FastStokes: A Fast 3-D Fluid Simulation Program for Micro-Electro-Mechanical Systems

by

Xin Wang

B.S. Mechanical Engineering, Beijing University of Aeronautics & Astronautics, Beijing (1995)
M.S. Mechanical Engineering, Washington University, St. Louis (1998)

Submitted to the Department of Electrical Engineering and Computer Science
In partial fulfillment of the requirements for the degree of

Doctor of Philosophy

at the

MASSACHUSETTS INSTITUTE OF TECHNOLOGY

June 2002

Signature of Author_____
Department of Electrical Engineering and Computer Science
May 23, 2002

Certified by_____
Jacob K. White
Professor of Electrical Engineering and Computer Science
Thesis Supervisor

Accepted by_____
Arthur C. Smith
Chairman, Departmental Committee of Graduate Students

FastStokes: A Fast 3-D Fluid Simulation Program for Micro-Electro-Mechanical Systems

by

Xin Wang

Submitted to the Department of Electrical Engineering and Computer Science
on May 23, 2002, in partial fulfillment of the
requirements for the degree of
Doctor of Philosophy

Abstract

We have developed boundary integral equation formulas and a corresponding fast 3-D Stokes flow simulation program named FastStokes to accurately simulate viscous drag forces on geometrically complicated MEMS (micro- electro- mechanical systems) devices. Unlike the 3-D finite element or finite difference solvers which often take days to run to completion or fail when geometry gets complicated, the FastStokes 3.0 simulation program is capable of simulating complicated devices such as resonators, accelerometers, and micro-mirrors on PC computers in minutes.

The FastStokes 3.0 simulation program is a fast 3-D boundary-element simulation program that uses only surface discretizations. The implementation of the Precorrected-FFT algorithm in combination with the GMRES algorithm substantially improves the speed of this simulation program. An efficient two-step approach that successfully handles the null space of the singular incompressible Stokes BEM operators is developed to avoid numerical errors and solution discontinuities. An analytical flat-panel kernel integration algorithm is implemented in FastStokes and an accurate curved-panel integration algorithm is also developed.

Both an incompressible FastStokes solver and a compressible FastStokes solver have been developed and tested. They are not only fast, but also accurate. The incompressible FastStokes solver solves the steady incompressible Stokes equation; the effectiveness of this fast solver has been repeatedly proved by the close matches between numerical simulation results and experiments, within engineering accuracy (5-10% error). The numerical simulation results of a comb drive resonator, the ADXL 76 accelerometer, and a micro-mirror are given. The compressible FastStokes solver solves a linearized compressible Stokes equation that is also capable of capturing the weak air compression effect in MEMS devices. Therefore, the compressible FastStokes solver is a more general simulation program, and it is especially

useful when the strength of the fluid compression effect is uncertain. The solutions of the compressible FastStokes are compared with the analytical solutions of the linearized compressible Reynolds equation. Numerical simulations of some common structures that may exhibit compression effect when packaged in gases are also given.

The development of the FastStokes simulation program makes possible the rapid full 3-D fluid simulation of geometrically complicated MEMS devices.

Thesis Supervisor: Jacob K. White
Title: Professor of Electrical Engineering and Computer Science

Acknowledgments

I would like to express my sincere thanks and appreciation to my advisor, Professor Jacob White. This research work would have been impossible without his patience, encouragement, advice, and insights. Many times I felt frustrated and confused, but giving up has never come to mind as an option because I could always feel support and guidance from Jacob. Working with him in the past four years has been a challenging and joyful experience. I am also very grateful to Professor Jeffrey Lang and Professor Jaime Peraire for serving on my graduate committee and offering many valuable suggestions.

Dr. Wenjing Ye, Dr. Jingfang Huang, and Dr. Peter Mucha kindly offered many valuable suggestions; I have benefited a lot from our discussions. I also want to thank all my officemates, particularly Deepak Ramaswamy, Igor Balk, Michal Rewienski, and David Willis, for their help and for sharing so many pleasures with me.

I cannot find words to express my gratitude to my parents, for their love and support that have helped me through the past thirty years.

