A comparative study on the numerical solution algorithms of gas Reynolds equation

Yongliang Wang, Yi Wang, Longkai Zheng
Naval Architecture and Ocean Engineering College, Dalian Maritime University,
Dalian 116026, China

E-mail: wangyl@dlmu.edu.cn

Abstract. The improved P* algorithm is studied for solving the Reynolds equation by finite difference method. P* algorithm has the advantages of distinct mathematical concepts and simple programming. Firstly, the P* algorithm is combined with the Successive Over-Relaxation (SOR) method, considering the influence of the selection of relaxation factor on the calculation results. The optimal relaxation factor is determined to be 0.97, in this case, the minimum film thickness of flexible gas foil bearing can be calculated to 3.8 μm. Then, by changing the position of P*, the algorithm accuracy, sensitivity to the initial value, and computational efficiency of different P* algorithms are compared and studied. In the end, the P* algorithm is compared with the Newton-Raphson algorithm, it is found that although the Newton-Simpson algorithm computations the matrix dimension is large, but it has the advantages of fast computing speed and high efficiency.

Nomenclature

\( R \) radius of bearing (m)
\( L \) length of bearing (m)
\( C \) nominal radial clearance of the bearing (m)
\( P_a \) ambient pressure (Pa)
\( h \) film thickness (m)
\( \tilde{h} \) dimensionless film thickness
\( p \) film pressure (Pa)
\( \bar{P} \) dimensionless film pressure
\( z \) axial coordinate
\( \xi \) dimensionless axial coordinate
\( \theta \) circumferential coordinate
\( T \) temperature of lubrication (K)
\( T_0 \) temperature of ambient gas (K)
\( \bar{T} \) dimensionless temperature
\( \mu_a \) viscosity of ambient gas (N·s/m²)
\( \mu_l \) viscosity of lubrication (N·s/m²)
1. Introduction

The traditional blower and air compressor are driven by gears, and the mechanical friction in each gear drive produces a large energy consumption [1], moreover, the noise problem is prominent and the stability performance needs to be strengthened [2]. At the same time, whether the design of the rotor bearing system is reasonable is related to the safety and stability of the whole unit [3]. At this point, the choice of bearing is very important. Gas foil bearing is a kind of self-acting dynamic pressure gas bearing with foil as elastic surface. Compared with traditional gas bearing, foil bearing has the advantages of high speed, long working life, low temperature resistance, high temperature resistance, low energy loss, no pollution to the environment and excellent dynamic characteristics. It can be used in air compressor [4]，blower [5] and aerospace and so on [6].

In the process of fluid lubrication calculation by numerical analysis method [7], the solution of gas Reynolds equation [8] can be said to be always in a core position. For a long time, people have done a lot of exploration and research on how to solve the gas Reynolds equation efficiently, quickly and accurately. In the process of numerical iteration, the Reynolds equation will be solved once for each iteration, so the efficiency of solving the Reynolds equation algorithm directly determines the speed of the simulation platform. Among them, the finite difference method is a common algorithm for solving differential equations. The original differential equation is discretized into difference equations and solved. Many scholars have done a lot of research on the basis of the finite difference method.

Huapen Wu et al. [9] studied GMRES algorithm and the traditional SOR algorithm to solve the Reynolds equation in terms of speed and accuracy. Through comparison, they found that the pre-processed GMRES algorithm had obvious advantages. Zhang Dexuan et al. [10] proposed an iterative algorithm combining the golden section method to determine the relaxation factor with the improved iterative scheme of symmetric successive super-relaxation method, which has the advantage of high precision and fast convergence. Xu Shengli [11] used the multi-grid method introduced in this paper is an accelerated over-relaxation iterative method with shorter computation time, higher precision and faster convergence rate than finite difference method. Zhang Xi et al. [12] proposed a method for grid optimization based on the Reynolds equation difference operator based on the idea of grid hierarchy, and on this basis developed an adaptive regional refinement algorithm based on pressure gradient change.

