PARALLEL DOMAIN DECOMPOSITION SOLVER FOR FLOWS IN HYDROSTATIC BEARINGS

M. Hanek\textsuperscript{1,3}, J. Šístek\textsuperscript{2,3}, P. Burda\textsuperscript{1}, E. Stach\textsuperscript{1}

\textsuperscript{1} Faculty of Mechanical Engineering, Czech Technical University in Prague, Karlovo náměstí 13, CZ - 121 35 Prague 2, Czech Republic
\textsuperscript{2} School of Mathematics, The University of Manchester, M13 9PL, Manchester, United Kingdom
\textsuperscript{3} Institute of Mathematics of the Czech Academy of Sciences, Žitná 25, CZ - 115 67, Prague 1, Czech Republic

Abstract

We perform simulations of oil flow in hydrostatic bearings. Stationary incompressible three-dimensional flow governed by the Navier-Stokes equations is considered. The finite element method is used for discretization. The arising nonlinear system of algebraic equations is linearized using the Picard’s iteration, and the Balancing Domain Decomposition based on Constraints (BDDC) method is used to solve the linear systems of equations. The solver is first validated with an experiment for the case of a bearing without motion, and it is then applied to simulation of flow in a sliding bearing.

Keywords: Hydrostatic bearings, Navier-Stokes equations, Finite element method, Balancing Domain Decomposition based on Constraints, BDDC.

1 Introduction

We deal with numerical simulation of oil flow inside a hydrostatic bearing. These are parts of machine tools that keep moving parts of the machines on a thin layer of oil to provide low friction. The layer is called a throttling gap and it is only a few tens of micrometers high while the other dimensions of the domain where the oil flows (called a hydrostatic cell) are typically at the order of millimeters, see Figure 1. The oil is pumped to the domain through the top face, and it flows out through the throttling gap where the pressure drops from several MPa to the atmospheric value.

Our main goal is to simulate oil flow in such bearings during their sliding. This is a challenging problem and, up to our knowledge, such type of 3-D simulation has not been considered in literature. The main difficulty is to perform simulations with real-scale height of the throttling gap, where elements with high aspect ratio inevitably occur (see Figure 3).

The oil flow is governed by the incompressible steady 3-D Navier-Stokes equations. The finite element method (FEM) is used for discretization of the problem, and the system of nonlinear algebraic equations is linearized by the Picard’s iteration. In our calculations, we apply one step of the Balancing Domain Decomposition based on Constraints (BDDC) method as a preconditioner for the sequence of linear systems of equations. The BDDC method was first introduced in [1], and we use the approach to nonsymmetric problems described in [2]. During our previous research, using this method with a partitioner tailored to the problem has allowed us to perform such simulations [3].

In Section 2, we introduce the mathematical model and briefly recall the application of finite element method with the BDDC preconditioner. Section 3 is devoted to the definition of the problem and its numerical results, where two kinds of problems are considered. The first problem is without motion of the bearing, and it has been used for validation of the solver by an experiment. The second problem is a simulation of the flow inside a sliding bearing.

2 Finite element method for the Navier-Stokes equations and the BDDC preconditioner

In this section, we introduce the mathematical model and its numerical approximation. In our calculations, we consider stationary incompressible flow in three-dimensional domain $\Omega$ governed
by the Navier-Stokes equations with zero body forces (see e.g. [4]),

\[
(\mathbf{u} \cdot \nabla)\mathbf{u} - \nu \Delta \mathbf{u} + \nabla p = 0 \quad \text{in } \Omega,
\]

\[
\nabla \cdot \mathbf{u} = 0 \quad \text{in } \Omega,
\]

where \( \mathbf{u} = (u_x, u_y, u_z)^T \) is an unknown velocity vector, \( p \) is an unknown pressure normalized by (constant) density, \( \nu \) is a given kinematic viscosity, and \( \Omega \) is the solution domain.

Next we add the following boundary conditions,

\[
\mathbf{u} = \mathbf{g} \quad \text{on } \Gamma_D,
\]

\[
-\nu(\nabla \mathbf{u}) \mathbf{n} + p \mathbf{n} = 0 \quad \text{on } \Gamma_N,
\]

where \( \Gamma_D \) and \( \Gamma_N \) are parts of the boundary \( \partial \Omega \), \( \Gamma_D \cup \Gamma_N = \partial \Omega \), \( \Gamma_D \cap \Gamma_N = \emptyset \), \( \mathbf{n} \) is the outer unit normal vector of the boundary, and \( \mathbf{g} \) is a given function.

