MỤC LỤC

MỤC LỤC 1
MỞ ĐẦU 2
1. Tính cấp thiết của vấn đề nghiên cứu 2
2. Tổng quan về tình hình nghiên cứu thuộc lĩnh vực đề tài 2
3. Mục tiêu, đối tượng, phạm vi nghiên cứu 2
4. Phương pháp nghiên cứu, kết cấu của công trình nghiên cứu 2
5. Kết quả đạt được của đề tài 3
CHƯƠNG 1. CƠ SỞ LÝ THUYẾT 4
1.1 Cơ sở lý thuyết 4
1.2 Thuật toán xác định hệ số sức cản 10
CHƯƠNG 2. XÂY DỰNG THUẬT TOÁN XÁC ĐỊNH HỆ SỐ SỨC CẢN HỆ TRỤC SERI TÀU B 170 V 14
2.1 Thuật toán xác định hệ số sức cản hệ trục tàu B170-V 14
2.2. Kết luận 22
TÀI LIỆU THAM KHẢO 23

MỞ ĐẦU

1. Tính cấp thiết của đề tài

Như đã biết, việc tính toán dao động xoắn hệ trục tàu thủy còn gặp một số khó khăn trong việc xác định các lực cản trong của vật liệu và kết cấu biên khuỷu, các xi lanh và lực cản của nước lên chong chóng do tính chất phức tạp của bài toán. Lực cản, ma sát bôi trơn v.v... là những đại lượng khó xác định hơn so với lực đàn hồi và lực cưỡng bức khác do chúng là các đại lượng bé bậc cao hơn và khó đo lường chính xác hoặc không thể đo lường được trong thực tế. Cho đến nay, thường các đại lượng này được xác định bằng các công thức thực nghiệm hoặc bán thực nghiệm như trong [1]. Tuy vậy, động cơ tàu thủy và chong chóng hiện đại đã khác nhiều so với vài chục năm trước đây về kích thước, đặc điểm kết cấu và bản thân các động cơ, chong chóng hiện đại cũng rất đa dạng về kiểu loại và đặc điểm động lực học. Vì vậy, có thể khẳng định rằng sử dụng các công thức cho các động cơ và chong chóng trước kia là không còn phù hợp nữa, do đó những tính toán theo công thức truyền thống không cho kết quả phù hợp thực tế.

Hiện nay, nhiều cơ quan thiết kế và đăng kiểm sử dụng các giả thiết là: coi các hệ số sức cản trong ở động cơ là hằng số; mô men cản tỉ lệ với vận tốc góc dao động của các khối lượng (còn gọi là hệ số cản tuyệt đối) hoặc với vận tốc góc biến dạng xoắn (còn gọi là hệ số cản tương đối); Hệ số cản của nước đối với chong chóng tính theo công thức của Arche [2]. Tính toán dao động với các giả thiết trên khá thuận tiện và tương đối phù hợp với mục tiêu thực tiễn, thỏa mãn các yêu cầu của các cơ quan Đăng kiểm đối với việc tính dao động xoắn trong thiết kế đóng mới. Tuy vậy các số liệu về các hệ số sức cản của động cơ và chong chóng đều do các hãng chế tạo máy cung cấp cho nên việc tính toán thiết kế ở Việt nam còn bị động.

Vì những lý do trên, mục tiêu nghiên cứu của đề tài là tìm cách xác định các hệ số sức cản nói trên dựa vào kết quả đo dao động xoắn của các tàu khi thử tàu sau đóng mới, để từ đó tổng kết, thống kê đánh giá và rút ra các công thức xác định các hệ số này cho những loại động cơ, chong chóng đặc trưng khác nhau.

2. Tổng quan về tình hình nghiên cứu thuộc lĩnh vực đề tài

Hiện nay, như tác giả được biết, sức cản dao động xoắn được tính bằng các công thức bán thực nghiệm: công cản nước tác dụng lên chong chóng tính theo công thức của Katudop và Cherski, cản trong của vật liệu theo Teykym; các loại hình cản khác trong động cơ (lên hoặc bên trong các cơ cấu biên khuỷu theo Holzer, Wydler…[1] hoặc Timoshenko… Cũng có thể còn nhiều công thức khác được các nhà sản xuất động cơ, cơ quan thiết kế hoặc Đăng kiểm nước ngoài đề xuất nhưng cơ sở lý thuyết và phương pháp xác định không được trình bày cụ thể.

3. Mục tiêu, đối tượng, phạm vi nghiên cứu

- Mục tiêu của đề tài là tìm phương pháp xác định các hệ số sức cản xoắn khi có kết quả đo dao động xoắn. Khi đã có nhiều dữ liệu về các hệ số này, từ kết quả đo cho nhiều hệ trục khác nhau, và kết hợp với các công thức đã biết, đưa ra công thức hiệu chỉnh tính các hệ số này phục vụ tính dao động xoắn ở giai đoạn thiết kế.
- Đối tượng nghiên cứu: hệ trục chong chong, động cơ chính là diesel, truyền động cơ khí.
- Phạm vi: với hệ trục có số khối lượng tối đa là 11.

4. Phương pháp nghiên cứu, kết cấu của công trình nghiên cứu

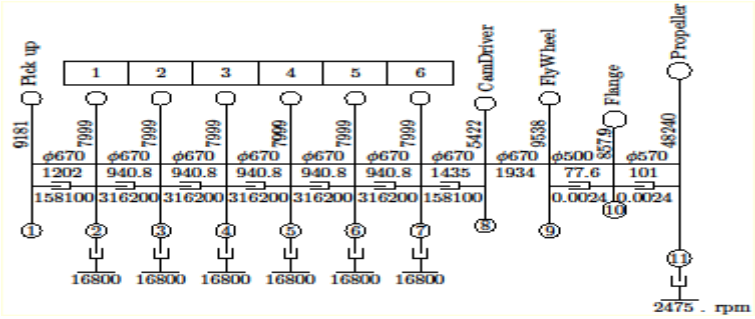
Phương pháp nghiên cứu: dựa trên lý thuyết dao động, phương trình vi phân và đại số tuyến tính và sử dụng gói Symbolic của Matlab.

5. Kết quả đạt được của đề tài

- Đưa ra được thuật toán tính các hệ số sức cản: với mỗi hệ trục cụ thể, cho biết số lượng kết quả đo cần có và sau khi có kết quả đo thì sẽ tính được các hệ số cản chưa biết.

CHƯƠNG 1. CƠ SỞ LÝ THUYẾT

1.1 Cơ sở lý thuyết



Hình 1. Mô hình dao động xoắn hệ trục tàu thủy

Hiện nay, mô hình tính dao động xoắn phổ biến có các đặc điểm sau (hình 1):

- Hệ trục được coi tương đương với hệ thống dao động xoắn gồm *n* khối lượng rời rạc, liên kết nhau bằng các khâu đàn tính không khối lượng như ở hình 1 [1, 2].
- Các lực cưỡng bức dao động là các lực tác dụng lên các khuỷu trục và chong chóng.
- Các loại hình sức cản chính được kể tới là cản trong cơ cấu piston-biên, các khuỷu trục, các cản trong vật liệu các đoạn trục và nước lên chong chóng.

Kí hiệu góc biến dạng xoắn của khối lượng thứ *i* là φ_i, mô men xoắn cưỡng bức tác dụng lên nó là M_i, mô men quán tính khối lượng lượng tập trung I_i, hệ số sức cản tuyệt đối là a_i (sức cản tác dụng lên khối lượng tập trung), hệ số sức tương đối $b_{i,i+1}$ (giữa các khối lượng *i, i+1*) và độ cứng chống xoắn $K_{i,i+1}$. Trong đó, các hệ số sức cản trên đều là các hằng số.

Sử dụng các giả thiết và dung các kí hiệu như trên ta lập được phương trình chuyển động của khối lượng thứ I có dạng:

$$-b_{i-1,i}\dot{\varphi}_{i-1} - K_{i-1,i}\varphi_{i-1} + I_i\ddot{\varphi}_i + (a_i + b_{i-1,i} + b_{i,i+1})\dot{\varphi}_i + (K_{i-1,i} + K_{i,i+1})\varphi_i - b_{i,i+1}\dot{\varphi}_{i+1} - K_{i,i+1}\varphi_{i+1} = M_i$$

với $i=1...n$. (1)

Hệ phương trình đặc trưng của (1) có dạng:

$$z^2.I + z.\beta + K = 0, \quad (2)$$

trong đó:

I, β, K- tương ứng là các ma trận khối lượng; hệ số sức cản và độ cứng:

$$I = \begin{bmatrix} I_1 \ldots\ldots\ldots\ldots \\ \ldots I_2 \ldots\ldots\ldots \\ \ldots\ldots\ldots \\ \ldots\ldots\ldots\ldots I_n \end{bmatrix}; \quad \beta = \begin{bmatrix} (a_1 + b_{1,2})\ldots - b_{1,2}\ldots\ldots\ldots \\ -b_{1,2}\ldots\ldots(a_2 + b_{1,2} + b_{2,3})\ldots - b_{2,3}\ldots\ldots \\ \ldots\ldots\ldots \\ \ldots\ldots\ldots\ldots\ldots - b_{n-1,n}\ldots(a_n + b_{n-1,n}) \end{bmatrix};$$

$$K = \begin{bmatrix} K_{1,2}..-K_{1,2}..... \\ -K_{1,2}..(K_{1,2}+K_{2,3})..-K_{2,3}... \\ ..... \\ .....-K_{n-1,n}..K_{n-1,n} \end{bmatrix}.$$

Để tính các tần số dao động tự do và các biên độ tương đối, nếu sử dụng Matlab thì có thể dùng hàm *polyeig:* [R,λ]=polyeig(A_0,A_1,...,A_p), trong đó λ là véc tơ các giá trị riêng, còn R là ma trận các véc tơ nghiệm của hệ phương trình

$$(A_0 + \lambda * A_1 + ... + \lambda^p * A_p) * R = 0.$$

Nếu sử dụng ngôn ngữ lập trình khác không có hàm trên, có thể xác định các nghiệm riêng của hệ (2) bằng cách tìm z là ẩn của phương trình

$$\det[z^2.I + z.\beta + K] = 0 \qquad (3)$$

bằng cách đặt $z = \mathrm{Re}^{i\varphi} = R(\cos\varphi + i\sin\varphi)$. Sau khi thay z vào (3) thu được hai phương trình đối với phần thực và ảo của vế trái:

- ảo: $\det[2R\cos\varphi * [I] + [\beta]] = 0$;
- thực: $\det[R^2\cos 2\varphi * [I] + R\cos\varphi * [\beta] + [K]] = 0$.

Trong hai phương trình trên, các ẩn cần tìm là Rcosφ và cos2φ. Sau khi giải phương trình thứ nhất, tìm được Rcosφ rồi thay vào phương trình thứ hai sẽ tìm được cos2φ và cuối cùng xác định được z.