\[ \bar{\mu}_g \text{ dimensionless air viscosity} \]
\[ \omega \text{ angular speed of shaft (rpm)} \]
\[ \omega_\gamma \text{ whirl speed (rpm)} \]
\[ t \text{ time (s)} \]
\[ \bar{t} \text{ dimensionless time} \]
\[ \gamma \text{ excitation frequency ratio ( } \omega_\gamma / \omega \text{ )} \]
\[ \Lambda_x \text{ bearing number} \]
\[ s \text{ bump pitch (m)} \]
\[ l \text{ half of bump length (m)} \]
\[ t_b \text{ bump foil thickness (m)} \]
\[ E_b \text{ young’s modulus of bump foil} \]
\[ v_b \text{ nominal radial clearance of the bearing} \]
\[ A_0 \text{ effective area that one bump covers} \]
\[ w_d \text{ foil deformation (m)} \]
\[ \bar{w}_d \text{ dimensionless foil deformation} \]
\[ k_{soa} \text{ bump stiffness (N·s/m}^3\text{)} \]
\[ \alpha \text{ bump foil compliance number} \]
\[ \varepsilon \text{ eccentricity ratio of bearing} \]
\[ \varphi \text{ attitude angle (degree)} \]
and regional integral value. Wang Qun [13] studied the effect of the number of grids on the accuracy of the algorithm and the effect of the number of grids on the efficiency of the algorithm. Although the solution method has been continuously improved and the convergence speed has been continuously improved, some algorithms have some difficulties in programming, especially in the process of modifying after changing boundary conditions. Based on the simplicity of programming, the P* algorithm is studied. The algorithm has the advantages of clear physical concept, easy programming and accurate calculation results. At the same time, the P* algorithm is combined with the dividing interactive method, which has a good convergence rate.

2. Theoretical analysis

2.1. General structure of gas foil journal bearing

Figure 1 shows the configuration of a first-generation bump type Gas Foil Bearing (GFB). Figure 2 shows the Cartesian coordinate system. The GFB consists of a top foil and a series of corrugated bumps (bump foil) to support the top foil. The leading edge of the top foil is free, and the foil trailing edge is fixed to the bearing sleeve.

![Schematic view of gas bump foil journal bearing](image1.png)

![Cartesian coordinate system](image2.png)

According to the bearing geometry shown in figure 2, assuming ideal gas for the working fluid, the dimensionless form of isothermal laminar Reynolds equation can be written as:

$$\frac{\partial}{\partial \theta} \left( \bar{\rho} h \frac{\partial \bar{P}}{\partial \theta} \right) + \left( \frac{2R}{L} \right)^2 \frac{\partial}{\partial \xi} \left( \bar{\rho} h \frac{\partial \bar{P}}{\partial \xi} \right) = \Lambda_s \frac{\partial (\bar{P} h)}{\partial \theta} + 2 \gamma \Lambda_s \frac{\partial (\bar{P} h)}{\partial t} \quad (1)$$

Where

$$\bar{\mu}_s = \frac{\mu_s}{\mu_0}, \quad \bar{T} = \frac{T}{T_0}, \quad \bar{P} = \frac{P}{P_a}, \quad \bar{h} = \frac{h}{C}, \quad \bar{t} = \nu t, \quad \xi = \frac{z}{0.5L}, \quad \gamma = \frac{\omega L}{\omega}, \quad \Lambda_s = \frac{6 \mu_0 \omega R^2}{C^2 P_a}$$

2.2. Film thickness equation

According to references [14], the thickness of the air film in a gas bearing can be formulated as:

$$\bar{h} = 1 + \varepsilon \cos(\theta - \varphi) \quad (2)$$

As for the foil bearing, the amount of foil deformation should be considered. So the foil bearing thickness should be written as:

$$\bar{h} = 1 + \varepsilon \cos(\theta - \varphi) + \frac{w_f}{C} \quad (3)$$
The bump foils are equivalent to linear spring supports, without considering the stiffness characteristics of top foils and the damping effect between top and bump foils. It also ignores the damping effect between bump foils and the inner wall of the bearing. The action relation between lubrication film pressure and foil deformation is given by \( p - p_a \) \( A_0 = k_{b0} Lw_0 \). Here, \( A_0 \) is an effective area that the bump covers, and the \( k_{b0} \) is the bump stiffness, which is given by \( k_{b0} = E_b \cdot t_b^3 / \left[ 2t^3 \left( 1 - v_b^2 \right) \right] \). The specific symbols are shown in figure 3 [15].

![Figure 3. Schematic diagram of bump foil and top foil structure [15].](image)

Converting it to dimensionless form can be written as \( \bar{h}_b = \alpha (\bar{P} - 1) \), \( \alpha \) is the bump foil compliance number given by \( \alpha = p_a \cdot s / (k_{b0} \cdot C) \). And dimensionless foil deformation is defined as \( \bar{w}_d = w_d / C \). The corresponding dimensionless film thickness expression is as follows [16]:

\[
\bar{h} = 1 + \varepsilon \cos (\theta - \varphi) + \alpha (\bar{P} - 1)
\]  

(4)

2.3. Grid generation
The meshing of the inner surface of the bearing is shown in figure 4 [17]. Since the bearing structure is symmetrical about the cross-section of the bearing, only half of the bearing is taken here for calculation. In figure 4, \( n_\theta \) is equally divided in the circumferential direction of the bearing and \( n_z \) is divided into the length of the axial half bearing. Therefore, there are \( n_\theta + 1 \) nodes in the circumferential direction and \( n_z + 1 \) nodes in the length direction of the half bearing. The solution domain of bearing is \((1: n_\theta + 1, 1: n_z + 1)\).