### 2.1 Approximation by the finite element method

We start from the weak formulation of the problem (1) and (2), which reads (see e.g. [4]):

Seek \( \mathbf{u} \in V_g \) and \( p \in L^2(\Omega) \), satisfying

\[
\int_{\Omega} (\mathbf{u} \cdot \nabla)\mathbf{u} \cdot \mathbf{v} d\Omega + \nu \int_{\Omega} \nabla \mathbf{u} : \nabla \mathbf{v} d\Omega - \int_{\Omega} p \nabla \cdot \mathbf{v} d\Omega = 0 \quad \forall \mathbf{v} \in V,
\]

\[
\int_{\Omega} q \nabla \cdot \mathbf{u} d\Omega = 0 \quad \forall q \in L^2(\Omega).
\]

Here the spaces are

\[
V_g := \{ \mathbf{u} \in H^1(\Omega)^3, \mathbf{u} = \mathbf{g} \text{ on } \Gamma_D \},
\]

\[
V := \{ \mathbf{v} \in H^1(\Omega)^3, \mathbf{v} = 0 \text{ on } \Gamma_D \}.
\]

The first step for using the finite element method is division of the solution domain \( \Omega \) into \( N \) hexahedra \( H_K \) fulfilling \( \bigcup_{K=1}^N \overline{H_K} = \overline{\Omega} \) with \( \mu_{\mathbb{R}}(H_K \cap H_L) = 0 \), \( K \neq L \). On each element, we approximate the solution of pressure and velocity and the test functions by polynomials of certain degrees. For the Navier-Stokes equations, several types of finite elements are suitable [4], and we have chosen the Taylor-Hood finite elements. These approximate pressure functions by polynomials of the first degree and velocity functions by polynomials of the second degree on each element.

During the assembly of the system of algebraic equations, we substitute into the weak formulation (5) and (6) for \( \mathbf{u}, p, \mathbf{v}, \) and \( q \) their finite element counterparts and obtain the following system of algebraic equations

\[
\begin{bmatrix}
\nu \mathbf{A} + \mathbf{N}(\mathbf{u}) & B^T \\
B & 0
\end{bmatrix}
\begin{bmatrix}
\mathbf{u} \\
p
\end{bmatrix}
=
\begin{bmatrix}
f \\
g
\end{bmatrix},
\]

where \( \mathbf{u} \) is the vector of unknown coefficients of velocity, \( \mathbf{p} \) is the vector of unknown coefficients of pressure, \( \mathbf{A} \) is the matrix of diffusion, \( \mathbf{N}(\mathbf{u}) \) is the matrix of advection which depends on the solution, \( B \) is the matrix from the continuity equation, and \( f \) and \( g \) are discrete right-hand side vectors arising from Dirichlet boundary conditions. Each part of system (7) is assembled as in [4].

System (7) is nonlinear due to the matrix \( \mathbf{N}(\mathbf{u}) \), and for its linearisation, we use the Picard’s iteration. This leads to solving a sequence of linear systems of equations in the form

\[
\begin{bmatrix}
\nu \mathbf{A} + \mathbf{N}(\mathbf{u}^k) & B^T \\
B & 0
\end{bmatrix}
\begin{bmatrix}
\mathbf{u}^{k+1} \\
p^{k+1}
\end{bmatrix}
=
\begin{bmatrix}
f \\
g
\end{bmatrix},
\]

where \( \mathbf{N}(\mathbf{u}^k) \) means that we substitute a solution of velocity from the previous step to the matrix \( \mathbf{N} \). This—already linear—nonsymmetric system is solved by means of iterative substructuring using one step of the BDDC method as the preconditioner.
2.2 The BDDC preconditioner

For using the BDDC method as a preconditioner, we need to decompose the solution domain $\Omega$ into $N_S$ nonoverlapping subdomains. Then the unknowns shared by several subdomains are those on the interface, denoted as $\Gamma$. For Taylor-Hood elements, it is important that unknowns for velocity $u_\Gamma$ as well as for pressure $p_\Gamma$ appear at the interface.

