



BỘ HẬU XỬ LÝ CHO MÁY CNC 5 TRỤC

Vương Sĩ Kông

Trường Đại học Sư phạm Kỹ thuật Hưng Yên

Ngày nhận: 27/06/2016

Ngày sửa chữa: 03/08/2016

Ngày xét duyệt: 12/09/2016

Tóm tắt:

Bộ hậu xử lý (Postprocessor- PP) là một giao diện liên kết các hệ thống CAM với máy CNC và nó chuyển đổi dữ liệu CL sao cho tương thích với từng máy CNC cụ thể [1], [2]. Nó có chức năng đọc, dịch các lệnh sản xuất được đưa ra bởi hệ thống CAM và chuyển đổi chúng thành mã NC tùy thuộc vào sự kết hợp của máy và cấu hình điều khiển. Bài báo này trình bày sơ lược về Postprocessor, đề xuất thuật toán tuyến tính hóa đường chạy dao và xây dựng bộ hậu xử lý cho máy CNC 5 trục.

Từ khóa: Tuyến tính hóa đường chạy dao; Bộ hậu xử lý; Máy CNC 5 trục; Động học ngược; Lập trình CNC; CAD/CAM/CNC.

Các từ viết tắt

CC	Cutter Contact
CL	Cutter Location
PP	Post-processor
CAD	Computer-Aided Design
CAM	Computer-Aided Manufacturing
CNC	Computer Numerical Control
MCS	Machine Coordinate System
WCS	Workpiece Coordinate System

1. Mở đầu

Postprocessor là một mô-đun của phần mềm CAM vận hành, giữ vai trò giao diện giữa CAM và CNC. CAM là phần mềm máy tính chuyên dụng có nhiệm vụ cung cấp chương trình điều khiển cho các thiết bị sản xuất là các máy CNC. Yêu cầu cơ bản đối với một chương trình NC là nó phải tương thích với máy mà nó phục vụ. Sự tương thích ở đây bao gồm cả về ngôn ngữ lẫn cấu trúc, tính năng của máy. Nói chung, các thông tin chứa trong chương trình NC gồm 2 loại: Loại thứ nhất chỉ liên quan đến các yếu tố nội tại của quá trình gia công, gồm quỹ đạo chuyên động tương đối của lưỡi cắt sao với phôi, các thông số công nghệ như: dụng cụ, lượng tiến dao, tốc độ trục chính,... đảm bảo chất lượng và tính kinh tế của quá trình. Loại thứ 2 liên quan đến đặc tính kỹ thuật của máy công cụ, như kiểu bộ điều khiển (mỗi bộ điều khiển chỉ có thể hiểu và xử lý được chương trình có cấu trúc và cú pháp nhất định), cấu trúc cơ khí của máy công cụ (số lượng và loại trục điều khiển, giới hạn các thông số công nghệ,...). Bất kỳ phần mềm CAM nào cũng phải thỏa mãn yêu cầu trên, có 2 giải pháp kỹ thuật cho vấn đề này:

Một là tạo ra phần mềm CAM duy nhất, chuyên dụng để trực tiếp xuất chương trình gia công

cho mỗi máy CNC cụ thể. Giải pháp này kém hiệu quả vì mỗi nhóm máy đòi hỏi một phần mềm CAM cho riêng mình.

Hai là tách phần mềm CAM thành 2 mô-đun. Mô-đun thứ nhất gọi là bộ xử lý (Processor), có nhiệm vụ xuất một chương trình trung gian, dùng chung chỉ mô tả quỹ đạo của dao và dùng một ngôn ngữ chung, chưa cần quan tâm đến cấu trúc, tính năng của máy và bộ điều khiển cụ thể. Mô-đun thứ hai gọi là bộ hậu xử lý thông dịch chương trình trung gian sang dạng tương thích với máy CNC cụ thể. Khi gặp một bộ máy CNC mới, ta chỉ việc tạo ra một PP tương ứng.

Các phần mềm CAD/CAM có đầy đủ chức năng rất đắt, nhưng chỉ đảm bảo chức năng xử lý. Nhờ có bộ hậu xử lý mà ta có thể dùng chung phần mềm CAD/CAM cho nhiều máy CNC.

Để chuẩn bị các chương trình NC cho các máy CNC phải quan tâm đến cả hai yếu tố: yếu tố hình học của sản phẩm gia công và yếu tố về các yêu cầu chuyên biệt khác nhau của các máy công cụ. Trong thực tiễn hiện nay, tất cả các hệ thống đều quan tâm đến điều đó bằng cách hoặc là sử dụng một chương trình đơn duy nhất, hoặc là bằng cách sử dụng hai chương trình (hai công đoạn riêng rẽ) để chia sẻ gánh nặng trong quá trình tạo lập các chương trình gia công, đó là:

- Quá trình xử lý đưa ra kết quả chung.
- Quá trình hậu xử lý đưa ra kết quả riêng biệt.