![Figure 4. Diagram of grid generation in computing domain [17].](image)

2.4. Boundary conditions
From the structure of figure 1, we can see that after the top foil is rolled into a cylinder, its head and tail are not connected, which means that the leading edge and trailing edge are not overlapped. At the same time, the top foil is supported by bump foil. Therefore, both the head and tail of the foil two sides of the
bearing edge are connected with the external environment, the film pressure of these parts is equal to the atmospheric pressure.

The boundary conditions of environmental pressure are as follows:

\[
P(1:n_y +1,0) = 1
\]

\[
P(1,0:n_z +1) = P(n_y +1,0:n_z +1) = 1
\]

As mentioned above, only the length of the half bearing is considered here. The symmetric boundary conditions are as follows:

\[
P(1:n_y +1, n_z -1) = P(1:n_y +1, n_z +1)
\]

\[
\frac{\partial P_{i,j}}{\partial \xi} = 0 (i = 1:n_y +1, j = n_z)
\]

2.5. Convergence criteria

When the result of the iteration reaches a certain precision, the iteration will terminate. The convergence conditions commonly used are as follow:

\[
\left( \sum_{i,j} \left( \frac{\Delta P_{i,j}}{P_{i,j}} \right)^2 \right)^{\frac{1}{2}} \leq 10^{-6}
\]

In this paper, finite element method and finite difference method are used to solve Reynolds equation and gas film thickness equation. And then we can get the static characteristic parameters such as film pressure distribution and film thickness distribution of gas foil bearing. In the following content, P* algorithm will be explained in detail, and different algorithms will be compared to it.

3. Algorithm P*

The method used in the reference [18] to solve The Reynolds equation is adopted by most researchers, namely the Newton-Raphson iterative method. This method is mainly to solve the linear equation \( Ax = B \).

In many studies, the method is considered to be improved. For example, the SOR iteration method is adopted in the reference [10], and the multi-grid computing method is adopted in the reference [11]. Although the solution speed is improved with continuous improvement, it is more complex in programming, especially after the boundary conditions are changed, the corresponding positions in the sparse matrix need to be revised again. Based on the simplicity of programming, the \( P^* \) algorithm is studied in this paper.

3.1. Algorithm \( P^* \) introductions

By expanding the partial differential items in equation (1), we can get the following results:

\[
\frac{\partial^2 P}{\partial \theta^2} + 3\frac{\partial^2 P}{\partial \theta \partial \xi} + P\frac{\partial^2 P}{\partial \xi^2} + \frac{2R}{L} \left( \frac{\partial \beta P}{\partial \xi} + \frac{3\beta P}{\partial \xi^3} \right) = \lambda_s \frac{\partial P}{\partial \theta}
\]

Considering the simple programming, the \( P^* \) algorithm is used for finite difference iteration. In equation (10), find outside the partial derivative on the left side of the equal, change it to \( P^* \) and define it as the initial pressure. Every time the calculation is completed, the value of \( P^* \) is updated, and then continues to participate in the next step of calculation in the form of a given value. After repeated iterative calculation, the convergence condition is finally satisfied and the convergence is realized. When \( P^* \) is set, the equation (10) becomes:
The equation (11) is reformulated with the central difference scheme as follow:

\[
\begin{align*}
\frac{\partial P}{\partial \theta} & = \frac{2R}{L} \left[ \frac{\partial P}{\partial \theta} \frac{\partial P}{\partial \xi} + P \left( \frac{\partial^2 P}{\partial \theta^2} \right) \right] \\
& = \Lambda_1 \frac{\partial P}{\partial \theta} + \Lambda_2 \frac{\partial P}{\partial \xi} \quad \text{(11)}
\end{align*}
\]

The equation (11) is reformulated with the central difference scheme as follow:

\[
\begin{align*}
\tilde{P}_{i,j} & = \frac{\partial P}{\partial \theta} \left[ \frac{\partial P}{\partial \theta} \frac{\partial P}{\partial \xi} + P \left( \frac{\partial^2 P}{\partial \theta^2} \right) \right] \\
& + \frac{\partial P}{\partial \xi} \left( \frac{\partial P}{\partial \theta} \frac{\partial P}{\partial \xi} + P \left( \frac{\partial^2 P}{\partial \theta \partial \xi} \right) \right) \\
& = \Lambda_1 \frac{\partial P}{\partial \theta} + \Lambda_2 \frac{\partial P}{\partial \xi} \quad \text{(12)}
\end{align*}
\]