Finally, and most importantly, I would like to thank my girlfriend Junyao. Her patient encouragement is the best way to relax my stressed body and soul. I dedicate this thesis to her.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

1.1 MEMS

Micro-electro-mechanical systems (MEMS) are multi-physic systems on single substrates using microfabrication technologies that are developed based on electrical integrated circuits (IC) fabrication technologies. The "multi-physic" systems may have mechanical components, such as beams and gears; or electrical components, such as circuits; or chemical subsystems, such as reaction chambers with solutions; or optical components, such as tiny lenses. Those components or subsystems are tiny but fully functional. Therefore, MEMS is a combination of miniaturization and multi-functionality. The rapid development of MEMS technology over the past twenty years not only reveals the great potential of the MEMS industry, but also realizes many miniaturization dreams our predecessors were able only to imagine. More importantly, the development of MEMS opened a new design space. Based on batch fabrication and system integration, MEMS devices are compact, complicated, and inexpensive. More and more people are starting to realize that MEMS are the key to many tough design projects. In the 70s and 80s, the success of MEMS pressure sensors and accelerometers attracted broad attention to MEMS products. During the past ten years, the MEMS industry has boomed. The boom has led to the generation of many new concepts, and the development of many new products such as optical switches, flat panel displays, and on-chip detectors. The predicted big markets for these MEMS products have pushed the interest in MEMS research to a still higher level.

MEMS devices are usually fabricated on silicon or quartz wafers. A single wafer typically contains many dies that can be separated and then packaged individually. Current microfabrication technologies can be classified into two categories: thin-film surface micro-

machining technologies and bulk micro-machining technologies. Thin-film surface micro-machining technologies are often used to generate thin surface 2-D structures on a silicon substrate, such as those early MEMS sensors and actuators. Bulk micro-machining technologies are relatively new and not as closely related to the IC fabrication process as thin-film micro-machining technologies. Bulk micro-machining processes such as DIRE (Deep Reactive Iron Etch) and LIGA (an acronym from German words for lithography, electroplating, and molding) are capable of generating thick (high-aspect-ratio) structures. But at the current stage, true 3-D micro-fabrication is still very difficult. Figure 1.1 shows a tiny spider mite and some MEMS gears fabricated at Sandia National Lab. Obviously, MEMS devices are dramatically different from their macro counterparts. The feature sizes, the materials, and the fabrication processes may be totally new even to many experienced design engineers. MEMS product designs are strongly limited by the microfabrication technologies. The following several paragraphs list the key features of MEMS.

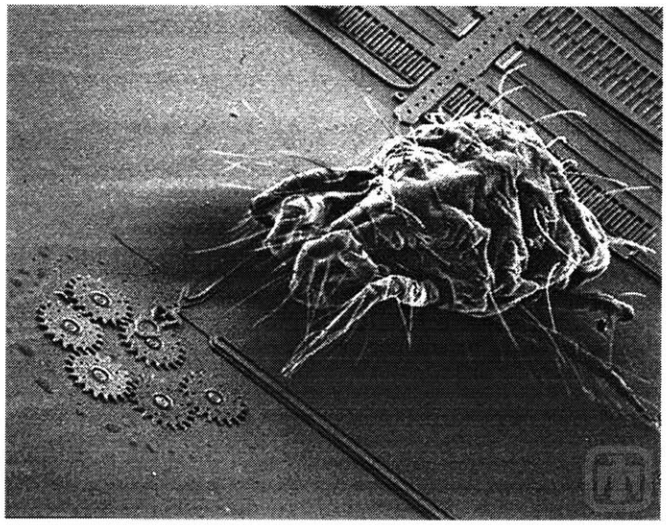

Figure 1.1: A spider mite and MEMS gears (http://mems.sandia.gov)

MEMS devices are small, with a feature size on the order of 1 to10 micrometers. This is the first and most important feature of MEMS. For example, a MEMS micro-mirror is well known to be a critical component of the optical switches to be used in the next generation of all-optical networks. A major reason is that MEMS micro-mirrors can be made small enough

to be comparable with the diameters of optical fibers and complicated enough to accurately perform the light-signal-switching function. The current state-of-the-art design integrates more than 1,000 micro-mirrors on one single chip to switch light signals from one bundle of optical fibers to another bundle of fibers. In addition to the mechanical components, circuits are usually integrated on the same chip to accurately control these tiny mirrors.

The second important feature of a MEMS device is that it covers many engineering disciplines or multiple physical domains, such as electrical, mechanical, chemical, and biomedical. As a result, many MEMS projects require the collaboration of researchers from different fields. And there are MEMS research projects in almost every engineering discipline. A few interesting ongoing projects are: a micro-engine power generator, 3-axis accelerometers, RF MEMS devices, MEMS data storage devices, MEMS heat exchangers, MEMS infrared vision systems, micro fuel cells, and lab-on-chip biomedical detectors (http://www.darpa.mil/mto). Most MEMS devices can be categorized into five major application areas: sensors, RF-MEMS, optical MEMS, biomedical MEMS/micro-fluidic devices, and power MEMS.