Trong trường hợp tính nghiệm cho tàu B 170-V, với các thông số của hệ trục và động cơ chính Sulzer 6RTA62U như sau:

- Các thông số:

Bảng 1. Các thông số

STT	Tên chi tiết	Momen quán tính, kg.m^2	Độ cứng, MNm	Đường kính, mm	Góc nổ, độ
1	Bộ phận đầu trục	9181.0	1201.92	670.0	
2	Xi lanh 1	7999.0	940.82	670.0	0
3	Xi lanh 2	7999.0	940.82	670.0	120
4	Xi lanh 3	7999.0	940.82	670.0	240
5	Xi lanh 4	7999.0	940.82	670.0	180
6	Xi lanh 5	7999.0	940.82	670.0	300
7	Xi lanh 6	7999.0	1434.39	670.0	60
8	Bộ truyền động trục cam	5422.0	1934.24	670.0	
9	Bánh đà	9538.3	77.16	500	
10	Trục trung gian	857.9	101.04	570	
11	Chong chóng	48240.1			

Hệ số sức cản trong động cơ RTA62U và ở chong chóng:

- hệ số cản tuyệt đối (trên một khuỷu): 16800 Nms/rad;
- tương đối giữa hai khuỷu: 316200 Nms/rad;
- ở chong chóng, tại vòng quay 113 v/ph: 237.3 kNms/rad [1].

- Công suất max của động cơ tại 113 v/ph là 13320 kW, tương ứng áp suất chỉ thị trung bình p_i=1.931 MPa.

Kết quả tính dao động tự do được trình bày ở bảng 2, dấu (*) chỉ hình thức dao động và tần số dao động tự do khi xét tới cả ảnh hưởng của các lực cản. Ở đây chỉ đưa ra tới ba hình thức dao động ứng với ba tần số nhỏ nhất.

Kết quả tính dao động tự do khi không kể tới ảnh hưởng của các lực cản thu được hoàn toàn khớp với của Cơ quan thiết kế tàu Ba lan (bảng 1). Khi kể đến ảnh hưởng của các lực cản, các nghiệm riêng của phương trình đặc trưng là các nghiệm phức, phần thực đều có giá trị âm và các dao động riêng sẽ tắt dần. Tần số dao động tự do có giảm đi một chút so với không có cản, song sai lệch này có thể bỏ qua. Điều này cũng dễ hiểu vì ở đây, độ cứng của trục lớn hơn nhiều so với lực cản và quán tính. Modun của biên độ tương đối không thay đổi đáng kể so với khi không cản, song có đặc điểm cần chú ý là khi có cản, pha dao động của các khối lượng sẽ khác nhau.

Để tính dao động cưỡng bức, ta thay momen cưỡng bức tại các xi lanh bằng chuỗi:

$$M_i = \sum_1^m M_{i,k} \sin k(\omega t + \beta_i) = \sum_1^m (M^s{}_{i,k} \sin k\omega t + M^c{}_{i,k} \cos k\omega t), \quad (4)$$

và tìm các nghiệm riêng hệ (1) ở dạng:

$$\varphi_i = \sum_1^m (S_{i,k} \sin k\omega t + C_{i,k} \cos k\omega t). \quad (5)$$

Bảng 2. Dao động tự do

STT	Biên độ tương đối					
	ts=354.1 lần/ph	ts=353.33 l/ph *	ts = 1197.6 lần/ph	ts=1197.4 l/ph *	ts = 2373.6 lần/ph	ts=2373.3 l/ph *
1	1.01061	1.0105-0.00096886i	1.13656	1.1365+0.0042782i	1.89366	1.8930+0.065814i
2	1.0000	1.0000	1.0000	1.0000e+000	1.0000	1.0000
3	0.97475	0.97485-0.0014428i	0.69181	0.69181+0.00099684i	-0.66695	-0.66759-0.015174i
4	0.98311	0.93835-0.0031679i	0.29111	0.29111+0.0032338i	-1.98356	-1.9843-0.029563i
5	0.89050	0.89090-0.0051441i	-0.14853	-0.14851+0.0056162i	-2.25827	-2.2583-0.049786i
6	0.83248	0.83307-0.0073336i	-0.56830	-0.56829+0.0069529i	-1.34676	-1.3455-0.068169i
7	0.76472	0.76552-0.0096933i	-0.91208	-0.91210+0.0062317i	0.27216	0.27490-0.065227i
8	0.71444	0.71542-0.011739i	-1.05750	-1.0575-9.6545e-005i	1.23989	1.2412+0.015521i
9	0.67438	0.67552-0.013625i	-1.11876	-1.1186-0.0042216i	1.74308	1.7412+0.076932i
10	-0.44443	-0.43901-0.069535i	-0.47895	-0.47887-0.0036165i	1.04463	1.0406+0.078248i
11	-1.293858	-1.2851 - 0.11094i	0.07358	0.073561+0.00014533i	-0.03666	-0.036587-0.00078627i

Thay các biểu thức (4) và (5) vào (1), cân bằng các thành phần sin và cos, ta thu được hệ phương trình đại số *2*n* phương trình đối với từng bậc điều hòa *k* dạng

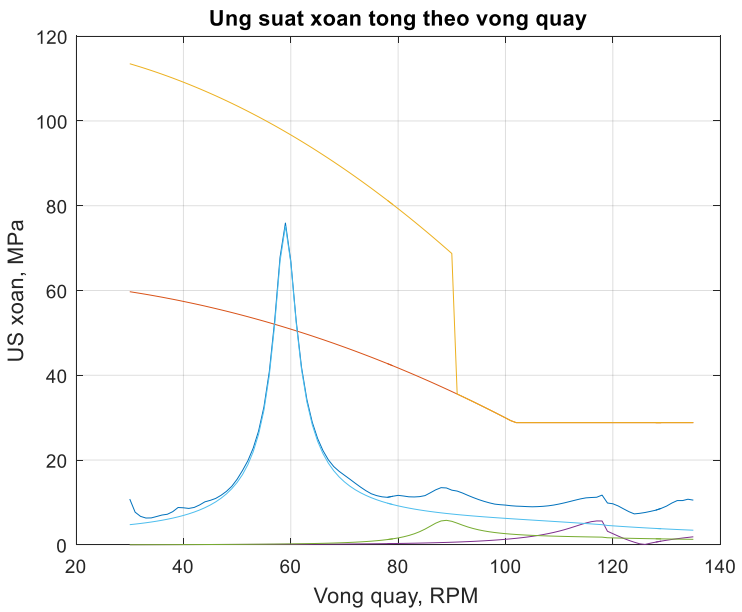
$$\left[(k\omega)^2.I + k\omega\beta + K\right]\left[\varphi\right] = \left[M\right], \tag{6}$$

trong đó: *I, β, K*- tương ứng là các ma trận khối lượng; hệ số sức cản và độ cứng (2n*2n); $\left[\varphi\right]$ và $\left[M\right]$ là các cột giá trị biên độ xoắn và mô men cưỡng bức.

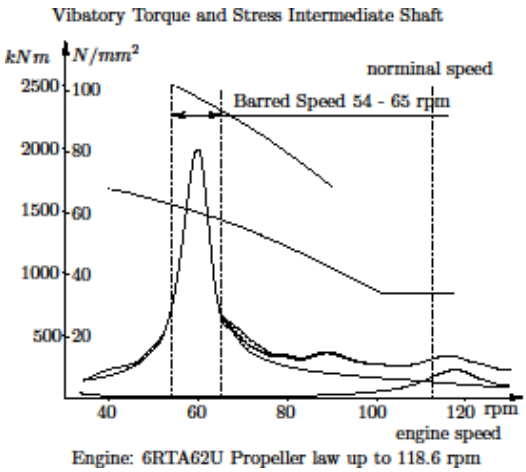
Sau khi giải (6) thu được các biên độ dao động của các khối lượng.

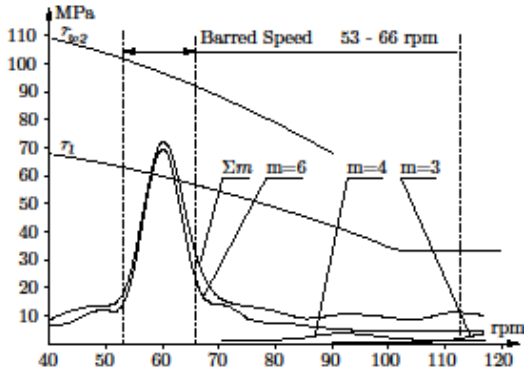
Biên độ ứng suất tại các vòng quay khác nhau tính theo chương trình của tác giả được thể hiện trên hình 2. Kết quả này khá thống nhất với kết quả tính của Ba lan (hình 3.a). Về mặt định tính, kết quả tính thu được khá phù hợp với kết quả đo khi thử tàu (hình 3.a): dải vòng quay cấm từ 53- 66 v/ph, tại khu vực này ứng suất dao động vượt quá cho phép đối với chế độ làm việc liên tục (xác định theo đăng kiểm Đức). Vòng quay nguy hiểm nhất là 59.1 v/ph, tại bậc điều hòa k=6.

Hình 4 biểu thị biên độ dao động max tại trục trung gian ở các vòng quay khác nhau nhau quá trình thử tàu. Hình 5 biểu thị ứng suất thay đổi theo thời gian đo được tại trục trung gian ở các vòng quay 39.1, 54.1 và 65.4 v/ph. So với kết quả đo thực tế (hình 4, 5), giá trị biên độ thu được bằng tính toán lớn hơn so với đo được (biên độ ứng suất tại trục trung gian tính được tại các vòng quay 39.1, 54.4 và 65.4 v/ph là: 8.9416, 27.565 và 24.494 MPa so với đo được (hình 5) là 8.1, 24,4 và 24.4 MPa) song khá giống nhau về định tính: vòng quay có ứng suất cực trị. Có nhiều lí do dẫn đến các sai lệch trên, nhưng kết quả như vậy hoàn toàn có thể chấp nhận được và cho phép kết luận là mô hình tính toán phù hợp với thực tế.



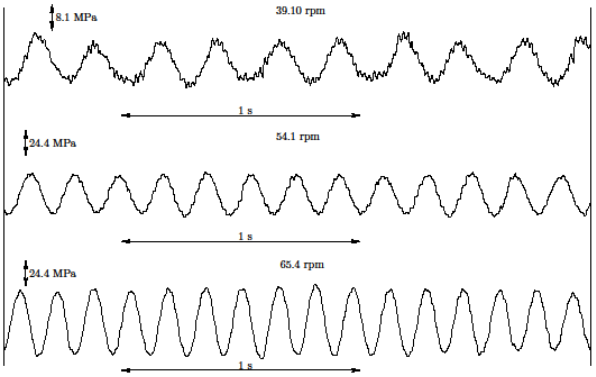
Hình 2. Biên độ ứng suất trục trung gian tác giả tính được ở các vòng quay.