Quá trình xử lý đưa ra xuất kết quả chung sẽ sinh ra dữ liệu vị trí qui ước của dụng cụ (CL-data). Thông thường, CL-data không phụ thuộc vào các máy CNC riêng biệt. Về mặt bản chất, CL-data là một tập hợp các bộ giá trị tọa độ X, Y, Z của vị trí mũi dao (Tooltip), các phần tử của tập hợp được

sắp xếp tuần tự theo trình tự di chuyển của dụng cụ theo quỹ đạo hình học. Các câu lệnh đặc biệt trong chương trình gia công (ví dụ như tốc độ trục chính, lượng tiến dao, thay đổi dụng cụ, đổi phôi,...) chỉ có thể được xử lý bởi Postprocessor.

Các hãng sản xuất máy công cụ khác nhau hoàn toàn có thể đưa ra các cấu trúc máy công cụ với các bộ điều khiển CNC khác nhau. Có những bộ điều khiển yêu cầu các vị trí chạy dao phải được tính toán sao cho gia tốc của các trục trong chuyển động chạy dao không được vượt quá giá trị tối đa cho phép. Tuy nhiên, hầu hết các hệ thống khác hiện nay, có tích hợp với các mạch điện tử, cho phép giải quyết vấn đề này. Do vậy, cần thiết phải tiến hành các tính toán đặc biệt sao cho chương trình gia công tương thích được với mỗi tổ hợp máy công cụ và bộ điều khiển cụ thể. Với các ngôn ngữ lập trình gia công đưa ra kết quả chung, ví dụ như: APT, ADAPT, EXAPT, v.v..., việc xây dựng các phép tính toán khác nhau phụ thuộc vào các tổ hợp máy khác nhau trong một chương trình là cần thiết, chương trình đó được gọi là Postprocessor.

Các hệ thống CAD/CAM ngày nay làm cho các ngôn ngữ lập trình xử lý kết quả chung trở nên lỗi thời, việc tính toán đường chạy dao là một nhiệm vụ của các phần mềm CAD/CAM này. Đối với các phần mềm như vậy, người sử dụng tiến hành gia công trên màn hình đồ họa (graphic machining), thông thường, các phần mềm CAD/CAM cũng sử dụng cùng kiểu các câu lệnh (menu lệnh) với các ngôn ngữ lập trình kể trên. Phần CAM của các hệ thống thực hiện các phép tính hoàn toàn tương tự để đưa ra đường chạy dao giống như các ngôn ngữ lập trình NC đưa ra kết quả chung khác. Đường chạy dao có thể được hiển thị trên màn hình đồ họa tùy thuộc vào người sử dụng. Điều này cho phép kiểm tra, đánh giá chương trình gia công không cần chạy máy trước khi sử dụng.

Các hệ thống CAD/CAM cũng thường bao gồm các công cụ để phát triển PP. Tuy nhiên rất nhiều hệ thống CAD/CAM không đưa ra kết nối trực tiếp với CL-data tiêu chuẩn (độc lập với các máy CNC). Phần PP phát triển trong các môi trường khép kín như thế này sẽ rất khó hoặc không thể sử dụng được cùng với các hệ thống khác. Một vài hệ thống CAD/CAM có thể sinh ra các mã chương trình gia công theo mô hình hình học hoặc thậm chí cả chương trình gia công hoàn chỉnh. Các mã chương trình này sau đó phải được xử lý thêm nhờ bộ xử lý ngôn ngữ hoặc bộ PP cho các máy CNC cụ thể.

2. Các yếu tố chính cần giải quyết trong PP

2.1. Động học máy CNC

Đối với các hệ điều khiển theo dạng đường

liên tục bộ postprocessor phải tính đến động học[5] của chuyển động để tránh sai số vị trí.

2.2. Phép biến đổi hình học đối với các máy CNC nhiều trục

Trong quá trình sinh ra đường chạy dao (toolpath generation), trên các máy CNC nhiều trục đòi hỏi một quá trình xử lý tính toán sự định vị của trục dụng cụ (tool axis orientations). Trong trường hợp này, quá trình tính toán chung sẽ sinh ra dữ liệu CL, trong đó mỗi bản ghi của dữ liệu này chứa đựng thông tin theo dạng X Y Z I J K [6]. Các giá trị X Y Z là giá trị tọa độ của các điểm mũi dao trên quỹ đạo chạy dao trong hệ tọa độ phôi, và các giá trị I J K là các cosin chỉ phương của véc tơ trục dụng cụ tương ứng với các vị trí của dao kể trên trong cùng một hệ tọa độ. Do vậy, các bộ PP cho các máy CNC nhiều trục phải có khả năng chuyển đổi các giá trị X Y Z I J K của CL data sang các dịch chuyển của các trục của mỗi máy CNC có cấu hình cụ thể.