The third feature of a MEMS device is its close relationship with integrated circuits. MEMS fabrication technologies are based on IC fabrication technologies; MEMS devices may also be integrated with IC and the combined system is capable of performing more complex functions. From a more general point of view, MEMS fabrication is an extension of IC fabrication techniques. Although most companies still prefer to fabricate MEMS and circuits separately to reduce the complexity of fabrication processes, experts believe that the fabrication costs can be reduced significantly in many cases by integrating MEMS with circuits. The *i*MEMS (Integrated Micro-Electro-Mechanical Systems) fabrication process was developed by the Micromachined Product Division of Analog Devices Inc.; it is a well-developed process that integrates circuits and MEMS on one single chip. This process has been used to fabricate many MEMS accelerometers, and it has proven to be a great success.

The fourth feature of a MEMS device is its geometric complexity. The special micro-fabrication techniques make it relatively easy to fabricate geometrically complicated devices.

For example, a comb-drive structure is commonly used in electrostatically actuated devices or capacitance-sensing devices; a single electrostatically actuated resonator may have tens of fingers, but some gyro sensors use several thousand fingers in a single device. As a result, many MEMS devices are geometrically very complicated.

1.2 MEMS modeling and fluid simulation of geometrically complicated MEMS devices

Batch fabrication helps keep many MEMS products at low prices; most of the time, people prefer to use large wafers that generate high yields. But the device development costs, as can be imagined, are still very high. The expensive fabrication equipment, the high equipment operating expenses, the extensive use of manpower, and the long fabrication process make developing even simple devices, such as pressure sensors, very expensive. The designers usually suffer countless failures before they see the first fully functioning device. Reducing process steps and step costs is only one way to reduce cost. Another important approach to cost reduction is to optimize the design and avoid making mistakes. A design failure found after fabrication means the loss of a large amount of capital and weeks of precious time. Accordingly, MEMS development relies heavily on modeling and numerical simulation. Fast and accurate simulation tools are highly valued by the designers, as these tools help verify designs, improve designs through optimization, and understand design parameters quantitatively. Applying CAD tools to device simulation has become a crucial part of MEMS product development.

However, available CAD resources are limited, and developing good MEMS simulation tools is very challenging since simulations are strongly limited by numerical methods and computer resources. Developing high-performance user-friendly CAD software requires theoretical break-throughs and handling tremendous numbers of details. Existing CAD software, such as ANSYS, ABAQUS, CoventorWare, IntelliSuite, and MEMSCAP CAD tools, is certainly capable of solving many simulation questions, but there are still plenty of problems that

existing simulation tools cannot handle. One of the toughest among those problems left unsolved is the fluid simulation of geometrically complicated MEMS devices.

Quite often air viscosity strongly affects the dynamic performance of air-packaged micromachined devices. Because of the narrow gaps and close distances between movable and static parts, the viscous damping forces are usually too strong to ignore. Lots of attention has been attracted to calculating the viscous drag forces on MEMS devices. But devices such as resonators, accelerometers, or micro-mirrors are geometrically too complicated for existing simulation tools. Even worse, these kinds of air damping problems are usually exterior problems, which means fluid (air) is outside, and discretizing the 3-D fluid domain may generate huge numbers of elements or nodes. Numerical approaches using finite-element-method (FEM) based or finite-difference-method (FDM) based 3-D Navier-Stokes equation solvers can only handle simple geometries; they become very time-consuming or fail when geometries get complicated [40]. For this reason, many people are forced to use empirical equations or simple analytical solutions [30]. But commonly used semi-analytical approaches based on simple 1-D or 2-D fluid models require profound understanding of both the theories and the devices. Such approaches only give reasonable estimates in special cases and fail to achieve sufficient accuracy in general cases [7, 34]. On the other hand, MEMS designers are actively developing new products and exploiting new design concepts; new designs tend to be more complicated so that they can perform sophisticated functions. With numerous fluid simulation problems left unsolved, developing a device-level full 3-D fast solver would be the best answer to these tough challenges [32].

