., Liu J. Machining localization and quality evaluation of parts with sculptured surfaces using SQP method. *The International Journal of Advanced Manufacturing Technology*. 2009. **42** (11). P. 1131–1139. <https://doi.org/10.1007/s00170-008-1673-4>
14. Mehrad V., Xue D., Gu P. Robust localization to align measured points on the manufactured surface with design surface for freeform surface inspection. *Computer-Aided Design*. 2014. **53**. P. 90–103. <https://doi.org/10.1016/j.cad.2014.04.003>
15. Plankovskyy S., Nikolaev A., Shypul O., et al. Balance layout problem with the optimized distances between objects. In: Vasant P., et al. (eds.) *Data Analysis and Optimization for Engineering and Computing Problems*. EAISICC. Springer, Cham, 2020. P. 85–93. https://doi.org/10.1007/978-3-030-48149-0_7
16. Besl P.J., McKay N.D. A method for registration of 3-D shapes. *IEEE Transactions on Pattern Analysis and Machine Intelligence*. 1992. **14** (2). P. 239–256. <https://doi.org/10.1109/34.121791>
17. Du S., Xu Y., Wan T., et al. Robust iterative closest point algorithm based on global reference point for rotation invariant registration. *PLoS One*. 2017. **12** (11). Art. No. e0188039. <https://doi.org/10.1371/journal.pone.0188039>
18. Lin C., Tai Y., Lee J., Chen Y. A novel point cloud registration using 2D image features. *EURASIP Journal on Advances in Signal Processing*. 2017. **2017**. Art. No. 5. <https://doi.org/10.1186/s13634-016-0435-y>
19. Bentley J.L. Multidimensional binary search trees used for associative searching. *Communications of the ACM*. 1975. **18** (9). P. 509–517. <https://doi.org/10.1145/361002.361007>
20. Orazi L., Tani G. Geometrical inspection of designed and acquired surfaces. *The International Journal of Advanced Manufacturing Technology*. 2007. **34** (1). P. 149–155. <https://doi.org/10.1007/s00170-006-0587-2>
21. Ko K.H., Maekawa T., Patrikalakis N.M. An algorithm for optimal free-form object matching. *Computer-Aided Design*. 2003. **35** (10). P. 913–923. [https://doi.org/10.1016/S0010-4485\(02\)00205-1](https://doi.org/10.1016/S0010-4485(02)00205-1)
22. Mora H., Mora-Pascual J.M., Garcia-Garcia A., Martinez-Gonzalez P. Computational analysis of distance operators for the iterative closest point algorithm. *PloS One*. 2016. **11** (10). Art. No. e0164694. <https://doi.org/10.1371/journal.pone.0164694>
23. Zhu L., Xiong Z., Ding H., Xiong Y. A distance function based approach for localization and profile error evaluation of complex surface. *Journal of Manufacturing Science and Engineering*. 2004. **126** (3). P. 542–554. <https://doi.org/10.1115/1.1763186>
24. Zhu L., Zhang X., Ding H., Xiong Y. Geometry of signed point-to-surface distance function and its application to surface approximation. *Journal of Computing and Information Science in Engineering*. 2010. **10** (4). Art. No. 041003. <https://doi.org/10.1115/1.3510588>
25. Sun Y., Xu J., Guo D., Jia Z. A unified localization approach for machining allowance optimization of complex curved surfaces. *Precision Engineering*. 2009. **33** (4). P. 516–523. <https://doi.org/10.1016/j.precisioneng.2009.02.003>
26. Chatelain J.F., Fortin C. A balancing technique for optimal blank part machining. *Precision Engineering*. 2001. **25** (1). P. 13–23. [https://doi.org/10.1016/S0141-6359\(00\)00050-7](https://doi.org/10.1016/S0141-6359(00)00050-7)
27. Chatelain J.F. A level-based optimization algorithm for complex part localization. *Precision Engineering*. 2005. **29** (2). P. 197–207. <https://doi.org/10.1016/j.precisioneng.2004.07.002>
28. Petrakov Y., Shuplietsov D. Contour milling programming technology for virtual basing on a CNC machine. *Eastern-European Journal of Enterprise Technologies*. 2019. **2** (1). P. 54–60. <https://doi.org/10.15587/1729-4061.2019.162673>
29. Petrakov Y., Shuplietsov D. Programming of adaptive machining for end milling. *Mechanics and Advanced Technologies*. 2017. **1** (79). P. 34–40. <https://doi.org/10.20535/2521-1943.2017.79.97342>
30. Stoyan Y., Pankratov A., Romanova T. Placement problems for irregular objects: Mathematical modeling, optimization and applications. In: Butenko S., Pardalos P., Shylo V. (eds.) *Optimization Methods and Applications*. SOIA. **130**. Springer, Cham, 2017. P. 521–559. https://doi.org/10.1007/978-3-319-68640-0_25
31. Дубинський В.М., Панкратов О.В., Романова Т.Є., та ін. Опукла багатокутна оболонка для пари нерегулярних об'єктів. *Cybernetics and Computer Technologies*. 2021. **2**. С. 13–24. <https://doi.org/10.34229/2707-451X.21.2.2>
32. Romanova T., Pankratov O., Litvinchev I., et al. Balanced circular packing problems with distance constraints. *Computation*. 2022. **10** (7). Art. No. 113. <https://doi.org/10.3390/computation10070113>