Where \( i, j \) is the node number. After simplifying equation (12), we can get:

\[
\begin{align*}
\frac{\partial P}{\partial \theta} & = \frac{2R}{L} \left[ \frac{\partial P}{\partial \theta} \frac{\partial P}{\partial \xi} + P \left( \frac{\partial^2 P}{\partial \theta^2} \right) \right] \\
& = \Lambda_1 \frac{\partial P}{\partial \theta} + \Lambda_2 \frac{\partial P}{\partial \xi} \quad \text{(13)}
\end{align*}
\]

Where \( a_{i,j}, b_{i,j}, c_{i,j}, d_{i,j}, e_{i,j}, m_{i,j}, n_{i,j} \) are the coefficient matrixs as follows:

\[
\begin{align*}
a_{i,j} & = \frac{\partial P}{\partial \theta} \left[ \frac{\partial P}{\partial \theta} \frac{\partial P}{\partial \xi} + P \left( \frac{\partial^2 P}{\partial \theta^2} \right) \right] + \frac{\Lambda_1 \frac{\partial P}{\partial \theta}}{2 \Delta \theta} \\
b_{i,j} & = \frac{\partial P}{\partial \xi} \left[ \frac{\partial P}{\partial \theta} \frac{\partial P}{\partial \xi} + P \left( \frac{\partial^2 P}{\partial \theta \partial \xi} \right) \right] - \frac{\Lambda_1 \frac{\partial P}{\partial \theta}}{2 \Delta \theta} \\
c_{i,j} & = -2 \frac{\partial P}{\partial \xi} \left[ \frac{\partial P}{\partial \theta} \frac{\partial P}{\partial \xi} + P \left( \frac{\partial^2 P}{\partial \theta \partial \xi} \right) \right] - \frac{\Lambda_1 \frac{\partial P}{\partial \theta}}{2 \Delta \theta} \\
d_{i,j} & = \frac{2R}{L} \left[ \frac{\partial P}{\partial \theta} \frac{\partial P}{\partial \xi} + P \left( \frac{\partial^2 P}{\partial \theta \partial \xi} \right) \right] - \frac{\Lambda_1 \frac{\partial P}{\partial \theta}}{2 \Delta \theta} \\
e_{i,j} & = \frac{2R}{L} \left[ \frac{\partial P}{\partial \theta} \frac{\partial P}{\partial \xi} + P \left( \frac{\partial^2 P}{\partial \theta \partial \xi} \right) \right] - \frac{\Lambda_1 \frac{\partial P}{\partial \theta}}{2 \Delta \theta} \\
m_{i,j} & = \frac{\partial P}{\partial \theta} \frac{\partial P}{\partial \xi} + P \left( \frac{\partial^2 P}{\partial \theta \partial \xi} \right) - \frac{\Lambda_1 \frac{\partial P}{\partial \theta}}{2 \Delta \theta} \\
n_{i,j} & = \frac{2R}{L} \left[ \frac{\partial P}{\partial \theta} \frac{\partial P}{\partial \xi} + P \left( \frac{\partial^2 P}{\partial \theta \partial \xi} \right) \right] - \frac{\Lambda_1 \frac{\partial P}{\partial \theta}}{2 \Delta \theta} \quad \text{(14) - (20)}
\end{align*}
\]

3.2. The accuracy of P* algorithm

Firstly, we need to compare the results of the P* algorithm and the Newton-Raphson algorithm. The parameters in table 1 are used for calculation, and \( \varepsilon = 0.8 \). Shown in figure 5 is the dimensionless pressure for the foil bearing running at different speeds. The corresponding film thickness for a foil bearing is also plotted for comparison as figure 6. From the comparison, it can be found that the calculation results of the two algorithms are consistent, thus proving the accuracy of the P* algorithm.
The parameters in reference [19] are selected for calculation. Shown in figure 7 is the dimensionless pressure for the rigid bearing running at 30 krpm with an assumed minimum film thickness of about 16 μm. The corresponding film thickness for a rigid bearing is also plotted for comparison as figure 8. The numerical analysis results calculated by the \( P^* \) algorithm is compared with the curve data in reference [19]. For the gas film pressure, the calculated results of this model are basically consistent with the data obtained in references. The maximum error occurs at the maximum film pressure, and the error value is less than 5%. As can be seen from figure 8 that the film thickness distribution of gas bearing with rigid surface coincides with the reference data completely. This is because the theoretical model of rigid surface bearing is simple and does not consider the influence of deformation and other factors. First of all, we need to compare the results of the \( P^* \) algorithm and the Newton-Raphson algorithm.