2.3. Hiệu chỉnh sai lệch (bù sai số)

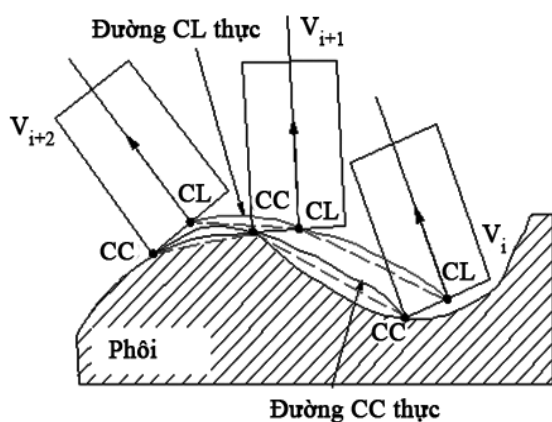
Khi yêu cầu tái định vị trục dụng cụ, thì cần thiết phải dịch chuyển các tâm quay trong khi quay một hoặc cả hai trục quay một cách đồng thời. Nếu các dịch chuyển tịnh tiến và dịch chuyển góc này là có ở trong cùng một câu lệnh gia công trong bộ điều khiển CNC, thì các dịch chuyển tuyến tính và dịch chuyển góc sẽ đồng thời diễn ra theo gia số thời gian. Kết quả là, quỹ đạo của điểm mũi dao sẽ tuân theo một đường cong nào đó có quan hệ với phôi gia công. Việc hiệu chỉnh sai lệch giữa các đường cong này và các đoạn tuyến tính như mô tả trong CL data là trách nhiệm của phần PP. Trong thực tế, các bộ PP hiện hành mới chỉ đảm bảo được rằng sai lệch của quỹ đạo thực của điểm mũi dao nằm trong giới hạn có thể nào đó so với quỹ đạo mong muốn.

Lẽ tự nhiên, quỹ đạo phi tuyến của điểm mũi dao luôn có nhu cầu được tuyến tính hóa [7],[8]. Khi độ cong của quỹ đạo dao xấp xỉ với độ cong của bề mặt gia công, thì có thể cho phép PP sát nhập một số các câu lệnh cắt thẳng thành một câu lệnh đơn. Điều này cho phép nâng cao chất lượng bề mặt gia công và hiệu suất làm việc của máy.

3. Tạo postprocessor cho máy CNC 5 trục

3.1. Thuật toán tuyến tính hóa đường chạy dao

Khi gia công trên máy CNC 5 trục, các đường chạy dao thực sự giữa hai điểm CL không phải tuyến tính mà là đường cong. Các đường cong trong Hình 1 cho thấy sự khác biệt giữa điểm CL và điểm CC thực tế. Đường chạy dao thực này rõ ràng là một lỗi do nội suy tuyến tính của máy 5 trục tạo ra. Lỗi này cần được kiểm tra và bù sai số, và điều này được thực hiện trong PP [10].



Hình 1. Đường dẫn công cụ thực tế giữa hai điểm CL [10]

Tuyến tính hóa đường chạy dao được thực hiện trong PP bởi nội suy dữ liệu điểm CL mới dọc theo đường chạy dao lý tưởng và do đó thêm khối mới cho chương trình NC. Một lệnh trong file CL có thể tạo ra một vài dòng trong tập tin NC, vì vậy kích thước của chương trình NC tăng lên. Các đường chạy dao cuối cùng bao gồm các vị trí có nguồn gốc

trực tiếp từ file dữ liệu CL, và vị trí mới được tạo ra bởi CL nội suy dữ liệu trong PP. Tuyến tính hóa sẽ không cung cấp chuyển động hoàn hảo của dụng cụ, nhưng độ sai lệch với đường chạy dao lý tưởng có thể được giảm đến một [9] mức độ chấp nhận được.

Nếu đường chạy dao thực lệch hơn dung sai T đối với đường CL tuyến tính, thì đường chạy dao sẽ phải được tuyến tính hóa. Vấn đề tuyến tính hóa được giải quyết trong PP bằng cách chèn các điểm trung gian giữa hai điểm CL liên tiếp, khi độ lệch nằm trong miền dung sai không thỏa mãn.