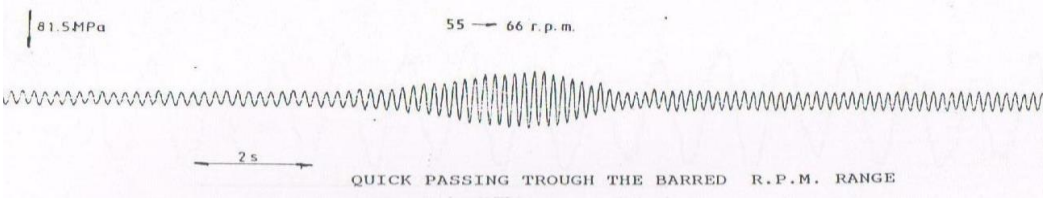


Hình 3.a. Biên độ ứng suất tính theo vòng quay tại trục trung gian của Ba lan [2]

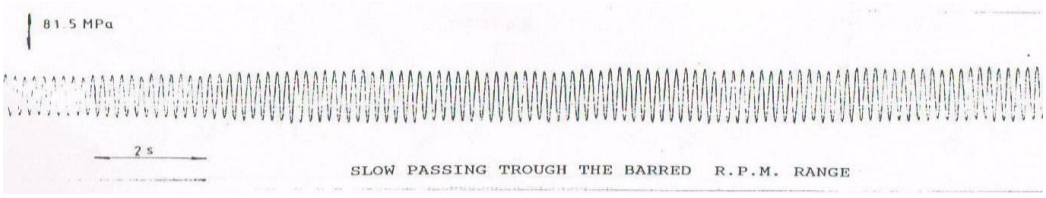
Hình 3.b. Biên độ ứng suất theo vòng quay do tại trục trung gian [2]



Hình 4. Ứng suất theo thời gian do tại trục trung gian ở các vòng quay [2]



a)

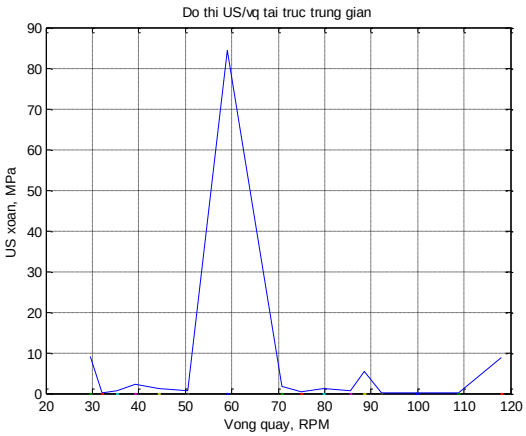


b)

Hinh 5. Dao động đo được khi chuyển qua vùng cấm nhanh chậm khác nhau

Nếu sử dụng phương pháp gần đúng với các giả thiết như vẫn thường dùng trong thực tế (hình thức dao động cưỡng bức khi cộng chấn tương tự như dao động tự do và chỉ xét tới ảnh

hưởng của các cưỡng bức ở các bậc điều hòa gây cộng chấn) thì thu được biên độ ứng suất dao động tại các vòng quay cộng chấn như trên đồ thị h. 6. Vòng quay nguy hiểm nhất ≈ 59.1 v/ph, do cưỡng bức điều hòa chính k=6. Các giá trị này cũng trùng khớp với kết quả tính theo cách tính chính xác trên (kể cả giá trị ứng suất) và như vậy cũng hoàn toàn có thể áp dụng được khi tính nghiệm dao động cộng hưởng của hệ trục, tuy vậy, ở các vòng quay khác nhau (không cộng chấn hoặc cộng chấn ứng với các điều hòa không chính k≠6, thì giá trị ứng suất thu được do giả thiết kể trên sẽ nhỏ hơn so với thực tế khá nhiều. Như vậy, cần nhấn mạnh thêm, chỉ khi cộng chấn xảy ra ở cấp chính (bậc điều hòa có biên độ hình học lớn nhất), thì ảnh hưởng của các cưỡng bức ở các bậc điều hòa khác có thể có ảnh hưởng không đáng kể. Dao động lúc đó rất giống với dao động điều hòa.



Hình 6. Ứng suất tính gần đúng tại các vòng quay cộng chấn (điểm gấp khúc)

Cần lưu ý khi sử dụng các phương pháp gần đúng để tính dao động cưỡng bức đối với hệ trục có hộp số. Trong trường hợp hệ trục có hộp số, để thuận tiện cho tính toán, thường qui đổi hệ trục thực về hệ tương đương (không có hộp số), trong đó độ cứng và khối lượng của các khối lượng sau hộp số được nhân với i^2 lần (i là tỷ số truyền: $i=n_{out}/n_{in}$). Song sau khi tính được dao động tự do của hệ tương đương trên, để tính dao động cưỡng bức ta phải đưa hệ này về hệ thực. Biên độ dao động và momen đàn hồi, cản... , của hệ thực ở sau hộp số phải được tương ứng nhân và chia cho i lần. Thật vậy, phương trình dao động của các khối lượng trước và sau hộp số:

$$-b_{hs-1,hs}\dot{\varphi}_{hs-1} - K_{hs-1,hs}\varphi_{hs-1} + I^{in}{}_{hs}\ddot{\varphi}^{in}{}_{hs} + (a^{in}{}_{hs} + b_{hs-1,hs})\dot{\varphi}^{in}{}_{hs} + K_{hs-1,hs}\varphi^{in}{}_{hs} = M^{in}{}_{hs},$$

$$I^{out}{}_{hs}\ddot{\varphi}^{out}{}_{hs} + (a^{out}{}_{hs} + b_{hs,hs+1})\dot{\varphi}^{out}{}_{hs} + K_{hs,hs+1}\varphi^{out}{}_{hs} - b_{hs,hs,+1}\dot{\varphi}_{i+1} - K_{hs,hs+1}\varphi_{hs+1} = M^{out}{}_{hs},$$

trong đó: *hs*- chỉ số của hộp số, chỉ số in- ứng với đầu vào hay bánh dẫn; out- đầu ra hoặc của bánh bị dẫn. Nếu thay $\varphi^{out}{}_{hs} = i*\varphi^{in}{}_{hs}$ và $i*M^{out} = -M^{in}$ vào hai phương trình trên, thu được:

$$-b_{hs-1,hs}\dot{\varphi}_{hs-1} - K_{hs-1,hs}\varphi_{hs-1} + (I^{in}{}_{hs} + I^{out}{}_{hs}.i^2)\ddot{\varphi}^{in}{}_{hs} + [a^{in}{}_{hs} + b_{hs-1,hs} + (a^{out}{}_{hs} + b_{hs,hs+1})i^2]\dot{\varphi}^{in}{}_{hs} +$$
$$+ (K_{hs-1,hs} + K_{hs,hs+1}.i^2)\varphi^{in}{}_{hs} - i.b_{hs,hs,+1}\dot{\varphi}_{i+1} - i.K_{hs,hs+1}\varphi_{hs+1} = 0.$$

Các phương trình cho các khối lượng sau hộp số, ví dụ *hs+1* sẽ có dạng:

$$-b_{hs,hs+1}\dot{\varphi}^{out}{}_{hs} - K_{hs,hs+2}\varphi^{out}{}_{hs} + I_{hs+1}\ddot{\varphi}_{hs+1} + (a_{hs+1} + b_{hs,hs+1} + b_{hs,hs+2}+)\dot{\varphi}_{hs+1} + (K_{hs,hs+1} + K_{hs,hs+2})\varphi_{hs}$$
$$-b_{hs,hs,+2}\dot{\varphi}_{hs+2} - K_{hs,hs+2}\varphi_{hs+2} = M_{hs+1}.$$

Để loại bỏ biến $\varphi^{out}{}_{hs}$ $\varphi^{in}{}_{hs}$ ra khỏi các phương trình, ta nhân tất cả các phương trình sau hộp số lên i lần, và thay $\varphi^{out}{}_{hs} = i * \varphi^{in}{}_{hs} = i * \varphi_{hs}$, thu được hai phương trình trên ở dạng:

$$-b_{hs-1,hs}\dot{\varphi}_{hs-1} - K_{hs-1,hs}\varphi_{hs-1} + (I^{in}{}_{hs} + I^{out}{}_{hs}.i^2)\ddot{\varphi}^{in}{}_{hs} + [a^{in}{}_{hs} + b_{hs-1,hs} + (a^{out}{}_{hs} + b_{hs,hs+1})i^2]\dot{\varphi}^{in}{}_{hs} +$$
$$+(K_{hs-1,hs} + K_{hs,hs+1}.i^2)\varphi^{in}{}_{hs} - i^2.b_{hs,hs,+1}\dot{\varphi}_{i+1}/i - i^2.K_{hs,hs+1}\varphi_{hs+1}/i = 0,$$

$$-i^2.b_{hs,hs+1}\dot{\varphi}_{hs} - i^2.K_{hs,hs+2}\varphi_{hs} + i^2 I_{hs+1}\ddot{\varphi}_{hs+1}/i + i^2.(a_{hs+1} + b_{hs,hs+1} + b_{hs,hs+2}+)\dot{\varphi}_{hs+1}/i +$$
$$+i^2.(K_{hs,hs+1} + K_{hs,hs+2})\varphi_{hs}/i - i^2.b_{hs,hs,+2}\dot{\varphi}_{hs+2}/i - i^2.K_{hs,hs+2}\varphi_{hs+2}/i = i.M_{hs+1}.$$

Như vậy, biên độ tương đối của hệ thực (φ_j, trong đó j>hs) sẽ lớn hơn i lần so với biên độ của hệ tương đương (bằng φ_j/i) và mômen của chúng bị giảm đi i lần so với hệ qui đổi (có mômen quán tính khối lượng và độ cứng, hệ số cản tăng i^2 lần).

Kết luận: sử dụng giả thiết các lực cản tỉ lệ tuyến tính với tốc độ dao động thuận tiện cho tính toán bằng máy tính và có thể cho kết quả phù hợp với thực tế. Vấn đề tiếp theo cần giải quyết là các phương pháp xác định các hệ số cản, hệ số các momen điều hòa cho các loại động cơ... Trong trường hợp chỉ nghiệm dao động xoắn cộng hưởng ở bậc điều hòa cấp chính thì có thể sử dụng các giả thiết về hình thức dao động giống với tự do và chỉ xét ảnh hưởng của cưỡng bức cộng chấn.

Ngoài ra, ưu điểm của sử dụng các giả thiết này là cho phép tuyến tính hóa hệ phương trình vi phân và tính dao động cưỡng bức một cách liên tục theo vòng quay [3]. Khi coi sức cản phụ thuộc không tuyến tính vào biên độ dao động và vận tốc dẫn đến phải giải hệ phương trình vi phân không tuyến tính nên chỉ thường dùng để tính toán dao động cộng hưởng với giả thiết là hình thức dao động cưỡng bức giống như của dao động tự do [1, 3]. Xong vấn đề cần thiết đặt ra hiện nay là xác định các hệ số sức cản trên như thế nào. Nếu chỉ phụ thuộc vào nhà sản xuất và số liệu của họ đưa ra, thì lấy cơ sở nào để cho rằng nó là tin cậy? Dựa vào đâu để khẳng định được việc tính toán dao động ở giai đoạn thiết kế đảm bảo chính xác? Có thể suy luận rằng, xác định các hệ số trên vẫn còn là vấn đề cần được giải quyết ngay cả đối với ngành đóng tàu thế giới vì các cơ quan Đăng kiểm nước ngoài vẫn đòi hỏi phải đo lại dao động xoắn đối với những thiết kế mới.