![Figure 5](image5.png)  ![Figure 6](image6.png)

**Figure 5.** Dimensionless pressure at the axial midsection of foil bearing compared with Newton-Raphson algorithm.

**Figure 6.** Dimensionless film thickness at the axial midsection of foil bearing compared with Newton-Raphson algorithm.

Table 1. The bearing parameters [18]

| Foil bearing parameters                  | Values                  |
|------------------------------------------|-------------------------|
| Shaft Radius, \( R \) (m)                | 19.05×10^{-5}           |
| Bearing Length, \( L \) (m)              | 38.10×10^{-5}           |
| Nominal clearance, \( C \) (m)           | 3.18×10^{-5}            |
| Top foil thickness, \( t_p \) (m)        | 76.2×10^{-6}            |
| Bump foil thickness, \( t_b \) (m)       | 101.6×10^{-6}           |
| Bump Length, \( h_b \) (m)              | 0.63×10^{-3}            |
| Bump pitch, \( s \) (m)                 | 4.572×10^{-3}           |
| Bump length, \( 2l \) (m)               | 3.556×10^{-6}           |
| Young’s Modulus of bump foil, \( E \) (Pa)| 214×10^{9}              |
| Poisson's ratio, \( \nu \)              | 0.29                    |
| Viscosity, \( \mu \) (pa·s)             | 1.932×10^{-5}           |
| Density, \( \rho \) (kg/m^3)            | 1.1614                  |
The Reynolds equation of dimensionless steady gas is expressed as follows:

$$\frac{\partial}{\partial \theta} \left( \overline{Ph} \frac{\partial \overline{P}}{\partial \theta} \right) + \left( \frac{2R}{L} \right)^2 \frac{\partial}{\partial \zeta} \left( \overline{Ph} \frac{\partial \overline{P}}{\partial \zeta} \right) = \Lambda \frac{\partial (\overline{Ph})}{\partial \theta}$$  \hspace{1cm} (21)

Expand the sign of partial derivative. For the convenience of the following description, each item is numbered in the form shown in equation (22).
In the term 3 after the difference into \[ \frac{\partial \overline{P}}{\partial \theta} \] , in addition to the changes of algorithm P*, item 5 into \[ L L \rightarrow \xi \xi \]. Algorithm P*, item 2 into \[ \xi \xi \rightarrow \xi \xi \]. In addition to the changes of algorithm P*, \[ (h P h P R L h P P R) \]. Compared with the first two algorithms, algorithm P* changes item 6 after the change, which is change item 8 into \[ \Lambda_f \overline{P} \cdot (\partial \overline{h} / \partial \theta) \]. P* is the method introduced in the previous section. Compared with algorithm P*, P* put P* on the right end of the equal sign.

On the basis of algorithm P*, algorithm P* changes item 1 into \[ \overline{h} \cdot (\partial \overline{P} / \partial \theta) (\partial \overline{P} / \partial \theta) \] , item 4 into \[ \Lambda_f \overline{P} \cdot (\partial \overline{h} / \partial \theta) \]. In addition to the changes of algorithm P*, P* also change item 8 to \[ \Lambda_f \overline{P} \cdot (\partial \overline{h} / \partial \theta) \]. Compared with the first two algorithms, algorithm P* and algorithm P* set P* before the partial derivative symbol is opened. In this way, when the partial derivative symbol is opened, a partial derivative term for P* will be generated.

\[ h \frac{\partial P}{\partial \theta} + 3 \overline{h} \overline{P} \frac{\partial h}{\partial \theta} + \frac{\partial^2 P}{\partial \theta^2} + \overline{h} \left( \frac{2R}{L} \right) \frac{\partial P}{\partial \xi} + 3 \overline{h} \left( \frac{2R}{L} \right) \frac{\partial \overline{h}}{\partial \xi} \]

\[ = \Lambda_f \overline{P} \cdot (\partial \overline{h} / \partial \theta) + \Lambda_f \frac{\partial \overline{h}}{\partial \theta} \]  

On the basis of equation (22), algorithm P* changes item 3 into \[ \overline{P} \overline{h} \cdot (\partial \overline{P} / \partial \theta^2) \] , item 2 into \[ 3 \overline{h} \overline{P} \cdot (\partial \overline{h} / \partial \theta)(\partial \overline{P} / \partial \theta) \] , and item 5 into \[ 3 \overline{h} \overline{P} \left( \frac{2R}{L} \right)^2 \cdot (\partial \overline{h} / \partial \xi)(\partial \overline{P} / \partial \xi) \] , at the same time changes item 6 into \[ \overline{P} \overline{h} \left( \frac{2R}{L} \right)^2 \cdot (\partial \overline{P} / \partial \xi^2) \]. In addition to the changes of algorithm P*, P* also adds another change, which is change item 8 into \[ \Lambda_f \overline{P} \cdot (\partial \overline{h} / \partial \theta) \]. P* is the method introduced in the previous section. Compared with algorithm P*, P* put P* on the right end of the equal sign.