We have been working on developing a fast 3-D Stokes equation solver for applications in MEMS fluid simulations. The Stokes equation is well known to be an accurate fluid model for many MEMS devices packaged in liquid or in gases at a pressure near 1 atmosphere. Because of the small feature size of the MEMS devices, even the velocities of fast oscillating parts are usually small. This implies a small Reynolds number. Based on the small Reynolds number assumption, the 3-D isothermal Navier-Stokes equation can be simplified to the Stokes equation. In the incompressible case, the Stokes equation is a linear equation that is easy to

solve; in the compressible case, the Stokes equation can be linearized and simplified. These 3-D linear Stokes models are accurate enough for modeling many geometrically complicated MEMS devices such as resonators, accelerometers, and micro-mirrors.

Our goal is to develop a fast, accurate, and memory-efficient solver that can perform fast 3-D fluid simulation. Therefore, the numerical solution scheme is also a critical aspect of the new simulation program. We chose the boundary element method (BEM) because it has significant advantages when applied to solving linear equations. The boundary discretization used by BEM generates many fewer elements compared to the volume discretization used by FEM or FDM; this dramatically reduces the discretization complexity and the number of unknowns. However, BEM generates dense system matrices, while FEM and FDM generate sparse matrices that are computationally easy to solve. Solving the BEM-generated dense matrix problems used to be computationally expensive, and it was mistakenly considered a major disadvantage of BEM. This situation changed substantially after the accelerated BEMs were developed. The Precorrected-FFT (PFFT) algorithm [21, 22], which has been implemented in several fast solvers, has proven to be a kernel-independent $O(n\log(n))$ algorithm; it is not only fast but also accurate. The PFFT-accelerated BEM is consequently applicable to the Stokes integral equations.

FastStokes is a fast 3-D Stokes flow solver we have developed [1, 49, 50]. This software program solves the incompressible Stokes equation and the linearized compressible Stokes equation using the PFFT-accelerated BEM; it is fast, accurate, and memory-efficient. The Stokes equations, numerical algorithms, solution schemes, and simulation results will be discussed in detail in later chapters.

1.3 Motivation

The nodal network method is another popular method for simulating large systems. If a large and complicated physical system can be subdivided into lumped elements, then the nodal

network method is an ideal choice for simulating the system. For example, circuit elements, such as capacitors, inductors, and resisters, are frequently modeled as lumped elements. KCL and KVL are then applied to generate a network system. However, the nodal network method does not apply to general cases. Many geometrically complicated MEMS structures cannot be subdivided into lumped sub-elements, because the accuracy would be low. Obviously, a fast full 3-D simulation program is the best choice for general 3-D problems.

This section lists several well-known MEMS devices to show the importance of a fast 3-D fluid simulation program. Because of the complicated geometries of these devices, 3-D simulations using FEM-based of FDM-based software usually generate many more unknowns than computers can efficiently handle. Some devices, like ADXL76, were designed many years ago. But the previous reports of viscous drag force simulations on this well-known accelerometer have been far from satisfying [33].

1.3.1 Comb-drive actuators, resonators, and accelerometers

Comb-drive structures have been extensively used as electrostatic actuation or sensing components by MEMS designers. The structure usually has two pairs of interdigitated combs, with one member of each pair fixed and the other movable. Whenever there is a potential difference between a pair of interdigitated combs, electrostatic force pulls the movable comb and that motion can be used for actuation. Comb-drive structures are also frequently used as sensing parts. The motion of the movable comb changes the capacitance between the interdigitated pair; this capacitance change can be sensed and used to compute the displacement. As mentioned before, simulating the damping behaviors of the MEMS comb-drive structure is not easy because of the complicated geometry, especially when there are large numbers of fingers. Figure 1.2 shows a micro-machined dynamometer fabricated in Sandia National Lab. This dynamometer is used to determine the coefficient of friction by measuring the normal and tangential forces. The rotation motion of the small "smooth gear" in the middle is actuated by the two sets of comb-drive actuators, the upper set and the right set. The left set (only part is shown here) is used to apply normal force, and the lower part is a

displacement-sensing component that senses the motion of the beam due to frictional force. Hundreds of fingers are used in each set of the comb-drive.