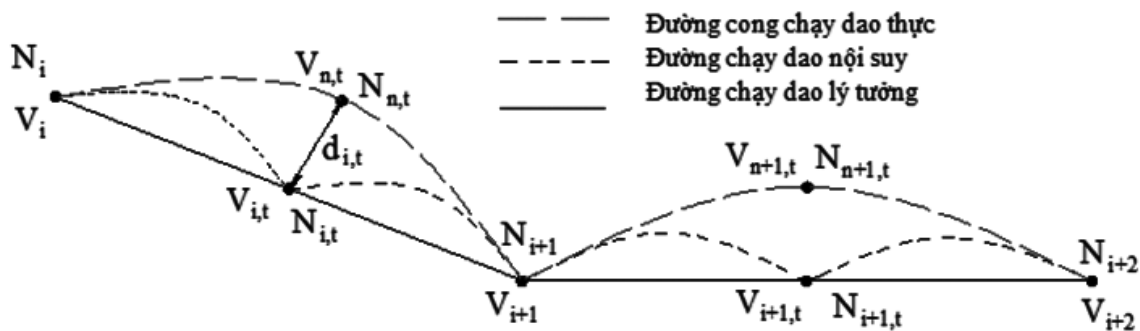
Cho rằng V_i , V_{i+1} và V_{i+2} là ba điểm liên tiếp trong dữ liệu CL (Hình 2). Các chương trình NC tương ứng của V_i là $N_i = [X_i, Y_i, Z_i, B_i, C_i]$. Mỗi trục được giả định để di chuyển tuyến tính giữa các điểm quy định. Vì vậy, mỗi điểm trong đường cong thực tế có thể được thể hiện là:

$$N_{n,t} = N_i + t(N_{i+1} - N_i) \quad (1)$$

Trong đó t là thời gian ($0 \leq t \leq 1$).

Tương ứng CL dữ liệu $V_{i,t}$ cho $N_{n,t}$ có thể được xác định bởi phương trình động lực học trong [3, 4]. Hơn nữa, mỗi điểm trên đường chạy dao lý tưởng có thể được xác định như sau:

$$V_{i,t} = V_{i+1}(V_{i+1} - V_i) \quad (2)$$



Hình 2. Tuyến tính đường chạy dao

Có thể được nhìn thấy từ Hình 2, khoảng cách giữa $V_{n,t}$ và $V_{i,t}$ tạo thành một độ lệch, ký hiệu $d_{i,t}$. Nếu độ lệch tối đa $(d_{i,t})_{max}$ vượt quá dung sai cho phép thì các dữ liệu $V_{i,t}$ được chèn vào các dữ liệu CL gốc. Thông thường, điểm trung bình có $t = 0,5$ là điểm được chọn. Sau khi điểm trung gian $V_{i,t}$ đã được đưa vào, các mã NC tương ứng có thể được tạo ra.

Thuật toán tuyến tính hóa đường chạy dao như sau:

1. Chọn $t = 0.5$
2. Đọc V_i, V_{i+1} từ CL file
3. Tính toán $V_{i,t} = V_i + t(V_{i+1} - V_i)$
4. Tính toán $N_{n,t}, N_{n+1,t}$ từ V_i và V_{i+1} trên cơ sở động lực học ngược [3, 4].

5. Tính $N_{n,t} = N_i + t(N_{i+1} - N_i)$

6. Tính $V_{n,t}$ từ $N_{n,t}$ trên cơ sở động lực học thuận

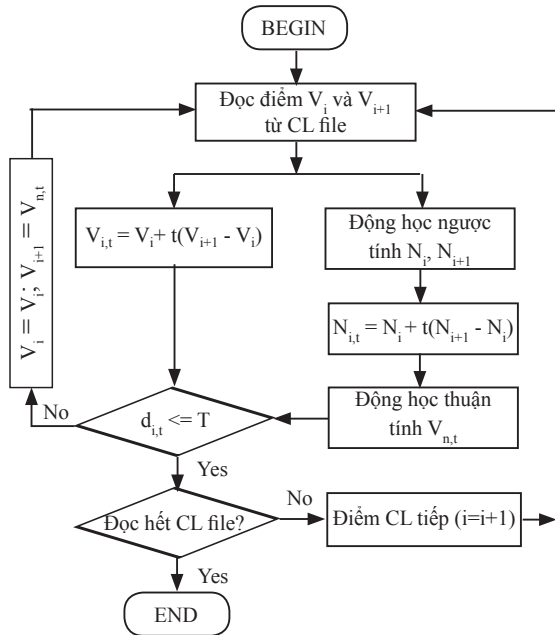
7. Tính khoảng cách $d_{i,t}$

8. So sánh $d_{i,t}$ với dung sai T

- Nếu $d_{i,t} \leq T$, thì đọc tiếp điểm CL, $i = i + 1$.

- Nếu $d_{i,t} > T$, chèn $V_{i,t}$ ở giữa V_i và V_{i+1} và lặp lại từ bước 2 đến bước 8.

Sơ đồ thuật toán tuyến tính hóa đường chạy dao được thể hiện trong Hình 3.