On the basis of algorithm P*, algorithm P* changes item 1 into \[ \overline{h} \cdot (\partial \overline{P} / \partial \theta) (\partial \overline{P} / \partial \theta) \] , item 4 into \[ \Lambda_f \overline{P} \cdot (\partial \overline{h} / \partial \theta) \]. In addition to the changes of algorithm P*, P* also change item 8 to \[ \Lambda_f \overline{P} \cdot (\partial \overline{h} / \partial \theta) \]. Compared with the first two algorithms, algorithm P* and algorithm P* set P* before the partial derivative symbol is opened. In this way, when the partial derivative symbol is opened, a partial derivative term for P* will be generated.

Figure 11. Dimensionless pressure at the axial midsection of different algorithms.

Figure 12. Dimensionless film thickness at the axial midsection of different algorithms.
is whether \( P^* \) is set at the right end of the equal sign. However, some serious things have been found in the numerical calculation by programming. When \( P \) is set as \( P^* \) in the second-order partial derivative after difference, the calculation cannot achieve convergence. Maybe try to change the iteration method can achieve convergence, but now the two algorithms have not found a solution.

4. Algorithm efficiency comparison

And until now, we can know that the \( P^* \) algorithm can solve the Reynolds equation accurately. In addition, we also need to pay attention to the efficiency of \( P^* \) algorithm. Then in the section 4, the efficiency of the \( P^* \) algorithm and Newton-Raphson algorithm will be compared.

4.1. Compare the Dimension of The Matrix

Figure 13 shows the \( P^* \) algorithm involved in calculating the matrix dimension, and figure 14 shows the Newton-Raphson algorithm. In the calculation below, set \( n_\theta=151, n_z=26 \). So, the number of grids participating in the calculation is 150×25. For the \( P^* \) algorithm, the size of the matrix involved in the calculation is 150×25. As for the Newton-Raphson algorithm, the method of solving the linear equation \( Ax=B \) is used. Therefore, the parameter values on each grid node need to be placed at the corresponding position, then \( A \) will become a large sparse matrix, and the number of diagonal elements is the number of grid nodes, which should be 150×25. And the matrix size should be (150×25,150×25). Intuitively, the size of the Newton-Raphson algorithm is the square of the \( P^* \) algorithm.

![Figure 13. Calculation matrix dimension of \( P^* \) algorithm.](image1)

![Figure 14. Calculation matrix dimension of Newton-Raphson algorithm.](image2)

4.2. Selection of Relaxation Factor

4.2.1. Algorithm \( P^* \). \( P^* \) algorithm introduces the Successive Over-Relaxation method (SOR)[20-23] at the solution of gas film thickness. \( H^{k+1,0} \) is the initial estimate of the \( k+1 \) cycle, \( H^{k,0} \) is the initial value of the last cycle, and \( H^k \) is the calculated result of the last cycle. The relationship between the above values is as follow: \( H^{k+1,0} = H^k + SOR \cdot (H^k - H^{k,0}) \). It should be noted that the selection of the relaxation factor is very important. If the selection is appropriate, the calculation efficiency can be greatly improved. Otherwise, iterative divergence may occur. In the calculation of flexible gas foil bearing, it is generally considered that the minimum gas film thickness is 4 microns, which is the closest to the engineering practice. In the case of the calculation parameters selected in this paper, if \( h_{\text{min}} \) of about 4 \( \mu \text{m} \), the eccentricity ratio should reach about 2.5.
Therefore, the calculation of the relaxation factor is within the range of 0.5 to 1, and the region for normal calculation can be realized. The solution result is shown in figure 15. The x-coordinate is the eccentricity rate, the y-coordinate is the calculation time, and each curve represents the value of a relaxation factor. As can be seen from the figure 15, when the eccentricity ratio is less than 1, the relaxation factor of 0.5 can realize the normal calculation of the whole region, and the calculation speed is the fastest and the efficiency is the highest. However, when the eccentricity ratio is greater than 1, choosing a smaller relaxation factor will lead to iterative divergence. After a lot of calculations, it can be seen that if the relaxation factor is 0.97, the maximum eccentricity can be calculated to 2.5, and the minimum film thickness is 3.81 μm.

4.2.2. Newton-Raphson algorithm. As for the Newton-Raphson algorithm, its general rule is the same as the P* algorithm, there is a calculation field that can do the normal calculation, but the difference is that the boundary line of the Newton-Raphson algorithm moves to the right. That is to say, with the same relaxation factor selected, the maximum eccentricity that can be calculated is larger than the P* algorithm. However, it will bring a long calculation time, and in some cases, it can be found that the convergence process is a run-out process by detecting the residual value.