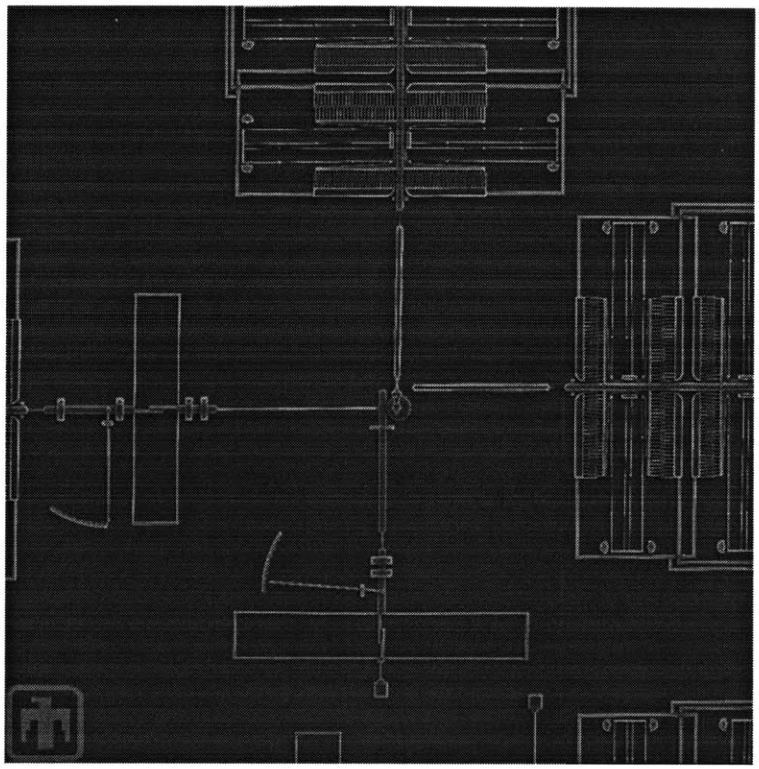

Figure 1.2: Sandia dynamometer (http://mems.sandia.gov/scripts/images.asp)

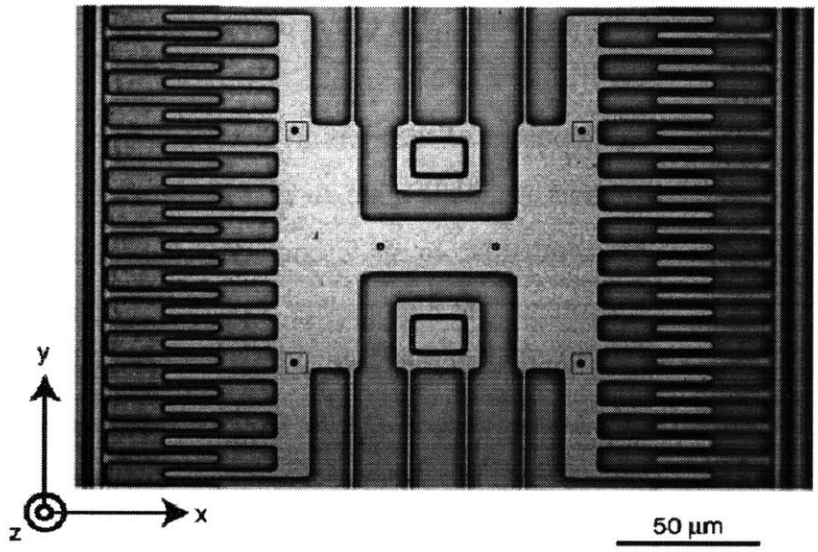

Figure 1.3: A MEMS resonator (http://umech.mit.edu)

The MEMS comb-drive resonator structure shown in Figure 1.3 is very similar to that of the comb-drive actuator. The shuttle is in the middle, supported on the thin tethers that are attached to the substrate. Only parts of the folded tethers are shown in this picture. The shuttle oscillates when an AC voltage is applied to one static comb-drive. Motions of the shuttle change the capacitance between the shuttle and the other static comb-drive, and the capacitance change can be detected. The MEMS resonator has potential applications in RF filters. When the device is packaged in air, the major source of damping is the viscous drag force, and this force is strongly dependent on geometry.

Figure 1.4 shows the ADXL76 accelerometer fabricated in the Micromachined Product Division of ADI (Analog Devices Inc), using the iMEMS (Integrated Micro Electro Mechanical System) process. This accelerometer has been used extensively in auto airbags as a collision sensor to trigger airbag deployment. The ADXL76 uses a capacitance-sensing principle. The shuttle is free to move along the axial direction; accelerations cause the shuttle

to move and the motion is detected using a differential capacitance sensing method. The displacement and acceleration are then calculated from the capacitance changes.