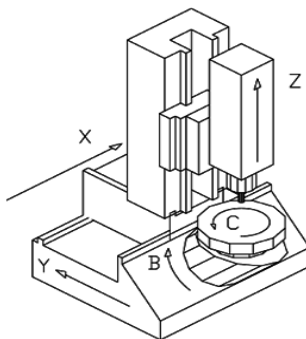


Hình 3. Thuật toán tuyến tính đường chạy dao

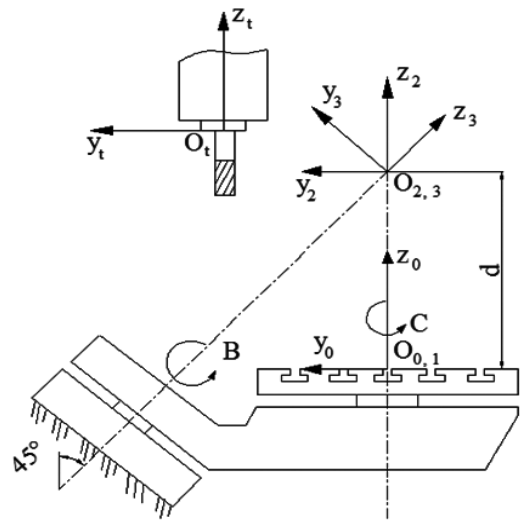
3.2. Tính toán hệ tọa độ máy

Dữ liệu CL là những vị trí cần cắt (x, y, z) và định hướng (i, j, k) được định nghĩa trong hệ tọa độ phôi. Ở đây các dữ liệu x, y, z, i, j, k phải được chuyển về hệ tọa độ máy X, Y, Z, và (B, C) hoặc (A, B) hoặc (A, C) để điều khiển sự chuyển động của trục máy. Bằng phương pháp biến đổi động học ngược, có thể được tính toán được dữ liệu CL gồm có: ba chuyển động liên tuyến tính (X, Y, Z) và hai chuyển động khớp quay (B, C) hoặc (A, B) hoặc (A, C). Điều này có thể được thực hiện bằng việc chuyển đổi hình học từ hệ tọa độ WCS sang hệ tọa độ MCS.

Hình 4 là mô hình máy CNC năm trục DMU 70 eVolution được sử dụng để chứng minh cho phương pháp được đề xuất. Để tính toán tọa độ máy (X, Y, Z, B, C) từ dữ liệu CL (x, y, z, i, j, k), thì phải bổ sung hệ tọa ở một số khớp. Các hệ tọa độ tham chiếu được xác định từ việc chuyển đổi từ hệ tọa độ phôi sang hệ tọa độ máy. Các hệ tọa độ tham chiếu trung gian được hiển thị trong Hình 5 và sự chuyển đổi được tính như sau:



Hình 4. Máy CNC năm trục DMU 70 eVolution



Hình 5. Các hệ tọa độ tham chiếu trung gian

$O_0 (x_0, y_0, z_0)$: nằm ở tâm bề mặt bàn C, khi $B = C = 0^0$. Trục z_0 trùng với trục C .

$O_1 (x_1, y_1, z_1)$: Thu được bằng cách quay (x_0, y_0, z_0) quanh z_0 một góc C .

$O_2 (x_2, y_2, z_2)$: Thu được bằng cách dịch (x_1, y_1, z_1) một khoảng cách +d theo z_0

$O_3 (x_3, y_3, z_3)$: Thu được bằng cách quay (x_2, y_2, z_2) quanh x_2 một góc $+45^0$

$O_4 (x_4, y_4, z_4)$: Thu được bằng cách quay (x_3, y_3, z_3) quanh z_3 một góc B

$O_5 (x_5, y_5, z_5)$: Thu được bằng cách quay (x_4, y_4, z_4) quanh x_4 một góc -45^0

$O_w (x_w, y_w, z_w)$: Thu được bằng cách dịch (x_5, y_5, z_5) một khoảng cách là -d theo z_4

$O_t (x_t, y_t, z_t)$: Hệ tọa độ máy cố định ở trục chính.

Bước 1: Quay quanh z_0 một góc C

$$\begin{aligned} x_1 &= x_0 \cos C - y_0 \sin C \\ y_1 &= x_0 \sin C + y_0 \cos C \\ z_1 &= z_0 \end{aligned} \tag{3}$$

Bước 2: Dịch chuyển $O_1 \rightarrow O_2$ theo trục z_0 khoảng cách +d

$$\begin{aligned} x_2 &= x_1 + x_{01o2} \\ y_2 &= y_1 + y_{01o2} \quad \text{với } z_{01o2} = d \\ z_2 &= z_1 + z_{01o2} \end{aligned} \tag{4}$$