1.2 Thuật toán xác định hệ số sức cản

Giả sử trong bài toán tính biên độ dao động oắn khi biết các hệ số sức cản, ta lập được hệ phương trình đại số dạng (6):

$$\left[(k\omega)^2.I + k\omega\beta + K\right]\left[\varphi\right] = \left[M\right],$$

Đặt:

$F_k = \left[(k\omega)^2.I + k\omega\beta + K\right] = \left[f_{i,j}\right]$ và biểu diễn (6) ở dạng:

$$F_k A_k = M_k \tag{7}$$

trong đó:

$M_k = [M^s_{1,k}, M^s_{2,k} ..., M^s_{n,k}, M^c_{1,k}, M^c_{2,k} ...., M^c_{n,k}]^T$ - véc tơ cột của các biên độ mô men cưỡng bức tại cấp điều hòa *k*;

$A_k = [S_{1,k}, S_{2,k}, ..., S_{n,k}, C_{1,k}, C_{2,k}, ..., C_{n,k}]^T$ là véc tơ cột của các biên độ góc xoắn ứng với bậc điều hòa *k*;

$F_k = [f_{i,j}]$ là ma trận vuông cấp *2n*, các phần tử $f_{i,j}$ là hàm của *k*ω, độ cứng các đoạn trục $K_{i,i+1}$, momen quán tính I_i của các khối lượng và các hệ số sức cản a_i, $b_{i,i+1}$ [4].

Nếu xác định được các phần tử $f_{i,j}$ của F_k, thì các biên độ biến dạng xoắn của các khối lượng do cấp điều hòa thứ *k* sẽ là:

$$A_k = F_k^{-1} M_k . \tag{8}$$

Từ đó biên độ mô men đàn hồi xoắn ở cấp điều hòa *k* trên đoạn trục *i*, *i+1* tính được bằng công thức:

$$E^k_{i,i+1} = K_{i,i+1}\sqrt{(S_{i,k} - S_{i+1,k})^2 + (C_{i,k} - C_{i+1,k})^2} = K_{i,i+1}\Delta\varphi_{i,i+1} . \tag{9}$$

Giả sử có *m* giá trị hệ số cản a_j và $b_{j,\,j+1}$ chưa biết cần xác định, thì về nguyên tắc, ta cần đo và phân tích mô men đàn hồi thành các hàm điều hòa để thu được *m* giá trị biên độ mô men đàn hồi xoắn trên một đoạn trục *i*, *i+1* (xem hình 2). Từ đó có thể lập được *m* phương trình đại số (8) để giải được *m* giá trị a_j và $b_{j,j+1}$ chưa biết đó. Tuy nhiên, hệ phương trình thu được là hệ nhiều ẩn và bậc cao nên để có thể giải được, số phương trình phải lập có thể nhiều hơn *m* nhằm thu được hệ phương trình tuyến tính với các ẩn mới là tổ hợp tích của các ẩn cũ: $x_i = a_1^{m1}.a_2^{m2}...b_1^{n1}.b_2^{n2}...$

Thật vậy, nếu đo được $E^k_{i,i+1}$ thì ta có thể tính được $\Delta\varphi_{i,i+1}$:

$$\Delta\varphi_{i,i+1} = E^k_{i,i+1} / K_{i,i+1} . \tag{10}$$

Mặt khác, $\Delta S_{i,i+1}$ có thể được tính từ phương trình (8) như sau:

$$\left\{\left[\Phi_{i+1,1...2n}\right] - \left[\Phi_{i,1...2n}\right]\right\} \times \left[M_k\right] = \Delta S_{i,i+1} \det(F_k);$$

$$\left\{\left[\Phi_{n+i+1,1...2n}\right]\right\} - \left\{\left[\Phi_{n+i,1...2n}\right]\right\} \times \left[M_k\right] = \Delta C_{i,i+1} \det(F_k),$$

nên thu được:

$$\Delta\varphi^2_{i,i+1} \det^2(F_k) = \left\{\left(\left[\Phi_{i+1,1...2n}\right] - \left[\Phi_{i,1...2n}\right]\right)^2 + \left(\left[\Phi_{i+1,1...2n}\right] - \left[\Phi_{i,1...2n}\right]\right)^2\right\}.M_k^2 , \tag{11}$$

trong đó $\left[\Phi_{i+1,1...2n}\right]$ và $\left[\Phi_{i,1...2n}\right]$... là các hàng thứ *i* và *i*+1 của ma trận $[\Phi]$ có các phần tử xác định bằng: $\Phi_{i,j} = D_{j,i}$, với $D_{j,i}$ là phần phụ đại số của phần tử (*j*,*i*) của ma trận F_k. Phương trình (11) có *m* ẩn cần tìm là các hệ số sức cản chưa biết. Chúng có thể được xác định nếu ta có *m* hoặc nhiều hơn các giá trị $\Delta S_{i,i+1}$ và $\Delta C_{i,i+1}$ tương ứng với từng ấy giá trị *k*ω.

Để minh họa ta xét một ví dụ đơn giản. Giả sử hệ có 2 khối lượng, các ẩn chưa biết là các hệ số cản tuyệt đối a_1, a_2 và tương đối b. Ngoài ra, giả sử biết độ cứng trục K_{12}, mô men quán tính các khối lượng I_1 và I_2 cùng các biên độ mô men điều hòa tác dụng lên khối lượng M^s_1, M^s_2, M^c_1, M^c_2. Các phần tử của F_k được biểu diễn như ở bảng dưới, trong đó:

$C = K_{12}$; $C_1 = -I_1(k\omega)^2 + K_{12}$; $C_2 = -I_2(k\omega)^2 + K_{12}$; $C_3 = k\omega$; $x_1 = a_1 + b$, $x_2 = b$; $x_3 = a_2 + b$.

Biểu thức của các phần tử F_k

C_1	$-C$	$-C_3x_1$	C_3x_2
$-C$	C_2	C_3x_2	$-C_3x_3$
C_3x_1	$-C_3x_2$	C_1	$-C$
$-C_3x_2$	C_3x_3	$-C$	C_2

Khai triển phương trình (8) nhờ sự hỗ trợ của gói Symbolic Math Toolbox trong phần mềm Matlab ta thu được phương trình tuyến tính đối với x_i, với $i = 1,...,16$ có dạng:

$$\{T_1\ T_2 \dots T_{16}\} \times \{x_1\ x_2 \dots x_{16}\}^T = P, \qquad (12)$$

trong đó:

$x_4 = x_1^2$, $x_5 = x_1x_2$, $x_6 = x_1x_3$, $x_7 = x_1x_2^2$, $x_8 = x_1^2x_3$, $x_9 = x_2x_3$, $x_{10} = x_1x_2x_3$, $x_{11} = x_2^2$,

$x_{12} = x_3^2$, $x_{13} = x_2^3$, $x_{14} = x_1^2x_3^2$, $x_{15} = x_1x_2^2x_3$, $x_{16} = x_2^4$;

$P = M^s_2(C_2C_1^2 - C^2C_1) + M^s_1(C^2C_2 - C_1C_2^2 + C^3 - CC_1C_2) - \Delta S_{i,i+1}(C^4 - 2C^2C_1C_2 + C_1^2C_2^2)$;

$T_1 = -M^c_2C^2C_3 + M^c_1CC_2C_3 - M^c_2CC_2C_3 + M^c_1C_2^2C_3$;

$T_2 = 2M^c_2CC_1C_3 + M^c_2C^2C_3 - M^c_1C_1C_2C_3 - M^c_1C^2C_3 + M^c_2C_1C_2C_3 - M^c_1CC_2C_3 + M^s_2C_2CC_3$;

$T_3 = M^c_1CC_1C_3 - M^c_2C_1^2C_3 - M^c_2CC_1C_3 - M^s_2C^2C_3$; $T_4 = \Delta S_{i,i+1}C_2^2C_3^2 - M^s_2C_2C_3^2 + M^s_2C_2C_3^2$;

$T_5 = 2M^s_2CC_3^2 - 4\Delta S_{i,i+1}CC_2C_3^2 + M^s_1C_2C_3^2 - M^s_2CC_3^2$; $T_6 = 2\Delta S_{i,i+1}C^2C_3^2 - M^s_1CC_3^2$; $T_7 = M^c_2C_3^3$;

$T_8 = -M^c_2C_3^3$; $T_9 = M^c_1C_2C_3^2 - 4\Delta S_{i,i+1}CC_1C_3^2 - 2M^s_1CC_3^2 + M^s_1C_1C_3^2$; $T_{10} = 2M^c_1C_3^3 + M^s_2C_3^3$;

$T_{11} = \{2\Delta S_{i,i+1}(C_1C_2C_3^2 + C^2C_3^2) - M^s_2C_1C_3^2 + M^s_1C_2C_3^2 - M^s_1CC_3^2\}$;

$T_{12} = \Delta S_{i,i+1}C_1^2C_3^2 - M^c_1CC_3^2 + M^s_1C_1C_3^2$; $K_{13} = M^c_1C_3^3 - Mc_1C_3^3 - M^s_2C_3^3 + M^c_2C_3^3$;

$T_{14} = \Delta S_{i,i+1}C_3^4$; $T_{15} = -2\Delta S_{i,i+1}C_3^4$; $T_{16} = \Delta S_{i,i+1}C_3^4$;

Nếu ta có được 16 giá trị P tương ứng với 16 giá trị $k\omega$ thì sẽ thu được 16 phương trình tuyến tính dạng (12). Chú ý rằng các giá trị T_i và P được xác định khi biết $k\omega$, còn các nghiệm cần tìm (là các hệ số cản) không phụ thuộc vào $k\omega$. Hơn nữa, các trị số T_i khác nhau và không tỉ lệ tuyến tính đối với $k\omega$ nên ma trận K(16×16) không bị suy biến và nghiệm x xác định.

Kết luận

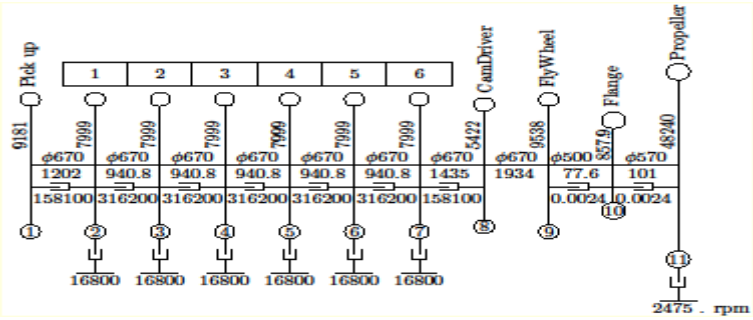
Như vậy, ở trên đã trình bày thuật toán và công thức cho phép xác định được các hệ số sức cản nếu đo được dao động xoắn ở một đoạn trục bất kì. Số kết quả đo cần phải xử lý sẽ phụ thuộc vào từng hệ trục cụ thể (ở ví dụ trên thì là 16). Sử dụng phần mềm Symbolic Matlab hoàn toàn có thể thiết lập được công thức ở dạng biểu tượng cho các hệ số T_i; P và x_i đối với hệ trục thực. Đối với hệ trục thực tế, số khối lượng tuy nhiều hơn (thường trên dưới 10 khối lượng, trong đó số xi lanh đã chiếm khoảng 6…8), xong số ẩn cần tìm- tức là các hệ số cản khó xác định bằng lý thuyết như thủy lực chóng chóng, cản trong xi lanh động cơ và cơ cấu biên khuỷu cũng không nhiều vì chúng như nhau trên một đơn vị khuỷu trục.