33. Wächter A., Biegler L.T. On the implementation of an interior-point filter line-search algorithm for large-scale nonlinear programming. *Mathematical Programming*. 2006. **106** (1). P. 25–57. <https://doi.org/10.1007/s10107-004-0559-y>

Одержано 06.12.2022

Планковський Сергій Ігорович,

доктор технічних наук, професор,
завідувач кафедри Автоматизації та комп'ютерно-інтегрованих технологій
Харківського національного університету міського господарства імені О.М. Бекетова, Харків,
<https://orcid.org/0000-0003-2908-903X>
sergiy.plankovskyy@kname.edu.ua

Цегельник Євген Володимирович,

кандидат технічних наук, старший дослідник,
докторант кафедри Автоматизації та комп'ютерно-інтегрованих технологій
Харківського національного університету міського господарства імені О.М. Бекетова, Харків,
<https://orcid.org/0000-0003-1261-9890>
y.tsegelnyk@kname.edu.ua

Панкратов Олександр Вікторович,

доктор технічних наук, старший науковий співробітник, провідний науковий співробітник
Інституту проблем машинобудування ім. А.М. Підгорного НАН України, Харків,
<https://orcid.org/0000-0002-2958-8923>
pankratov2001@yahoo.com

Романова Тетяна Євгенівна,

доктор технічних наук, професор, провідний науковий співробітник
Інституту проблем машинобудування ім. А.М. Підгорного НАН України, Харків,
<https://orcid.org/0000-0002-8618-4917>
tarom27@yahoo.com

Максимов Сергій Володимирович,

аспірант Інституту проблем машинобудування ім. А.М. Підгорного НАН України, Харків,
maksimovsergey08@gmail.com

Комбаров Володимир Вікторович,

кандидат технічних наук, доцент кафедри Автоматизації та комп'ютерно-інтегрованих технологій
Харківського національного університету міського господарства імені О.М. Бекетова, Харків,
<https://orcid.org/0000-0002-6158-0374>
volodymyr.kombarov@kname.edu.ua

УДК 621.7/9:004.946

С.І. Планковський¹, Є.В. Цегельник¹, О.В. Панкратов², Т.Є. Романова^{2*}, С.В. Максимов², В.В. Комбаров¹

Математична модель та метод розв'язання задачі віртуального базування

¹ Харківський національний університет міського господарства імені О.М. Бекетова, Україна