Bước 3: Quay quanh x_2 một góc $+45^0$

$$\begin{aligned} x_3 &= x_2 \\ y_3 &= y_2 \cos 45^0 + z_2 \sin 45^0 \\ z_3 &= -y_2 \sin 45^0 + z_2 \cos 45^0 \end{aligned} \tag{5}$$

Bước 4: Quay quanh z_3 một góc B

$$\begin{aligned} x_4 &= x_3 \cos B + y_3 \sin B \\ y_4 &= -x_3 \sin B + y_3 \cos B \\ z_4 &= z_3 \end{aligned} \tag{6}$$

Bước 5: Quay quanh x_4 một góc -45^0

$$\begin{aligned} x_5 &= x_4 \\ y_5 &= y_4 \cos 45^0 - z_4 \sin 45^0 \\ z_5 &= y_4 \sin 45^0 + z_4 \cos 45^0 \end{aligned} \quad (7)$$

Bước 6: Dịch chuyển một khoảng $-d$ theo z_5

$$\begin{aligned} x_w &= x_5 + x_{o5ow} \\ y_w &= y_5 + y_{o5ow} \quad \text{với } z_{o5ow} = -d \\ z_w &= z_5 + z_{o5ow} \end{aligned} \quad (8)$$

Dữ liệu NC của các thông số X, Y, Z được xác định:

$$\begin{cases} X = x_t = x_w \\ Y = y_t = y_w \\ Z = z_t = z_{O_w O_t} - z_T + z_w \end{cases} \quad (9)$$

Trong đó: z_T là chiều dài của dụng cụ cắt.

Với $\cos B = 2k_0 - 1$, X, Y, Z sẽ là:

$$\begin{aligned} X &= \left[-y_0 \sqrt{2(k_0 - k_0^2)} - x_0 + 2x_0 k_0 \right] \cos C + \\ &+ \left[x_0 \sqrt{2(k_0 - k_0^2)} + 2y_0 k_0 - y_0 \right] \sin C + \\ &+ (d - z_0) \sqrt{2(k_0 - k_0^2)} \end{aligned} \quad (10)$$

$$\begin{aligned} Y &= \left[x_0 \sqrt{2(k_0 - k_0^2)} + y_0 k_0 \right] \cos C + \\ &+ \left[y_0 \sqrt{2(k_0 - k_0^2)} - x_0 k_0 \right] \sin C - z_0 + d - dk_0 + z_0 k_0 \end{aligned} \quad (11)$$

$$\begin{aligned} Z &= \left[x_0 \sqrt{2(k_0 - k_0^2)} + y_0 k_0 - y_0 \right] \cos C + \\ &+ \left[y_0 \sqrt{2(k_0 - k_0^2)} - x_0 k_0 + x_0 \right] \sin C + d - dk_0 + zk_0 \end{aligned} \quad (12)$$

Ở đây $x_0, y_0, z_0, i_0, j_0, k_0$ là vị trí mũi dao và định hướng dụng cụ được đưa ra trong dữ liệu CL.

Bằng phương pháp tương tự và lưu ý rằng

$$\begin{cases} i_4 = 0 \\ j_4 = 0, \text{ thì các trục C và B được xác định:} \\ k_4 = 1 \end{cases}$$

$$C = \arctan \left[\frac{(1 - k_0) i_0 + \sqrt{2(k_0 - k_0^2)} j_0}{(k_0 - 1) j_0 + \sqrt{2(k_0 - k_0^2)} i_0} \right] \quad (13)$$

$$\cos B = 2k_0 - 1 \Rightarrow B = \arccos(2k_0 - 1) \quad (14)$$

3.3. Phần mềm ứng dụng

Một mô-đun phần mềm tuyến tính hóa đường chạy dao đã được phát triển bằng ngôn ngữ Visual Basic. Giao diện người dùng của chương trình được thể hiện trong Hình 6.

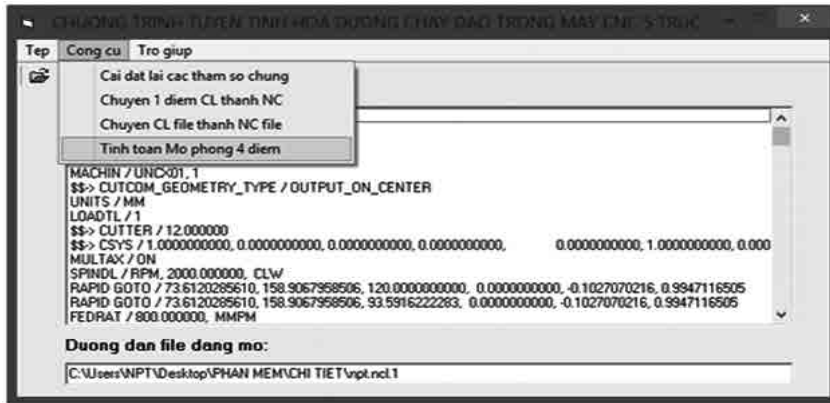
Để thể hiện rõ hiệu quả của công cụ thuật toán tìm đường tuyến tính đầu tiên chúng ta kiểm tra đối với trường hợp của bốn điểm CL. Bảng 1 cho thấy bốn điểm dữ liệu CL để thử nghiệm, và Bảng 2 cho thấy các dữ liệu CL sau khi tuyến tính hóa bằng các thuật toán được đề xuất. Trong ví dụ này chọn dung sai $T = 0.1$, dữ liệu CL sau khi tuyến tính có 11 điểm thay vì bốn điểm; các dòng in đậm trong bảng 2 là các điểm chèn vào. Hình 7 là giao diện người dùng nhập dữ liệu kiểm tra CL. Hình 8 là giao diện của Postprocessor.