Nghiệm của (12) chỉ đúng (nghĩa là thỏa mãn cả các điều kiện $x_4 = x_1^2$, $x_5 = x_1x_2, \ldots, x_{16} = x_2^4$) khi các giả thiết các hệ số cản bằng hằng số là đúng (hoặc tương đối đúng). Khi đó, nghiệm của (4), (5), (6) sẽ đúng như trong thực tế (hoặc tương đối đúng). Ngược lại thì có nghĩa là có sai số giữa đo và tính toán, và các giả thiết là không phù hợp với thực tế.

CHƯƠNG 2. XÂY DỰNG THUẬT TOÁN XÁC ĐỊNH HỆ SỐ SỨC CẢN HỆ TRỤC SERI TÀU B 170 V

2.1 Thuật toán xác định hệ số sức cản hệ trục tàu B170-V

Coi hệ trục tàu thủy tương đương với hệ thống dao động xoắn gồm *n* khối lượng rời rạc, liên kết nhau bằng các khâu đàn tính không khối lượng như ở hình 7 [1, 2].



Hình 7. Mô hình dao động xoắn hệ trục tàu B_170 V, động cơ chính 6 RTA 62 U.

Các thông số đặc trưng tính toán của hệ trục tương đương:

- Số khối lượng tập trung: n=11;
- Mô men quán tính khối lượng của các khối lượng (kgm^2):

I_i = [9181.0 7999.0 7999.0 7999.0 7999.0 7999.0 7999.0 5422.0 9538.3 857.9 48240.1];

- Độ cứng các đoạn trục (MNm/rad)

K_i=[1201.92 940.82 940.82 940.82 940.82 940.82 1434.93 934.24 77.16 101.04];

- Đường kính các đoạn trục (mm):

D_i =[670 670 670 670 670 670 670 670 500 570];

Kí hiệu các hệ số sức cản chưa biết a_i và $b_{i,i+1}$ bằng các biểu tượng *s, z, f, y, r* và thay chúng vào phương trình thay chúng vào phương trình (1), ta thu được ma trận F_K(22,22) cho mỗi giá trị của *kw* (*k*- bậc điều hòa, *w*- vận tốc góc của trục chong chóng), có dạng:

F_k =

[1201920-9.18*kw^2, -1201920, 0,… -kw*r, kw*r,… 0]

[-1201920,2142740-7.99*kw^2, -940820, 0,0,0,0,0,0,0,0, kw*r, -kw*z, kw*s,…… 0]

[0, -940820,1881640-7.99*kw^2,-940820, … kw*s, -f*kw, kw*s, …0]

……

[-kw*r, kw*z, -kw*s,.. ,-1201920, 2142740-7.99*kw^2,-940820,… 0]

……

[0, 0, ... ,-77160, 178200-0.85*kw^2, -101040]

[0, … 0,.. ,kw*y, ... 0, -101040, 101040-4.82*kw^2]

Để xác định định thức ma trận *det(F_k)* và các phần phụ đại số các phần tử $D_{i,j}$ của ma trận F_k của ma trận F_k ta có thể sử dụng gói Symbolic Math Toolbox của Matlab bằng lệnh "*det(F_k)*"

nếu kích thước của ma trận này đủ nhỏ (phụ thuộc vào số các phần tử khác không), kết quả thu được sẽ là công thức ở dạng biểu tượng trên màn hình, ví dụ:

ans=

$287104808668908863951708990 1.315*kw^{14}*r^3*s^4*z$ +…+

+ $2218063572479354423019604109325.3*kw^{16}*r^2*s^3*z$+ … +1288381536882496209041837 6

Cuối cùng, với mỗi giá trị của *kw*, sau khi tính được tất cả các phần phụ đại số $D_{i,j}$ của các phần tử *i, j* của ma trận F_k ở các hàng quan tâm, thay chúng vào phương trình (11) ta sẽ thu được phương trình, giả sử có dạng:

$$C_1 f^{15} r^2 z + ... + C_i z^7 s^4 .... + C_{p-2} s + C_{p-1} y + C_p r = C_0 . \qquad (13)$$

Ở phương trình (13) giả sử có *p* hằng số C_i ở vế trái tương ứng với mỗi giá trị của *kw*. Ở đây, để xác định các ẩn *r, f, z*…, ta cần *p* giá trị biên độ mô men xoắn đàn hồi tương ứng với *p* giá trị *kw* để lập thành *p* phương trình (13) và hệ phương trình tuyến tính *p* ẩn: x_1=r, x_2=f, x_3=z, …, $x_i=f^m r^n$…z,.. Giải hệ này sẽ xác định được các ẩn *r, f, s*…

Nếu việc tính toán thực hiện đúng và số số liệu đầu vào (giá trị mô men đàn hồi) có được từ tính toán như mô tả ở phần trên (giả thiết các hệ số sức cản là hằng), thì các kết quả phải trùng khớp, có nghĩa là:

nếu: $x_1=r$, $x_2=f$, $x_3=z$, $x_i=f^m r^n z$,

thì: $x_i=x_2^m x_1^n x_3$.

Khi chắc chắn tính toán đúng, lấy các giá trị mô men đàn hồi đo được thực tế thay vào (13) tính toán. Nếu, ví dụ: $x_i \neq x_2^m x_1^n x_3$, thì tùy theo mức độ sai lệch ta có thể đánh giá được sự phù hợp của các giả thiết về sức cản đã sử dụng. Tuy vậy, ở trường hợp ví dụ, không may là ta không thể sử dụng được trực tiếp lệnh "*det(F_k)*". Nó tạo lỗi, hoặc "Out of Memory" hoặc dòng công thức quá dài, ví dụ:

ans=

287104808668908863951708990 1.315*kw^14*r^3*s^4*z+……

… +.2218063572479354423019604109325.3*kw^16*r^2*s^3*z + 1288381536882496209041837 6...

Output truncated. Text exceeds maximum line length of 25.000 characters for Command Window display.

Để khắc phục vấn đề này, ta phải chia thủ tục tính toán thành một số bước sau:

- Bước 1: thay các số hạng ma trận ban đầu có dạng là tổng, tích các biến với các hệ số bằng một biểu tượng (cho công thức có dạng gọn hơn), ví dụ: I1=-I1*kw^2+K1; r= kw*r; I2=-Ic*kw^2+K1+Kx; z= kw*z; s=kw*s; I3=-Ic*kw^2+2*Kx; I4= -Ic*kw^2+2*Kx; I5= -Ic*kw^2+K3+Kx; I6= -I3*kw^2+K3+K4=I6; I7= -I4*kw^2+K4+K5; I8= -I5*kw^2+K5+K6; I9= -Ic*kw^2+K6. Thu được ma trận F_k dạng:

F_k=

[I1, -K1, 0, 0, 0,0,0,0,0,0,0,-r, r,0,0,0,0,0,0,0,0,0];

[-K1, I2,-Kx, 0,0,0,0,0,0,0,0, r,-z, s,0,0,0,0,0,0,0,0];

[0,-Kx, I3,-Kx,0,0,0,0,0,0,0,0,s,-f,s,0,0,0,0,0,0,0];

[0,0,-Kx,I4,-Kx,0,0,0,0,0,0,0,0,s,-f,s,0,0,0,0,0,0];

[0,0,0,-Kx,I4,-Kx,0,0,0,0,0,0,0,0,s,-f,s,0,0,0,0,0];
[0,0,0,0,-Kx,I4,-Kx,0,0,0,0,0,0,0,0,s,-f,s,0,0,0,0];
[0,0,0,0,0,-Kx,I5,-K3,0,0,0,0,0,0,0,0,s,-f,s,0,0,0];
[0,0,0,0,0,0,-K3,I6,-K4,0,0,0,0,0,0,0,0,s,-s,0,0,0];
[0,0,0,0,0,0,0,-K4,I7,-K5,0,0,0,0,0,0,0,0,0,0,0,0];
[0,0,0,0,0,0,0,0,-K5,I8,-K6,0,0,0,0,0,0,0,0,0,0,0];
[0,0,0,0,0,0,0,0,0,-K6,I9,0,0,0,0,0,0,0,0,0,0,-y];
[r,-r,0,0,0,0,0,0,0,0,0,I1,-K1,0,0,0,0,0,0,0,0,0];
[-r,z,-s,0,0,0,0,0,0,0,0,-K1,I2,-Kx,0,0,0,0,0,0,0,0];
[0,-s,f,-s,0,0,0,0,0,0,0,0,-Kx,I3,-Kx,0,0,0,0,0,0,0];
[0,0,-s,f,-s,0,0,0,0,0,0,0,0,-Kx,I4,-Kx,0,0,0,0,0,0];
[0,0,0,-s,f,-s,0,0,0,0,0,0,0,0,-Kx,I4,-Kx,0,0,0,0,0];
[0,0,0,0,-s,f,-s,0,0,0,0,0,0,0,0,-Kx,I4,-Kx,0,0,0,0];
[0,0,0,0,0,-s,f,-s,0,0,0,0,0,0,0,0,-Kx,I5,-K3,0,0,0];
[0,0,0,0,0,0,-s,s,0,0,0,0,0,0,0,0,0,-K3,I6,-K4,0,0];
[0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,-K4,I7,-K5,0];
[0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,-K5,I8,-K6];
[0,0,0,0,0,0,0,0,0,0,y,0,0,0,0,0,0,0,0,0,0,-K6,I9];

- Bước 2: tính định thức ma trận trong góc ¼ thứ nhất, cỡ nhỏ hơn, ví dụ:
  D1=det(F_k(1:6,1:6)), thu được kết quả:

D1 =

- I3*I4^3*K1^2 - I1*I4^3*Kx^2 + I1*I2*I3*I4^3 + I4^2*K1^2*Kx^2 - I1*I2*I4^2*Kx^2 + 2*I3*I4*K1^2*Kx^2 + 2*I1*I4*Kx^4 - 2*I1*I2*I3*I4*Kx^2 - K1^2*Kx^4 + I1*I2*Kx^4

- Bước 3: biến đổi ma trận F_k bằng phương pháp Gause, sao ma trận F^1_k(1:6,1:6) thành ma trận tam giác, thay ma trận còn lại bằng ma trận có các phần tử là các biểu tượng:

B(1:16,1:16) = F^1_k(7:22,7:22)

B có dạng:

B =

[b1_1, b1_2, 0,0,0,b1_6,b1_7,b1_8,b1_9,b1_10,b1_11,b1_12,b1_13,0,0,0]
[b2_1,b2_2,b2_3,0,0,0,0,0,0,0,0,b2_12,b2_13,0,0,0]
[0,b3_2, b3_3, b3_4,0,0,0,0,0,0,0,0,0,0,0,0]
[0,0, b4_3, b4_4, b4_5,0,0,0,0,0,0,0,0,0,0,0]
[0,0, 0, b5_4, b5_5,0,0,0,0, 0, 0, 0, 0, 0, 0, b5_16]
[b6_1,0, 0,0,0, b6_6, b6_7, b6_8, b6_9, b6_10, b6_11, b6_12, 0,0,0,0]
[b7_1,0,0,0,0,b7_6,b7_7,b7_8,b7_9,b7_10,b7_11,b7_12,0,0,0,0]
[b8_1,0,0,0,0,b8_6,b8_7,b8_8,b8_9,b8_10,b8_11,b8_12,0,0,0,0]
[b9_1,0,0,0,0,b9_6,b9_7,b9_8,b9_9,b9_10,b9_11,b9_12,0,0,0,0]
[b10_1,0,0,0,0,b10_6,b10_7,b10_8,b10_9,b10_10,b10_11,b10_12,0,0,0,0]
[b11_1,0,0,0,0, b11_6, b11_7, b11_8, b11_9, b11_10, b11_11, b11_12,0,0,0,0]
[b12_1,b12_2,0,0,0,b12_6,b12_7,b12_8,b12_9,b12_10,b12_11,b12_12,b12_13,0,0,0]

[b13_1,b13_2,0,0,0,0,0,0,0,0,0,b13_12,b13_13,b13_14,0,0]

[0,0,0,0,0,0,0,0,0,0,0,0,b14_13,b14_14, b14_15,0]

[0,0,0,0,0,0,0,0,0,0,0,0,0, b15_14, b15_15, b15_16]

[0,0,0,0, b16_5,0,0,0,0,0,0,0,0,0, b16_15, b16_16].