Figure 15. Influence of relaxation factor selection on computational efficiency.

4.2.2. Newton-Raphson algorithm. As for the Newton-Raphson algorithm, its general rule is the same as the P* algorithm, there is a calculation field that can do the normal calculation, but the difference is that the boundary line of the Newton-Raphson algorithm moves to the right. That is to say, with the same relaxation factor selected, the maximum eccentricity that can be calculated is larger than the P* algorithm. However, it will bring a long calculation time, and in some cases, it can be found that the convergence process is a run-out process by detecting the residual value.

Figure 16. Convergent residual graph with $\varepsilon=1.3$, $SOR=0.55$ of Newton-Raphson algorithm.

Figure 17. Convergent residual graph with $\varepsilon=1.12$, $SOR=0.90$ of Newton-Raphson algorithm.
For example, select $SOR = 0.55$. The $P^*$ algorithm can calculate normally until $\varepsilon = 1.12$, and the calculation time is less than 7 seconds. The Newton-Raphson algorithm can calculate the eccentricity to 1.30, the calculation time is 46 seconds, and the iteration steps are 1074 steps. The residual curve starting from the second step is shown in figure 16.

At the same time, if we choose a better segmentation factor and initial value, the convergence effect will be very good. For example, in figure 17, the $SOR = 0.90$, and the initial value is 1.1 times the atmospheric pressure. However, it is worth mentioning that in order to adapt to the calculation of all eccentricity ranges, we have made a lot of attempts, and finally choose the initial value of 10 times atmospheric pressure and the $SOR = 0.97$.

4.3. Efficiency of different algorithms

In order to compare the calculation time of different algorithms, the same parameters were used for calculation. In this case, the selected parameters are: $\varepsilon = 0.8$ and $SOR = 0.98$. Firstly, the computation time of different algorithms under different initial values are calculated, and then fit the data points, as shown in figure 18. It is worth mentioning that the selection of the initial value has an important impact on whether the normal iterative calculation can be carried out. Attention should be paid to avoid geometric interference caused by too small initial pressure, that is, to ensure that the gas film thickness value is positive.

It can be seen from the figure 18 that although the coefficient expressions of algorithms 1 and 2 are different, their calculation time is almost the same. Algorithms 3 and 4 are the same too. From the comparison of computing time, it is obvious that the efficiency of the first two algorithms is better than that of the latter two algorithms. And the Newton-Raphson algorithm is the fastest.

From the initial value selection, the latter two $P^*$ algorithms are more sensitive to the initial value selection. In contrast, the initial value selection has little effect on the Newton-Raphson algorithm. At the same time, if the four algorithms choose the initial value closer to the calculation results, the speed can be improved. However, when the initial value of iteration pressure is greater than 10 times atmospheric pressure, the effect of initial value increase on calculation efficiency is not obvious. At the same time, if the initial value is too large, it is likely to cause abnormal calculation.

Figure 18. Calculation time of four algorithms under different initial values.
5. Conclusions
In the process of combining P* algorithm with SOR method, the choice of relaxation factor has an important influence on the normal calculation. Under the parameters selected in this paper, when the relaxation factor is 0.5, the calculation speed is fast and the efficiency is high. However, the calculation scope is limited, and the iteration divergence will occur when the eccentricity ratio is greater than 1. When the relaxation factor is 0.97, the calculation speed is slow, but the normal calculation range is wide, and the eccentricity can be calculated to 2.5. At this time, the minimum film thickness is 3.8 μm, which is close to the actual engineering value.

This paper studies different algorithms after changing P* position. From the perspective of calculation accuracy, the maximum relative error of the four algorithms is 0.39%, and the calculation results are consistent. From the perspective of computational efficiency, whether P* is placed on the right side of the equal sign has no influence on the computational speed. The calculation speed of the first two algorithms is better than that of the other two algorithms, and the calculation speed can be increased by about 1.5 times under the same parameter. From the perspective of initial value influence, the latter two P* algorithms are more sensitive to initial value selection. However, when the initial iteration pressure is greater than 10 atmospheres, the sensitivity of the four algorithms to the initial pressure decreases.

Compared with the Newton-Raphson algorithm, the P* algorithm has a distinct mathematical concept and simple programming process. The P* algorithm adopts the form of matrix operation and combines with the SOR method in the calculation process, which can greatly improve the calculation speed. In terms of computing the dimension of the matrix, the matrix dimension of Newton-Simpson algorithm is the square of P* algorithm. Intuitively, the P* algorithm has the advantage of a small matrix dimension. Although the Newton-Raphson algorithm has large dimension, the process of setting initial value for iteration is not involved in the calculation. In terms of computational efficiency, the Newton-Raphson algorithm is superior to the P* algorithm.