Bảng 1. CL data trước khi tuyến tính hóa

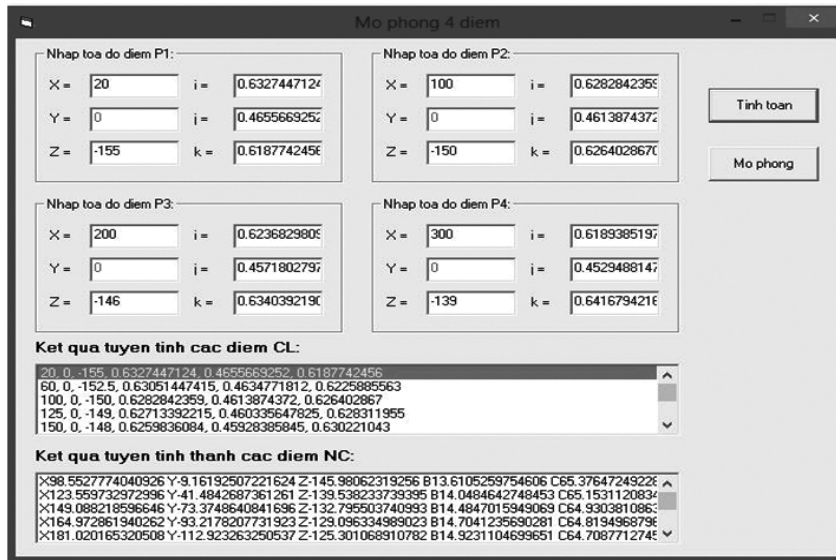
	x	y	z	i	j	k
Điểm 1	20	0	-155	0.6327447124	0.465569252	0.6187742456
Điểm 2	100	0	-150	0.628284235	0.461387437	0.62640867
Điểm 3	200	0	-146	0.623829809	0.457180279	0.634039219
Điểm 4	300	0	-139	0.618938519	0.452948814	0.6416794216

Bảng 2. CL data sau khi tuyến tính hóa

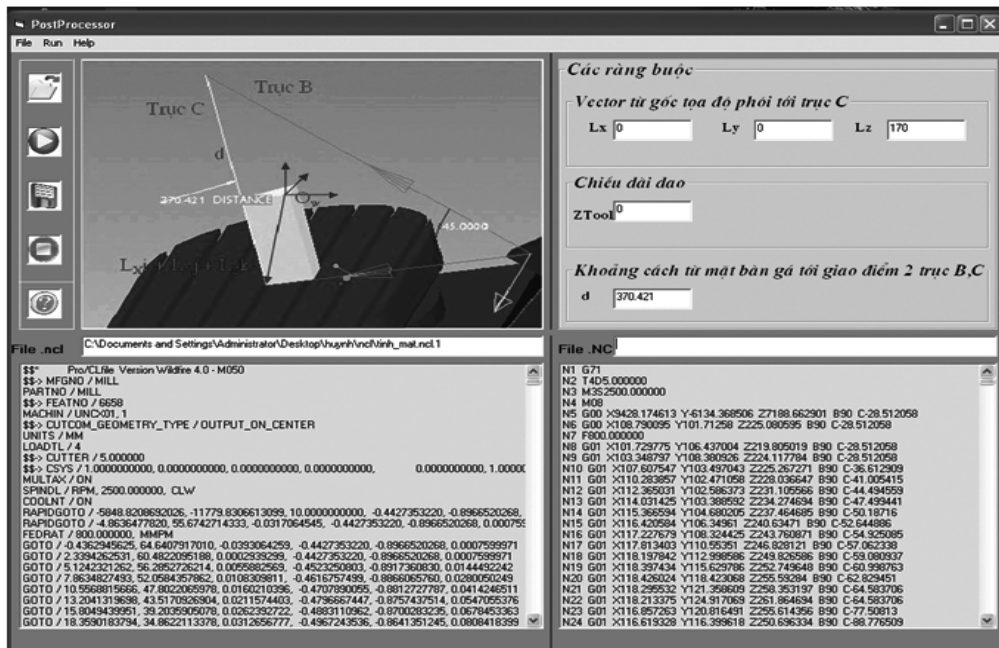
	x	y	z	i	j	k
Điểm 1	20	0	-155	0.6327447124	0.465569252	0.6187742456
Điểm 11	60	0	-152.5	0.63051447415	0.4634771812	0.6225885563
Điểm 2	100	0	-150	0.628284235	0.461387437	0.62640867
Điểm 21	125	0	-149	0.62713392215	0.460335647825	0.628311955
Điểm 22	150	0	-148	0.6259836048	0.45928385845	0.630221043
Điểm 23	175	0	-147	0.62483329465	0.458232069075	0.632130131
Điểm 3	200	0	-146	0.623829809	0.457180279	0.634039219
Điểm 31	255	0	-144	0.6224968656	0.45612241345	0.63594926965
Điểm 32	250	0	-142.5	0.6213107503	0.4550645472	0.6378593203
Điểm 33	275	0	140.75	0.620126435	0.45400668095	0.63976937095
Điểm 4	300	0	-139	0.618938519	0.452948814	0.6416794216