Trong đó, ví dụ:

b1_1=B(1,1,1:2)=F^1_k(7,7,1:2)=

= [I5*(Kx^2*(I4*(Kx^2-I2*I3)+I2*Kx^2)+I4*(Kx^2*(Kx^2-I2*I3)-I4*(I4*(Kx^2-I2*I3)+I2*Kx^2)))-Kx^2*(Kx^2*(Kx^2-I2*I3)-I4*(I4*(Kx^2-I2*I3)+I2*Kx^2)), I5*(I4*(I4*(K1^2*Kx^2-I3*I4*K1^2)+I3*K1^2*Kx^2)-Kx^2*(K1^2*Kx^2-I3*I4*K1^2))-Kx^2*(I4*(K1^2*Kx^2-I3*I4*K1^2)+I3*K1^2*Kx^2)];

v(1,1,1:2)=[I1,1];

Tiếp tục lắp lại các bước từ 1, 2, 3 cho đến khi biến đổi hết ma trận (22,22) ban đầu thành ma trận tam giác. Trong đó, sau một số lần biến đổi, khi công thức tính mỗi phần tử trở nên dài thì ta lại thay chúng bằng biểu tượng khác. Ví dụ:

C(1:11,1:11) =

[c1_1, c1_2,0,0, c1_5, c1_6, c1_7,0,0,0,0]

[c2_1, c2_2, c2_3,0, c2_5, c2_6, c2_7,0,0,0,0]

[0, c3_2, c3_3, c3_4,0, c3_6, c3_7,0,0,0,0]

[0,0, c4_3, c4_4, c4_5, c4_6, c4_7,0,0,0,0]

[c5_1, c5_5,0, c5_4, c5_5, c5_6, c5_7,0,0,0,0]

[c6_1, c6_2, c6_3, c6_4, c6_5, c6_6, c6_7,0,0,0,0]

[c7_1, c7_2, c7_3, c7_4, c7_5, c7_6, c7_7, c7_8,0,0,0]

[0,0,0,0,0,0, c8_7, c8_8, c8_9,0,0]

[0,0,0,0,0,0,0, c9_8, c9_9, c9_10,0]

[0,0,0,0,0,0,0,0, c10_9, c10_10, c10_11]

[0,0,0,0,0,0,0,0,0, c11_10, c11_11],

trong đó, ví dụ:

c1_1=C(1,1)=C^1 (1,1,1:5)=

= [b2_2*b6_6*(b5_5*(b4_4*(b2_2*b3_3-b2_3*b3_2)-b2_2*b3_4*b4_3)-b4_5*b5_4*(b2_2*b3_3-b2_3*b3_2))*(b2_2*b3_3-b2_3*b3_2)*(b4_4*(b2_2*b3_3-b2_3*b3_2)-b2_2*b3_4*b4_3),-(b5_5*(b4_4*(b2_2*b3_3-b2_3*b3_2)-b2_2*b3_4*b4_3)-b4_5*b5_4*(b2_2*b3_3-b2_3*b3_2))*(((b2_2*b3_3-b2_3*b3_2)*(b1_2*b2_1*b6_6+b1_6*b2_2*b6_1)+b1_2*b2_1*b2_2*b3_3*b6_6)*(b4_4*(b2_2*b3_3-b2_3*b3_2)-b2_2*b3_4*b4_3)+b2_2*b6_6*(b1_2*b2_1*b3_3*b4_4-b1_2*b2_1*b3_4*b4_3)*(b2_2*b3_3-b2_3*b3_2))-b2_2*b6_6*(b5_5*(b1_2*b2_1*b3_3*b4_4-b1_2*b2_1*b3_4*b4_3)-b1_2*b2_1*b3_3*b4_5*b5_4)*(b2_2*b3_3-b2_3*b3_2)*(b4_4*(b2_2*b3_3-b2_3*b3_2)-b2_2*b3_4*b4_3),(b5_5*(b4_4*(b2_2*b3_3-b2_3*b3_2)-b2_2*b3_4*b4_3)-b4_5*b5_4*(b2_2*b3_3-b2_3*b3_2))*((b1_2*b2_1*b3_3*b4_4-b1_2*b2_1*b3_4*b4_3)*((b2_2*b3_3-

```
b2_3*b3_2)*(b1_2*b2_1*b6_6+b1_6*b2_2*b6_1)+b1_2*b2_1*b2_2*b3_3*b6_6)+(b1_
2*b2_1*b3_3*(b1_2*b2_1*b6_6+b1_6*b2_2*b6_1)-
b1_2*b1_6*b2_1*b2_3*b3_2*b6_1)*(b4_4*(b2_2*b3_3-b2_3*b3_2)-
b2_2*b3_4*b4_3)-
b1_2*b1_6*b2_1*b2_2*b2_3*b3_2*b3_4*b4_3*b6_1)+(b5_5*(b1_2*b2_1*b3_3*b4_4-
b1_2*b2_1*b3_4*b4_3)-b1_2*b2_1*b3_3*b4_5*b5_4)*(((b2_2*b3_3-
b2_3*b3_2)*(b1_2*b2_1*b6_6+b1_6*b2_2*b6_1)+b1_2*b2_1*b2_2*b3_3*b6_6)*(b4_
4*(b2_2*b3_3-b2_3*b3_2)-b2_2*b3_4*b4_3)+b2_2*b6_6*(b1_2*b2_1*b3_3*b4_4-
b1_2*b2_1*b3_4*b4_3)*(b2_2*b3_3-b2_3*b3_2))-
b1_2*b1_6*b2_1*b2_2*b2_3*b3_2*b3_4*b4_3*b4_5*b5_4*b6_1*(b2_2*b3_3-
b2_3*b3_2),b1_2^2*b1_6*b2_1^2*b2_3*b3_2*b3_4*b4_3*b4_5*b5_4*b6_1*(b2_2*b3
_3-b2_3*b3_2)-(b5_5*(b4_4*(b2_2*b3_3-b2_3*b3_2)-b2_2*b3_4*b4_3)-
b4_5*b5_4*(b2_2*b3_3-b2_3*b3_2))*((b1_2*b2_1*b3_3*b4_4-
b1_2*b2_1*b3_4*b4_3)*(b1_2*b2_1*b3_3*(b1_2*b2_1*b6_6+b1_6*b2_2*b6_1)-
b1_2*b1_6*b2_1*b2_3*b3_2*b6_1)-
b1_2^2*b1_6*b2_1^2*b2_3*b3_2*b3_4*b4_3*b6_1)-(b5_5*(b1_2*b2_1*b3_3*b4_4-
b1_2*b2_1*b3_4*b4_3)-b1_2*b2_1*b3_3*b4_5*b5_4)*((b1_2*b2_1*b3_3*b4_4-
b1_2*b2_1*b3_4*b4_3)*((b2_2*b3_3-
b2_3*b3_2)*(b1_2*b2_1*b6_6+b1_6*b2_2*b6_1)+b1_2*b2_1*b2_2*b3_3*b6_6)+(b1_
2*b2_1*b3_3*(b1_2*b2_1*b6_6+b1_6*b2_2*b6_1)-
b1_2*b1_6*b2_1*b2_3*b3_2*b6_1)*(b4_4*(b2_2*b3_3-b2_3*b3_2)-
b2_2*b3_4*b4_3)-
b1_2*b1_6*b2_1*b2_2*b2_3*b3_2*b3_4*b4_3*b6_1)+b1_2^2*b1_6*b2_1^2*b2_2*b2
_3*b3_2*b3_3*b3_4*b4_3*b4_5*b5_4*b6_1,((b1_2*b2_1*b3_3*b4_4-
b1_2*b2_1*b3_4*b4_3)*(b1_2*b2_1*b3_3*(b1_2*b2_1*b6_6+b1_6*b2_2*b6_1)-
b1_2*b1_6*b2_1*b2_3*b3_2*b6_1)-
b1_2^2*b1_6*b2_1^2*b2_3*b3_2*b3_4*b4_3*b6_1)*(b5_5*(b1_2*b2_1*b3_3*b4_4-
b1_2*b2_1*b3_4*b4_3)-b1_2*b2_1*b3_3*b4_5*b5_4)-
b1_2^3*b1_6*b2_1^3*b2_3*b3_2*b3_3*b3_4*b4_3*b4_5*b5_4*b6_1];
v1(1,1,1:5)=[b1_1^5,b1_1^4,b1_1^3,b1_1^2,b1_1];
```