Understanding the dynamic performances of these comb-drive structures is very important in predicting or controlling the motion of the movable combs; the quality factor, which reflects the energy lose of the movable combs due to air viscous damping, is often preferred when modeling these actuators, resonators and accelerometers. However, accurately simulating the mechanical damping is almost impossible without a fast 3-D simulation program. This problem became very significant in modeling ADXL76. Because of the narrow gaps between fingers (see Figure 1.4), the viscous damping effect is very strong. An accurate optimization is necessary in this case since there are many design parameters. Unfortunately, the critical damping effect could not be accurately modeled without a fast solver that can handle complicated 3-D geometry. There are many reported efforts to model the air damping effect in ADXL76; the closest simulation result was 30% off the actual value [33]. Clearly, so inaccurate a result is not very helpful.

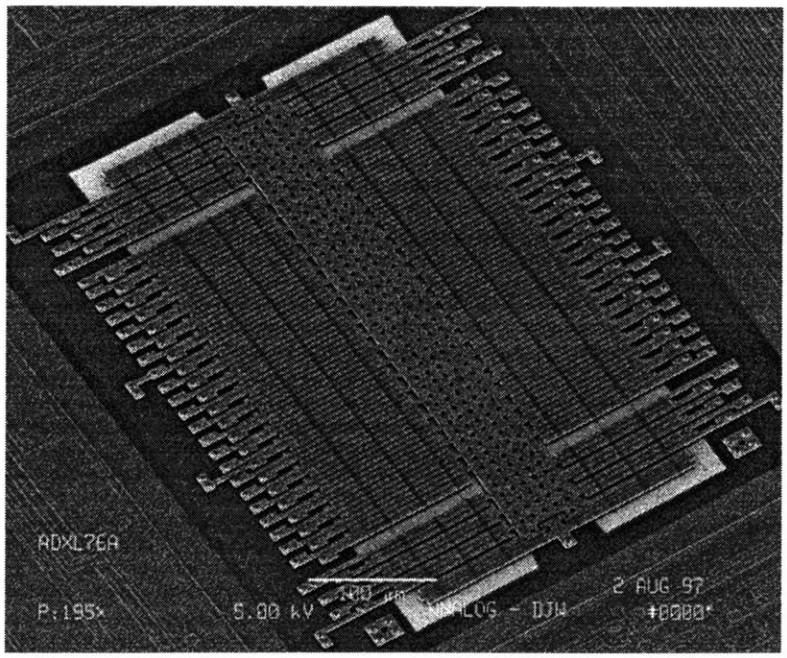

Figure 1.4: ADXL76 accelerometer

1.3.2 The MEMS micro-mirror

The micro-mirror has been used extensively in flat-panel projection displays, space telescopes, optical switches, etc. Based on applications, there are designs using different actuation and sensing principles. So far, the most promising application is the micro-mirror array for all-optical routers.

The micro-mirror array that directs light signals directly from fibers to fibers is the core component of the all-optical router. It must satisfy strict requirements to perform accurate and fast light signal switching [46]. Currently there are many micro-mirror designs using different actuation principles, but most designs still use air-packaged thin mirrors to direct light signals. By "air-packaged," we mean the device is packaged in gases with pressure around ambient air pressure (1 atm). The gas viscous damping force helps keep the micro-mirror mechanically stable so that it is not sensitive to outside incidental excitations. Figure 1.5 and Figure 1.6 show an electrostatically actuated micro-mirror. The mirror is sitting above the electrodes (not shown here) with two tethers attaching it to the gimbal, and another two tethers attaching the gimbal to the substrate. Two desired motion modes are the "mirror+gimbal" rotational mode around one axis and the mirror rotational mode around another axis.

However, calculating the quality factor is not easy. Because of the complex structure and the large aspect ratio of the air gap between the mirror and the electrode, only a full 3-D simulation program can accurately predict the viscous damping. The experimental results show that the damping force is so strong that some design has a quality factor of approximately two, though that was difficult to predict using semi-analytical approaches. Optimizing the quality factor has been almost impossible since the designers have had no accurate quantitative understandings of quality factor design sensitivities due to the lack of good simulation tools.