The next step is a reduction of the system to the interface problem

$$ S \begin{bmatrix} u_\Gamma \\ p_\Gamma \end{bmatrix} = g, \quad (9) $$

where $S$ is the Schur complement with respect to interior unknowns and $g$ is the reduced right-hand side. Problem (9) is solved by the BiCGstab method [5] using one step of BDDC as the preconditioner for the residual obtained from the $k$-th iteration. More details of this strategy can be found in [2].

3 Numerical results

Our calculations aim at an industrial problem of oil flow inside hydrostatic bearings. In this section we present results for two kinds of problems. The first problem is without considering motion of the bearing. For this set-up, we have performed an experiment to validate the numerical solver. The second problem is a bearing sliding in a straight direction, and this set-up corresponds to a running production machine. The solution domain with its dimensions is depicted in Figure 1, and the two studied cases correspond to prescribing zero and non-zero velocity at the bottom side of the domain (see Figure 4).

We have used the Gmsh\(^1\) software [6] to create the mesh, which we have decomposed into 21 subdomains using our ‘geometric’ partitioner preferring straight cuts between subdomains described in [3]. The mesh consists of 124,828 elements which correspond with 3,269,679 unknowns and 186,834 interface unknowns. The decomposed mesh is presented in Figure 2. A detail of the mesh in the throttling gap, where elements with high aspect ratio occur, is in Figure 3.

We consider the following boundary conditions (see Figure 4):

\(^1\)http://gmsh.info
Figure 2: Computational mesh decomposed into 21 subdomains.

Figure 3: Detail of the elements inside the throttling gap.

- $\Gamma_{\text{input}}$ - prescribed velocity of the oil at the entrance to the hydrostatic cell
- $\Gamma_{\text{wall}}$ - prescribed zero velocity of the oil on the walls
- $\Gamma_{\text{output}}$ - the ‘do-nothing’ boundary condition (4) on the exit of hydrostatic cell
- $\Gamma_{\text{bottom wall}}$ - prescribed velocity of the movement of the bearing

The computations are performed by a parallel finite element solver described in [7, 8]. For solving the linearized systems of equations by the BDDC method, the BDDCML\textsuperscript{2} library [9] is used. Our simulations were performed on the Salomon supercomputer at the IT4Innovations National Supercomputing Center in Ostrava.

3.1 Bearing without motion

At first we dealt with the problem without sliding of the bottom wall. We have been able to validate our calculation for this case. The experiment was performed at the Research Center of Manufacturing Technology at the Faculty of Mechanical Engineering of the Czech Technical University in Prague. A scheme of the measurement is shown in Figure 5.

In this experiment, a throttling gap $h$ of the hydrostatic bearing (HSK), the flow rate $Q$ through the selected branch of the circuit, and the temperature $T_2$ of the oil Fuchs Renolin B 46 HVI were measured. The mean inflow velocity $v_{\text{mid}}$ is then computed from $Q$. The last input parameter is the dynamic viscosity $\mu$, which we derive from the measured temperature $T_2$ using tables provided with the oil. For comparing our simulations with the experiment, we use the value of pressure at the entrance of the hydrostatic cell $p_2$. The values obtained by the experiment are summarized in Table 1.

\textsuperscript{2}http://users.math.cas.cz/~sistek/software/bddcml.html
The last step is adding boundary conditions as in Figure 4. More specifically, we set the parabolic velocity profile with the prescribed mean velocity $v_{\text{mid}}$ on the circular input. The velocity of sliding at the bottom wall is set to zero in this case. The oil flows into the atmospheric pressure at the edge of the throttling gap $\Gamma_{\text{output}}$. Hence, after the computation, we postprocess the field of pressure in the whole domain by adding this constant value 100,000 Pa.

In Figures 6 and 7, results for pressure and velocity fields are presented. In this case, 3 Picard's iterations were needed to reach precision $\|u^k - u^{k-1}\|_2 \leq 10^{-4}$, and in average 798 BiCGstab iterations were needed for the linearised system to achieve precision of the relative residual $\|r^k\|_2 / \|g\|_2 \leq 10^{-5}$.