Và:

```
D =
[d1_1, d1_2,    0,    0,    0,    0]
[ d2_1, d2_2, d2_3,    0,    0,    0]
[    0, d3_2, d3_3, d3_4,    0,    0]
[    0,    0, d4_3, d4_4, d4_5,    0]
[    0,    0,    0, d5_4, d5_5, d5_6]
[    0,    0,    0,    0, d6_5, d6_6],
```

trong đó, ví dụ:

d1_1 (1:5)=[((c2_2*c3_3-c2_3*c3_2)*(c2_2*c4_4-c2_4*c4_2)-(c2_2*c3_4-c2_4*c3_2)
*(c2_2*c4_3-c2_3*c4_2))*((c2_2*c3_3-c2_3*c3_2)*(c2_2*c5_5-c2_5*c5_5)-
(c2_2*c3_5-c2_5*c3_2)*(c2_2*c5_3-c2_3*c5_5))-((c2_2*c3_3-c2_3*c3_2)
*(c2_2*c4_5-c2_5*c4_2)-(c2_2*c3_5-c2_5*c3_2)*(c2_2*c4_3-c2_3*c4_2))
*((c2_2*c3_3-c2_3*c3_2)*(c2_2*c5_4-c2_4*c5_5)-(c2_2*c3_4-c2_4*c3_2)*
(c2_2*c5_3-c2_3*c5_5)),((c2_2*c3_3-c2_3*c3_2)*(c2_2*c5_4-c2_4*c5_5)-
(c2_2*c3_4-c2_4*c3_2)*(c2_2*c5_3-c2_3*c5_5))*((c2_2*c4_5-c2_5*c4_2)*
(c1_2*c2_1*c3_3-c1_2*c2_3*c3_1-c1_3*c2_1*c3_2+c1_3*c2_2*c3_1)-
(c2_2*c4_3-c2_3*c4_2)*(c1_2*c2_1*c3_5-c1_2*c2_5*c3_1-c1_5*c2_1*c3_2
+c1_5*c2_2*c3_1)-(c2_2*c3_5-c2_5*c3_2) *(c1_2*c2_1*c4_3-
c1_2*c2_3*c4_1-c1_3*c2_1*c4_2+c1_3*c2_2*c4_1) +(c2_2*c3_3-
c2_3*c3_2)*(c1_2*c2_1*c4_5-c1_2*c2_5*c4_1-c1_5*c2_1*c4_2
+c1_5*c2_2*c4_1))-((c2_2*c3_3-c2_3*c3_2)*(c2_2*c5_5-c2_5*c5_5)-
(c2_2*c3_5-c2_5*c3_2)*(c2_2*c5_3-c2_3*c5_5))*((c2_2*c4_4-c2_4*c4_2)
*(c1_2*c2_1*c3_3-c1_2*c2_3*c3_1-c1_3*c2_1*c3_2+c1_3*c2_2*c3_1)-
(c2_2*c4_3-c2_3*c4_2)*(c1_2*c2_1*c3_4-c1_2*c2_4*c3_1-c1_4*c2_1*c3_2
+c1_4*c2_2*c3_1)-(c2_2*c3_4-c2_4*c3_2)*(c1_2*c2_1*c4_3-
c1_2*c2_3*c4_1 - c1_3*c2_1*c4_2+c1_3*c2_2*c4_1)+(c2_2*c3_3-
c2_3*c3_2)*(c1_2*c2_1*c4_4-c1_2*c2_4*c4_1-c1_4*c2_1*c4_2+
c1_4*c2_2*c4_1))+((c2_2*c3_3-c2_3*c3_2)*(c2_2*c4_5-c2_5*c4_2)-
(c2_2*c3_5-c2_5*c3_2)*(c2_2*c4_3-c2_3*c4_2))*((c2_2*c5_4-c2_4*c5_5)*
(c1_2*c2_1*c3_3-c1_2*c2_3*c3_1-c1_3*c2_1*c3_2 +c1_3*c2_2*c3_1)-
(c2_2*c5_3-c2_3*c5_5)*(c1_2*c2_1*c3_4-c1_2*c2_4*c3_1-c1_4*c2_1*c3_2
+c1_4*c2_2*c3_1)-(c2_2*c3_4-c2_4*c3_2) *(c1_2*c2_1*c5_3-
c1_2*c2_3*c5_1+c1_3*c2_2*c5_1-c1_3*c2_1*c5_5) +(c2_2*c3_3-
c2_3*c3_2)*(c1_2*c2_1*c5_4-c1_2*c2_4*c5_1 +c1_4*c2_2*c5_1-
c1_4*c2_1*c5_5))-((c2_2*c3_3-c2_3*c3_2)*(c2_2*c4_4-c2_4*c4_2)-
(c2_2*c3_4-c2_4*c3_2)*(c2_2*c4_3-c2_3*c4_2))*((c2_2*c5_5-c2_5*c5_5)*
(c1_2*c2_1*c3_3-c1_2*c2_3*c3_1-c1_3*c2_1*c3_2 +c1_3*c2_2*c3_1)-
(c2_2*c5_3-c2_3*c5_5) *(c1_2*c2_1*c3_5-c1_2*c2_5*c3_1 -
c1_5*c2_1*c3_2+c1_5*c2_2*c3_1)-(c2_2*c3_5-c2_5*c3_2)*
(c1_2*c2_1*c5_3-c1_2*c2_3*c5_1+c1_3*c2_2*c5_1-c1_3*c2_1*c5_5)+
(c2_2*c3_3-c2_3*c3_2)*(c1_2*c2_1*c5_5-c1_2*c2_5*c5_1+c1_5*c2_2*c5_1-
c1_5*c2_1*c5_5)),((c2_2*c3_3-c2_3*c3_2)*(c2_2*c5_5-c2_5*c5_5)-
(c2_2*c3_5-c2_5*c3_2)*(c2_2*c5_3-c2_3*c5_5))*((c1_2*c2_1*c3_3-
c1_2*c2_3*c3_1-c1_3*c2_1*c3_2 +c1_3*c2_2*c3_1)*(c1_2*c2_1*c4_4-
c1_2*c2_4*c4_1-c1_4*c2_1*c4_2 +c1_4*c2_2*c4_1)-(c1_2*c2_1*c3_4-
c1_2*c2_4*c3_1-c1_4*c2_1*c3_2 +c1_4*c2_2*c3_1)*(c1_2*c2_1*c4_3-
c1_2*c2_3*c4_1-c1_3*c2_1*c4_2 +c1_3*c2_2*c4_1))-((c2_2*c3_3-
c2_3*c3_2)*(c2_2*c5_4-c2_4*c5_5)-(c2_2*c3_4-c2_4*c3_2)*(c2_2*c5_3-

c2_3*c5_5))*((c1_2*c2_1*c3_3-c1_2*c2_3*c3_1-c1_3*c2_1*c3_2+
c1_3*c2_2*c3_1)*(c1_2*c2_1*c4_5-c1_2*c2_5*c4_1-c1_5*c2_1*c4_2+
c1_5*c2_2*c4_1)-(c1_2*c2_1*c3_5-c1_2*c2_5*c3_1-c1_5*c2_1*c3_2+
c1_5*c2_2*c3_1)*(c1_2*c2_1*c4_3-c1_2*c2_3*c4_1-c1_3*c2_1*c4_2+
c1_3*c2_2*c4_1))-((c1_2*c2_1*c3_3-c1_2*c2_3*c3_1-c1_3*c2_1*c3_2+
c1_3*c2_2*c3_1)*(c1_2*c2_1*c5_4-c1_2*c2_4*c5_1+c1_4*c2_2*c5_1-
c1_4*c2_1*c5_5)-(c1_2*c2_1*c3_4-c1_2*c2_4*c3_1-c1_4*c2_1*c3_2+
c1_4*c2_2*c3_1)*(c1_2*c2_1*c5_3-c1_2*c2_3*c5_1+c1_3*c2_2*c5_1-
c1_3*c2_1*c5_5))*((c2_2*c3_3-c2_3*c3_2)*(c2_2*c4_5-c2_5*c4_2)-
(c2_2*c3_5-c2_5*c3_2)*(c2_2*c4_3-c2_3*c4_2))+((c1_2*c2_1*c3_3-
c1_2*c2_3*c3_1-c1_3*c2_1*c3_2 +c1_3*c2_2*c3_1)*(c1_2*c2_1*c5_5-
c1_2*c2_5*c5_1+c1_5*c2_2*c5_1-c1_5*c2_1*c5_5)-(c1_2*c2_1*c3_5-
c1_2*c2_5*c3_1-c1_5*c2_1*c3_2 +c1_5*c2_2*c3_1)*(c1_2*c2_1*c5_3-
c1_2*c2_3*c5_1+c1_3*c2_2*c5_1-c1_3*c2_1*c5_5))*((c2_2*c3_3-
c2_3*c3_2)*(c2_2*c4_4-c2_4*c4_2)-(c2_2*c3_4-c2_4*c3_2)*(c2_2*c4_3-
c2_3*c4_2))+((c2_2*c4_4-c2_4*c4_2) *(c1_2*c2_1*c3_3-c1_2*c2_3*c3_1-
c1_3*c2_1*c3_2+c1_3*c2_2*c3_1)-(c2_2*c4_3-c2_3*c4_2)*
(c1_2*c2_1*c3_4-c1_2*c2_4*c3_1-c1_4*c2_1*c3_2 +c1_4*c2_2*c3_1)-
(c2_2*c3_4-c2_4*c3_2)*(c1_2*c2_1*c4_3-c1_2*c2_3*c4_1-c1_3*c2_1*c4_2
+c1_3*c2_2*c4_1)+(c2_2*c3_3-c2_3*c3_2)*(c1_2*c2_1*c4_4-
c1_2*c2_4*c4_1-c1_4*c2_1*c4_2 +c1_4*c2_2*c4_1))*((c2_2*c5_5-
c2_5*c5_5)*(c1_2*c2_1*c3_3-c1_2*c2_3*c3_1-c1_3*c2_1*c3_2+
c1_3*c2_2*c3_1)-(c2_2*c5_3-c2_3*c5_5)*(c1_2*c2_1*c3_5-c1_2*c2_5*c3_1-
c1_5*c2_1*c3_2 +c1_5*c2_2*c3_1)-(c2_2*c3_5-c2_5*c3_2)*
(c1_2*c2_1*c5_3-c1_2*c2_3*c5_1+c1_3*c2_2*c5_1-c1_3*c2_1*c5_5)+
(c2_2*c3_3-c2_3*c3_2) *(c1_2*c2_1*c5_5-c1_2*c2_5*c5_1+
c1_5*c2_2*c5_1-c1_5*c2_1*c5_5))-((c2_2*c4_5-c2_5*c4_2)*
(c1_2*c2_1*c3_3-c1_2*c2_3*c3_1-c1_3*c2_1*c3_2 +c1_3*c2_2*c3_1)-
(c2_2*c4_3-c2_3*c4_2)*(c1_2*c2_1*c3_5-c1_2*c2_5*c3_1-c1_5*c2_1*c3_2
+c1_5*c2_2*c3_1)-(c2_2*c3_5-c2_5*c3_2) *(c1_2*c2_1*c4_3-
c1_2*c2_3*c4_1-c1_3*c2_1*c4_2+c1_3*c2_2*c4_1) +(c2_2*c3_3-c2_3*c3_2)
*(c1_2*c2_1*c4_5-c1_2*c2_5*c4_1-c1_5*c2_1*c4_2 +c1_5*c2_2*c4_1))*
((c2_2*c5_4-c2_4*c5_5)*(c1_2*c2_1*c3_3-c1_2*c2_3*c3_1-
c1_3*c2_1*c3_2+c1_3*c2_2*c3_1)-(c2_2*c5_3-c2_3*c5_5)*
(c1_2*c2_1*c3_4-c1_2*c2_4*c3_1-c1_4*c2_1*c3_2+ c1_4*c2_2*c3_1)-
(c2_2*c3_4-c2_4*c3_2)*(c1_2*c2_1*c5_3-c1_2*c2_3*c5_1
+c1_3*c2_2*c5_1-c1_3*c2_1*c5_5)+(c2_2*c3_3-c2_3*c3_2)*
(c1_2*c2_1*c5_4-c1_2*c2_4*c5_1+c1_4*c2_2*c5_1-c1_4*c2_1*c5_5)),
((c1_2*c2_1*c3_3-c1_2*c2_3*c3_1-c1_3*c2_1*c3_2+c1_3*c2_2*c3_1)
*(c1_2*c2_1*c5_4-c1_2*c2_4*c5_1+c1_4*c2_2*c5_1-c1_4*c2_1*c5_5)-