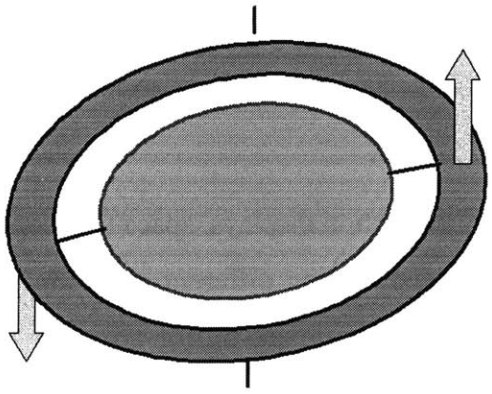

Figure 1.5: "Mirror+gimbal" rotational motion of the micro-mirror

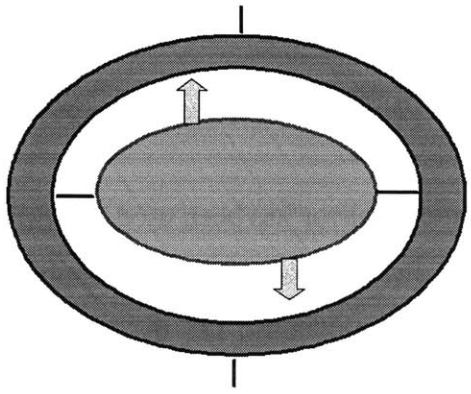

Figure 1.6: Mirror rotational motion of the micro-mirror

1.3.3 Fluid compression effect

Many studies have shown that air compression can make a significant contribution to the dynamic behavior of some MEMS devices [47]. When air is compressed, the compression causes a spring force that may change the resonance frequency of the device. This frequency shift is clearly indicated by the testing results of the out-of-plane motion of an accelerometer [37, 38]. A thin silicon wafer containing a large proof mass supported on a cantilever beam is bonded between two glass-covered thick wafers. There are narrow gaps between the proof mass and the two thick wafers. Gas damping and spring forces strongly affect the motion of the proof mass.

The measured results on this accelerometer were published by Veijola et al. Significant resonance frequency shifts were observed (Figure 1 in [38]) when the spring force generated by gases compression was increased. (In this case, the researchers changed the pressure in the package.).

When device is fabricated using the thin-film micro-machining technologies, the movable structures are usually too thin to generate large forces, which means there is only a weak air compression effect. To model the weak air compression effect, we need not only a computationally easier model but also a fast solution scheme.

1.4 The organization of this thesis

Chapters 2 to 4 of this thesis summarize related mathematical and numerical background. Chapter 5 to Chapter 9 will focus on the contributions of this thesis work, particularly those that have been implemented in FastStokes version 3.0.

As a starting point, Chapter 2 uses the simple electrostatic problem as an example to describe the major procedures of the PFFT-accelerated BEM. The PFFT (pre-corrected FFT) algorithm was specifically developed to accelerate the BEM solvers. When used in combination with a Krylov subspace solver, such as GMRES, both the computational cost and memory requirement of the BEM solver are reduced substantially.

Kernel integration is an important part of the BEM; Chapter 3 summarizes commonly used kernel integration algorithms. An analytical kernel integration algorithm is described in Section 3.4 and the Appendix [19]; this fast and accurate algorithm has been implemented in FastStokes 3.0.

Chapter 4 summarizes the equations and solution schemes used in the previous versions of the incompressible FastStokes solver.

It is well known that discretizing curved surfaces using flat panels may cause significant discretization errors; an accurate curved-panel integration algorithm is needed if curved-panel discretization is used. Chapter 5 presents an accurate curved-panel integration algorithm.

Chapter 6 describes a major problem found during the development of the incompressible FastStokes solver: the singular BEM operator issue, which has not been fully addressed in the literature. If not treated properly, this problem might cause large numerical errors and solution discontinuities across closed surfaces. We will discuss this problem in detail and present

solutions. The two-step method presented in this chapter has been implemented in the incompressible FastStokes solver to make it numerically robust [44].

In Chapter 7, we compare numerical simulation results with analytical solution and experimental results to show the accuracy of the steady incompressible FastStokes solver. A translating sphere, a comb-drive resonator, a micro-mirror, and the ADXL76 accelerometer are examined.

Chapter 8 has a description of a linearized compressible Stokes model. This fluid model successfully captures the weak fluid compression effect, and it is capable of predicting accurate damping and spring forces, the later can be responsible for resonance frequency shifts. The major part of this chapter focuses on the derivation of the BEM kernels and the boundary integral equations. A simplified linearized compressible Stokes equation is also presented. This equation neglects a very small term in the linearized compressible Stokes equation, but is much easier to treat numerically.