We can see from the results that the pressure inside the hydrostatic cell (and therefore at the input) is almost constant and the pressure drop is realized in the throttling gap. We can also see that the oil flows out equally along the edge of the throttling gap.

The value of pressure at the entrance of the hydrostatic cell $p_2$ obtained from our calculation is 2.826 MPa, which presents only 4% difference from the measured value 2.7124 MPa. This is an encouraging agreement within the accuracy of our experimental set-up.

$$
\begin{array}{|c|c|c|c|}
\hline
v_{\text{mid}} \text{ [m/s]} & h \text{ [\mu m]} & \mu \text{ [Ns/m$^2$]} & p_2 \text{ [MPa]} \\
\hline
0.44 & 61.3 & 0.0712 & 2.7124 \\
\hline
\end{array}
$$

Table 1: Summary of the measured parameters used for the simulation. The measured value of $p_2$ is compared with the one from the simulation.
3.2 Bearing with sliding

<table>
<thead>
<tr>
<th>$v_{\text{mid}}$ [m/s]</th>
<th>$h$ [$\mu$m]</th>
<th>$\mu$ [Ns/m$^2$]</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.3</td>
<td>50</td>
<td>0.1</td>
</tr>
</tbody>
</table>

Table 2: Parameters for the simulation of the sliding bearing.

After a successful validation on the previous problem, we consider the problem of a sliding bearing, which corresponds to operational conditions of the machine. The boundary conditions are the same as in the previous case except for the velocity of the bottom wall, which is set to $v_{\text{bottom wall}} = 1$ m/s. The input parameters for this computation are summarized in Table 2. For this case, we needed 3 Picard’s iterations to reach precision $\|u^k - u^{k-1}\|_2 \leq 10^{-4}$, while in average 748 BiCGstab iterations were needed for the linearised system to achieve precision of the relative residual $\|r^k\|_2 / \|g\|_2 \leq 10^{-5}$. A plot of the pressure is shown in Figure 8, and the streamlines are presented in Figure 9.

We can see from the results that the pressure field is almost indistinguishable from the case without motion, with the pressure drop realized only in the throttling gap, and an almost constant pressure in in the rest of the hydrostatic cell. The velocity field, however, changes dramatically with the sliding. We can see that a large vortex rolling inside of the hydrostatic cell is formed, and the oil is pulled by the motion of the bottom wall towards one side of the throttling gap. The latter effect is important for setting the operational regime of the machine, especially the maximum
The bearing is moving to the left-hand side.

velocity of the sliding such that the oil flow towards the front of the bearing is still maintained.

4 Conclusion

Our goal has been to simulate flows of oil in hydrostatic bearings during their motion. The flow is modelled as three-dimensional, and the steady Navier-Stokes equations are discretized by the finite element method. The problem is solved in parallel using Picard’s linearization. One step of BDDC is used as a preconditioner to the linearized problem solved by the BiCGstab method.

By applying this approach, we have been able to solve the dimensionally complicated problems of hydrostatic bearings with real-scale geometries. The results by the developed parallel solver have been compared with an experiment, and a remarkable agreement has been achieved.

We have then applied the solver to the problem of a sliding bearing. While the sliding does not seem to have a large effect on the pressure distribution in the bearing, the velocity field changes dramatically. Our simulations have revealed that a dominant feature of the flow during sliding is a large vortex ‘rolling’ inside the hydrostatic cell.

As the next step, we would like to include the temperature field and heat transfer and dissipation into our simulations. This should allow us to determine the thermo-mechanical state of the bearing.

Acknowledgment

This work was supported by the Czech Technical University in Prague through the student project SGS16/206/OHK2/3T/12, by the Czech Science Foundation through grant no. 18-09628S, and by the Czech Academy of Sciences through RVO:67985840. Computational time on the Salomon supercomputer
has been provided by the IT4Innovations Centre of Excellence project (CZ.1.05/1.1.00/02.0070), funded by the European Regional Development Fund and the national budget of the Czech Republic via the Research and Development for Innovations Operational Programme, as well as Czech Ministry of Education, Youth and Sports via the project Large Research, Development and Innovations Infrastructures (LM2011033).

References