(c1_2*c2_1*c3_4-c1_2*c2_4*c3_1-c1_4*c2_1*c3_2+c1_4*c2_2*c3_1)
*(c1_2*c2_1*c5_3-c1_2*c2_3*c5_1+c1_3*c2_2*c5_1-c1_3*c2_1*c5_5))
*((c2_2*c4_5-c2_5*c4_2)*(c1_2*c2_1*c3_3-c1_2*c2_3*c3_1-
c1_3*c2_1*c3_2+c1_3*c2_2*c3_1)-(c2_2*c4_3-c2_3*c4_2)*
(c1_2*c2_1*c3_5-c1_2*c2_5*c3_1-c1_5*c2_1*c3_2+c1_5*c2_2*c3_1)-
(c2_2*c3_5-c2_5*c3_2)*(c1_2*c2_1*c4_3-c1_2*c2_3*c4_1-c1_3*c2_1*c4_2
+c1_3*c2_2*c4_1)+(c2_2*c3_3-c2_3*c3_2)*(c1_2*c2_1*c4_5-
c1_2*c2_5*c4_1-c1_5*c2_1*c4_2+c1_5*c2_2*c4_1))-((c1_2*c2_1*c3_3-
c1_2*c2_3*c3_1-c1_3*c2_1*c3_2+c1_3*c2_2*c3_1)*(c1_2*c2_1*c5_5-
c1_2*c2_5*c5_1+c1_5*c2_2*c5_1-c1_5*c2_1*c5_5)-(c1_2*c2_1*c3_5-
c1_2*c2_5*c3_1-c1_5*c2_1*c3_2+c1_5*c2_2*c3_1)*(c1_2*c2_1*c5_3-
c1_2*c2_3*c5_1+c1_3*c2_2*c5_1-c1_3*c2_1*c5_5))*((c2_2*c4_4-
c2_4*c4_2)*(c1_2*c2_1*c3_3-c1_2*c2_3*c3_1-c1_3*c2_1*c3_2+
c1_3*c2_2*c3_1)-(c2_2*c4_3-c2_3*c4_2)*(c1_2*c2_1*c3_4-c1_2*c2_4*c3_1-
c1_4*c2_1*c3_2+c1_4*c2_2*c3_1)-(c2_2*c3_4-c2_4*c3_2)*
(c1_2*c2_1*c4_3-c1_2*c2_3*c4_1-c1_3*c2_1*c4_2+c1_3*c2_2*c4_1)+
(c2_2*c3_3-c2_3*c3_2)*(c1_2*c2_1*c4_4-c1_2*c2_4*c4_1-c1_4*c2_1*c4_2
+c1_4*c2_2*c4_1))+((c1_2*c2_1*c3_3-c1_2*c2_3*c3_1-c1_3*c2_1*c3_2+
c1_3*c2_2*c3_1)*(c1_2*c2_1*c4_5-c1_2*c2_5*c4_1-c1_5*c2_1*c4_2+
c1_5*c2_2*c4_1)-(c1_2*c2_1*c3_5-c1_2*c2_5*c3_1-c1_5*c2_1*c3_2+
c1_5*c2_2*c3_1)*(c1_2*c2_1*c4_3-c1_2*c2_3*c4_1-c1_3*c2_1*c4_2+
c1_3*c2_2*c4_1))*((c2_2*c5_4-c2_4*c5_5)*(c1_2*c2_1*c3_3-
c1_2*c2_3*c3_1-c1_3*c2_1*c3_2+c1_3*c2_2*c3_1)-(c2_2*c5_3-c2_3*c5_5)
*(c1_2*c2_1*c3_4-c1_2*c2_4*c3_1-c1_4*c2_1*c3_2+c1_4*c2_2*c3_1)-
(c2_2*c3_4-c2_4*c3_2)*(c1_2*c2_1*c5_3-c1_2*c2_3*c5_1+c1_3*c2_2*c5_1-
c1_3*c2_1*c5_5)+(c2_2*c3_3-c2_3*c3_2)*(c1_2*c2_1*c5_4-c1_2*c2_4*c5_1
+c1_4*c2_2*c5_1-c1_4*c2_1*c5_5))-((c1_2*c2_1*c3_3-c1_2*c2_3*c3_1-
c1_3*c2_1*c3_2+c1_3*c2_2*c3_1)*(c1_2*c2_1*c4_4-c1_2*c2_4*c4_1-
c1_4*c2_1*c4_2+c1_4*c2_2*c4_1)-(c1_2*c2_1*c3_4-c1_2*c2_4*c3_1-
c1_4*c2_1*c3_2+c1_4*c2_2*c3_1)*(c1_2*c2_1*c4_3-c1_2*c2_3*c4_1-
c1_3*c2_1*c4_2+c1_3*c2_2*c4_1))*((c2_2*c5_5-c2_5*c5_5)*
(c1_2*c2_1*c3_3-c1_2*c2_3*c3_1-c1_3*c2_1*c3_2+c1_3*c2_2*c3_1)-
(c2_2*c5_3-c2_3*c5_5)*(c1_2*c2_1*c3_5-c1_2*c2_5*c3_1-c1_5*c2_1*c3_2+
c1_5*c2_2*c3_1)-(c2_2*c3_5-c2_5*c3_2)*(c1_2*c2_1*c5_3-c1_2*c2_3*c5_1
+c1_3*c2_2*c5_1-c1_3*c2_1*c5_5)+(c2_2*c3_3-c2_3*c3_2)*
(c1_2*c2_1*c5_5-c1_2*c2_5*c5_1+c1_5*c2_2*c5_1-c1_5*c2_1*c5_5)),
((c1_2*c2_1*c3_3-c1_2*c2_3*c3_1-c1_3*c2_1*c3_2+c1_3*c2_2*c3_1)*
(c1_2*c2_1*c5_5-c1_2*c2_5*c5_1+c1_5*c2_2*c5_1-c1_5*c2_1*c5_5)-
(c1_2*c2_1*c3_5-c1_2*c2_5*c3_1-c1_5*c2_1*c3_2+c1_5*c2_2*c3_1)*
(c1_2*c2_1*c5_3-c1_2*c2_3*c5_1+c1_3*c2_2*c5_1-c1_3*c2_1*c5_5))*

```
((c1_2*c2_1*c3_3-c1_2*c2_3*c3_1-c1_3*c2_1*c3_2+c1_3*c2_2*c3_1)
*(c1_2*c2_1*c4_4-c1_2*c2_4*c4_1-c1_4*c2_1*c4_2+c1_4*c2_2*c4_1)-
(c1_2*c2_1*c3_4-c1_2*c2_4*c3_1-c1_4*c2_1*c3_2+c1_4*c2_2*c3_1)*
(c1_2*c2_1*c4_3-c1_2*c2_3*c4_1-c1_3*c2_1*c4_2+c1_3*c2_2*c4_1))-
((c1_2*c2_1*c3_3-c1_2*c2_3*c3_1-c1_3*c2_1*c3_2+c1_3*c2_2*c3_1)*
(c1_2*c2_1*c5_4-c1_2*c2_4*c5_1+c1_4*c2_2*c5_1-c1_4*c2_1*c5_5)-
(c1_2*c2_1*c3_4-c1_2*c2_4*c3_1-c1_4*c2_1*c3_2+c1_4*c2_2*c3_1)*
(c1_2*c2_1*c5_3-c1_2*c2_3*c5_1+c1_3*c2_2*c5_1-c1_3*c2_1*c5_5))*
((c1_2*c2_1*c3_3-c1_2*c2_3*c3_1-c1_3*c2_1*c3_2+c1_3*c2_2*c3_1)*
(c1_2*c2_1*c4_5-c1_2*c2_5*c4_1-c1_5*c2_1*c4_2+c1_5*c2_2*c4_1)-
(c1_2*c2_1*c3_5-c1_2*c2_5*c3_1-c1_5*c2_1*c3_2+c1_5*c2_2*c3_1)*
(c1_2*c2_1*c4_3-c1_2*c2_3*c4_1-c1_3*c2_1*c4_2+c1_3*c2_2*c4_1))];
v2(1,1,1:5)=[c1_1^8,c1_1^7,c1_1^6,c1_1^5,c1_1^4];
```

Tiếp tục phân tích các phần tử trên thành các số hạng bằng các lệnh '[$f_{i,j}$,v]=coeffs(d1_1,c1_1)'…, cuối cùng sẽ tính được các số hạng của định thức $|F_k|$. Sau mỗi lần thực hiện lệnh trên, kết quả được hiển thị trên màn hình, ta lại phải copy lại và soạn lại thành công thức theo đúng cú pháp của Matlab và tính tiếp. Do số lượng phép tính lớn, hiện nay tác giả mới đang thực hiện phân tích được tới lần thứ hai, và như thấy trình bày ở trên, các số hạng của các phần tử ma trận cuối cùng vẫn còn rất dài nên khả năng khi, giả sử khi tính d1_1 theo công thức cuối cùng, thay các biểu tượng Ci_j bằng các công thức ở bên trên nữa (Ci_j=f(bi_j) thì công thức thu được vượt quá chiều dài tối đa.

2.2. Kết luận

Bên trên đã trình bày mục đích, cơ sở lý thuyết và thuật toán xác định các hệ số sức cản xoắn. Tuy vậy, so với ví dụ đơn giản trước đây trình bày ở [5], ở hệ trục thực tế có số lượng khối lượng lớn nên thủ tục tính toán hơi dài dòng. Nhưng nếu, với dạng ma trận đã cho giống nhau với các hệ dao động xoắn tương đương, thì sau khi đã có công thức giải tích thì có thể sử dụng để tính cho các hệ khác với điều kiện có số khối lượng nhỏ hơn.

Do thủ tục tính dài, qua nhiều bước trung gian sử lý thủ công nên kết quả cuối cùng chưa đạt được, xong thời gian tới hi vọng sẽ có.

TÀI LIỆU THAM KHẢO

[1] Đặng Hộ. Thiết kế trang trí động lực tàu thủy (tập 2). Nhà xuất bản GTVT, 1986.

[2] Propulsion System Torsional Vibration Analysis B 170-V. Tài liệu về kết quả tính và đo dao động trình Đăng kiểm Đức của cơ quan thiết kế tàu Ba Lan.

[3] Nguyễn Mạnh Thường. *Tính nghiệm dao động xoắn cho hệ trục loạt tàu B 170* -V. TCGTVT số tháng 4-2011.

[4] Nguyễn Mạnh Thường. Nghiên cứu xây dựng phương pháp xác định các hệ số sức cản bằng thực nghiệm phục vụ tính toán dao động hệ trục tàu thủy. Hội thảo khoa học về đóng tàu, vận tải thủy, công nghiệp dầu khí biển và thiết bị, phương tiện giao thông cơ giới đường bộ, đường sắt. Nhà xuất bản Giao thông vận tải, Hà nội, tháng 4-2014, tr. 34-46.

[5] Nguyễn Mạnh Thường. The determining torsional vibration damping coefficients algorithm for computing marine shafting's vibrations. Tạp chí Khoa học-Công nghệ Hàng hải, số 42, 04-2015, tr 32-33, 36