Reference
[1] Ju L, Zhang J X, Xiao B and Miao Z 2020 The Analysis and Treatment of HSRL350 Fan Reduction Gearbox[J] Chinese Journal of Turbomachinery 62 59-61
[2] Allaire P Dimond T and Cao J M 2019 Bearing Fix for Large Sub-synchronous Vibration in Steam Turbine-gearbox-generator System[J] Chinese Journal of Turbomachinery 61 4 34-38+4
[3] Jiang L J, Yuan J B, Huo W H and Sun H 2020 Dynamic Characteristics Analysis of Cantilever Rotor Considering Impeller Coupling Effect[J] Chinese Journal of Turbomachinery 62 16-20
[4] Deng Z H 2016 Design and Experimental study of an oil-free high speed Compressor for Fuel Cell[D] Hunan University
[5] Shu X J, Xu G, Zheng Y Q, Cui H L and Lan H 2017 Key Technologies for Centrifugal Blowers with Foil Bearings and High Speed Magnet Motor[J] Chinese Journal of Turbomachinery 59 36-42
[6] Zhu B S, Zhang B Z, Rao J Yuan T Z, Niu S T and Lu S C 2020 The Application of Air Bearing in Blower and Turbojet Engine Field[J] Chinese Journal of Turbomachinery 62 25-28
[7] Tian Q Y, Huang W J, Rao J, Zhu X N and Wang X Z 2018 Numerical Calculation and Analysis of Oil Film Characteristics of Hydrodynamic Radial Sliding Bearing[J] Chinese Journal of Turbomachinery 60 44-47
[8] Wang X K, Xu Q, Zhang L X and Yang H 2019 Numerical Analysis of the Static Performance of Aerostatic Journal Bearings[J] Machinery Design & Manufacture 339 14-17
[9] Wu H P, Kong X M and Chen Dong 2000 The Application of GMRES Algorithm in Calculating Reynolds Equation[J] Lubrication Engineering 000 2-4
[10] Zhang D X, Wu G W and Deng H Y 2010 Determination of The Best Relaxation Factor of Symmetric Successive Over Relaxation Iterative Method by Golden Ratio Method[J] Science and Technology Innovation Herald 169 217-219
[11] Xu S L and Fan W X 2013 Multi-grid Method for Solving Reynolds Equation[J] Mechanical Engineering & Automation 177 70-71
[12] Zhang X, Luo H X, Zuo B Q and Li Z Q 2019 Research on Adaptive Step-size Difference Algorithm for Reynolds Equation Based on Hierarchical Model[J] Lubrication Engineering 44 123-130
[13] Wang Q 2020 Influence of Mesh Number on Solving Reynolds Equation by FDM[J] Lubrication Engineering 45 125-129
[14] Heshmat H, Walowit J.A and Pinkus O 1983 Analysis of Gas Lubricated Foil Journal Bearings ASME Journal of Lubrication Technology 105 647–655
[15] Xu F C and Kim D J 2016 Dynamic performance of foil bearings with a quadratic stiffness model[J] Neurocomputing 216 666-671
[16] Peng Z C and Khonsari M M 2004 On the Limiting Load-Carrying Capacity of Foil Bearings[J] Journal of Tribology 126 817-818
[17] Yan J J 2016 Rotor Dynamic Characteristics of Foil Bearing System with Surface Micro Structure[D] Harbin Institute of Technology
[18] Xu F C 2016 Vibration Characteristics of Gas Foil Journal Bearing-Rotor System Based on Effects of Coulomb Friction[D] Harbin Institute of Technology
[19] Peng Z C and Khonsari M M 2006 A Thermohydrodynamic Analysis of Foil Journal Bearings[J] Journal of Tribology 128 534-541
[20] Liu S M and Liu H M 2016 A Method for Simplifying the Process of Solving Reynolds Equation[J] Journal of North China University (Natural Science Edition) 27 165-167
[21] Lu P 2016 Numerical Study of Static Characteristics of the Foil Hydrodynamic Thrust Gas Bearing with Hemispherical Convex Dots in Turboexpander[D] Xi'an University of Architecture and Technology
[22] Wu B Y, Jin Jian and Zhang G X 2002 Solving Reynolds Equation with SOR Method[J] Journal of Shanghai University(Natural Science) 8 130-132,155
[23] Hu F and Yu F X 2006 The Selection Method of the Relaxation Factor ω In Successive Over-Relaxation Iteration Algorithm[J] Journal of Qinghai Normal University (Natural Science) 01 42-45