² Інститут проблем машинобудування ім. А.М. Підгорного НАН України, Харків

* Листування: tarom27@yahoo.com

Вступ. Розглянуто задачу віртуального базування яка є характерною для випадку ЧПК оброблення при виготовленні деталей з заготовок близької геометричної форми. Відомі алгоритми розв'язку задачі віртуального базування зводяться до визначення параметрів розміщення САД моделі деталі всередині хмари точок, які отримано шляхом сканування поверхні заготовки. Основним недоліком таких алгоритмів є використання критеріїв, нечутливих до перетину поверхонь деталі та заготовки. Для запобігання подібним похибкам у виробничих умовах, необхідно залучати людину-оператора для прийняття рішень

щодо віртуального базування. Таким чином, задача віртуального базування об'єктів складної форми є надзвичайно важливою.

Мета роботи – пропозиція нового підходу до розв'язання задачі віртуального базування.

Результати. Запропоновано нову математичну модель задачі віртуального базування на основі методу *phi*-функцій. Розроблено стратегію розв'язання, яка поєднує алгоритм генерації можливих стартових точок із процедурою нелінійної оптимізації. Тестування запропонованого підходу проводилось для двовимірного випадку. Наведено результати обчислень, проілюстровані графічними матеріалами, що демонструють ефективність запропонованого алгоритму.

Висновки. Отримані результати свідчать про те, що використання методу *phi*-функцій запобігає виникненню помилкових рішень із перетином поверхонь заготовки. Розроблено алгоритм вирішення задачі віртуального базування в двовимірній постановці для випадку, коли деталь і заготовка є опуклими багатокутниками. Для розглянутих тестових задач час розв'язування не перевищував 2,5 с, що повністю відповідає вимогам промислового використання. У подальшому планується поширити запропонований метод на випадки, коли CAD модель деталі має довільну форму та формується за допомогою булевих операцій над геометричними примітивами.

Ключові слова: багатокутна область, метод *phi*-функцій, віртуальне базування, ЧПК оброблення.

UDC 621.7/9:004.946

Sergiy Plankovskyy¹, Yevgen Tsegelnyk¹, Oleksandr Pankratov², Tetyana Romanova^{2*}, Serhiy Maximov², Volodymyr Kombarov¹

Mathematical Model and Solution Algorithm for Virtual Localization Problem

¹ O.M. Beketov National University of Urban Economy in Kharkiv, Ukraine

² Anatolii Pidhornyi Institute of Mechanical Engineering Problems of the NAS of Ukraine, Kharkiv

* Correspondence: tarom27@yahoo.com

Introduction. The optimization placement problem referred to virtual localization is studied. This problem is motivated by the need to optimize the production of parts from near-net shape blanks using CNC machines. The known algorithms for solving the virtual localization problem come down to determining the location parameters of the part CAD model inside the point cloud obtained by scanning the workpiece surface. The main disadvantage of such algorithms is the use of criteria that are insensitive to the intersection of the surfaces of the part and the workpiece. In order to prevent such errors in production conditions, it is necessary to involve a human operator in conducting operations based on virtual localization. In this way, the virtual localization problem of complex shape objects is of paramount importance.

The purpose of the paper is to propose a new approach for solving the virtual localization problem.

Results. A new mathematical model of the virtual localization problem based on the *phi*-function technique is proposed. We developed a solution strategy that combines algorithm of generating feasible starting points with non-linear optimization procedure. The testing of the proposed approach was carried out for a two-dimensional case. The computational results illustrated with graphical illustrations are provided that show the efficiency of the proposed algorithm.

Conclusions. The obtained results show that the use of the *phi*-functions technique prevents the occurrence of erroneous solutions with the intersection of the workpiece surfaces. An algorithm for solving the problem of virtual localization in a two-dimensional formulation for the case when the part and the workpiece are convex polygons has been developed. For the considered test problems, the solution time did not exceed 2.5 sec, which fully meets the requirements of industrial use. In the future, it is planned to extend the proposed method to the cases when the CAD model of the part has an arbitrary shape and is formed by Boolean operations on geometric primitives.

Keywords: polygonal domain, *phi*-function technique, virtual localization, CNC machining.