Hình 6. Giao diện chương trình tuyến tính hóa đường chạy dao



Hình 7. Giao diện người dùng để nhập 4 điểm dữ liệu CL



Hình 8. Giao diện của postprocessor

4. Kết luận

Dựa trên phương pháp đã tính toán, tác giả xây dựng một mô đun phần mềm trên nền windows bằng ngôn ngữ lập trình Visual Basic. Kết quả mô phỏng kiểm chứng, áp dụng cho máy phay CNC 5 trục với hai trục quay trên bàn gá phôi (Deckel

Maho DMU 70 eVolution), chỉ ra rằng bộ hậu xử lý có tính toán tuyến tính hóa đường chạy dao làm giảm sai số đường chạy dao, bộ hậu xử lý đáng tin cậy và có thể tùy biến để áp dụng cho các kiểu cấu hình máy CNC khác nhau.

Tài liệu tham khảo

- [1]. Karlo Apro, *Secrets of 5-Axis Machining*, Industrial Press, Inc. New York 2008.
- [2]. Mihir Adivarekar and Frank Liou, *Developing a General Postprocessor for Multi-Axis CNC Milling Centers*, Computer- Aided Design & Applications, pp.57- 68. 2012.
- [3]. Tran Duc Tang, *A Five-axis Postprocessor based on Inverse Kinematics Transformation*, Advanced Material Research 662-663, pp.525-530, 2013.
- [4]. Tran Duc Tang, *Calculation of Machine Coordinates and Postprocessor for 5-axis CNC Milling Machine*, Tuyển tập công trình Hội nghị Cơ điện tử toàn quốc lần thứ 6, tr350-356, 2012.
- [5]. Chen-Hua She, *Design of A Generic Five-axis Postprocessor based on Generalized Kinematics Model of Machine Tool*, International Journal of Machine Tools & Manufacture, pp. 537–545, 2007
- [6]. Hai-Yin Xu & Lian Hu, *A Novel Kinematic Model for Five-axis Machine Tools and its CNC Applications*, Int J Adv Manuf Technol , pp.1297–1307, 2013.
- [7]. Tran Duc Tang, Nguyen Phu Thuy, Vuong Si Kong, *Tool Path Linearization Algorithm and Postprocessor for Five-axis CNC Machine*, Hội nghị toàn quốc lần thứ 3 về Điều khiển và Tự động hoá - VCCA-2015, tr 860-868, 2015.
- [8]. Nuodi Huang, Yong qiao Jin, and QingzhenBi, *Integrated Post-processor for 5-axis Machine Tools with Geometric Errors Compensation*, International Journal of Machine Tools & Manufacture, pp.65–73. 2015
- [9]. Knut Sorby, *Inverse Kinematics of Five-Axis Machines Near Singular Configurations*, International Journal of Machine Tools & Manufacture 47, pp.299-306, 2007.
- [10]. Erik L.J. Bohez, *Compensating System Errors in 5-axis NC Machining*, Computer-Aided Design 34, pp.391-403, 2002.

POSTPROCESSOR FOR FIVE-AXIS CNC MACHINE

Abstract:

Postprocessing is a link interface with computer systems NC and its CAM data conversion CL-compatible so that each particular CNC machine [1], [2]. It has functions to read, translate production orders given by the CAM system and convert them into NC code depending on the combination of machine and control configuration. This paper outlines the postprocessors, proposed linearization algorithm toolpath and construction of post-processing for 5-axis CNC machine.

Keywords: *Linearization toolpath; post-processor; 5-axis CNC machine; Inverse kinematics; CNC programming; CAD/CAM/CNC.*