Chapter 9 uses numerical examples to show the efficiency of the compressible FastStokes solver. Numerical comparisons between the linearized compressible Reynolds equation, the simplified linearized compressible Stokes equation, and the steady incompressible Stokes equation are given. Simulation results show that the new simplified linearized compressible Stokes equation accurately captures the weak fluid compression effect. Therefore, the compressible FastStokes solver is a more general-purpose solver; it is good for both incompressible and compressible cases. Results also show that the incompressible Stokes equation, though not capable of modeling the spring forces, does give reasonably accurate damping force estimations when the fluid compression effect is weak.

In Chapter 10 we summarize our research work on FastStokes and give a brief conclusion.

Chapter 2

Background and basic algorithms

Advanced algorithms are the keys to developing a fast simulation program. The PFFT-accelerated boundary element method makes FastStokes fast and memory-efficient. Before touching the details of the Stokes flow problem, this chapter uses the electrostatic solver as an example to illustrate the boundary element method and the combined "GMRES+PFFT" approach.

Many MEMS fluid problems can be accurately modeled by the steady incompressible Stokes equation and the linearized compressible Stokes equation discussed in later chapters (Chapter 8 and Chapter 9). We choose to use a BEM for solving these equations because BEM is ideal for solving linear equations. BEM reduces the complexity of the simulations at the very first step, discretization. When applied to solving 3-D problems, BEM only needs the surface discretization, which is a 2-D discretization. This boundary discretization makes BEM particularly attractive since most of the structures we want to simulate are complicated 3-D structures with fluid in the exterior domain.

The PFFT algorithm was developed to accelerate the BEM. When coupled with an iterative matrix-vector solver, such as GMRES, PFFT-accelerated BEM has a low computational complexity of $O(n\log(n))$. This is a substantial improvement over the traditional BEM using Gaussian elimination method, which is an $O(n^3)$ method. In addition, the discretization advantage of BEM is fully revealed with a solution scheme as fast as those used in FEM or FDM. The development of these acceleration algorithms, such as the PFFT algorithm ($O(n\log(n))$) [21] and the Fast Multi-pole algorithm ($O(n)$) [8], brought new life to the old boundary element method.

This chapter starts by solving the simple electrostatic problem using the PFFT-accelerated BEM. Details of the algorithms are described in later sections of this chapter.

2.1 The electrostatic problem and the corresponding PFFT-accelerated BEM

Electrostatic actuation and capacitance-sensing have been used extensively in MEMS designs. Mathematically, those are electrostatic problems governed by the Laplace equation. Solving the electrostatic problem in complicated 3-D geometry is another difficult topic in MEMS simulation, although it is much easier than the fluid solution. Hence, we use it here as an example to illustrate the basic solution scheme of PFFT-accelerated BEM.

Given a system in a medium with dielectric constant ε, and assuming the system is in a quasi-steady state, the Laplace equation is,

$$\nabla^2 \phi = 0 \tag{2.1}$$

where ϕ is the potential. The potential boundary condition of a system with N objects is typically given in the format of,

$$\phi = \phi_i, \qquad i = 1, 2, \ldots, N \tag{2.2}$$

where ϕ_i is the potential of the i^{th} object. Assuming the potential is purely generated by the charge density on the surface of the objects, the charge density can be solved by solving a boundary integral equation:

$$\phi(\vec{x}) = \frac{1}{4\pi\varepsilon} \int \frac{1}{\|\vec{x} - \vec{y}\|} q'(\vec{y}) ds(\vec{y}) \tag{2.3}$$

where q' is the charge density, and $\|\vec{x}-\vec{y}\|$ is the distance between point $\vec{x}$ and $\vec{y}$. The charge Q' is simply an integration of the charge density over the surface, $Q'=\int q'(\vec{y})ds(\vec{y})$. The capacitance can be extracted using

$$C = Q'/\phi \tag{2.4}$$

To derive the boundary integral equation (2.3), we first assume there is a point charge q at position $\vec{y}$ in the free space; the governing equation (2.1) can then be rewritten as

$$\nabla^2\phi = \frac{-q\delta(\vec{x},\vec{y})}{\varepsilon} \tag{2.5}$$

where $\delta(\vec{x},\vec{y})$ is the 3-D delta function. The most straightforward way to solve (2.5) is using a Fourier transform. After taking a Fourier transform of (2.5); the analytical solution of the potential is solved in the Fourier domain; the solution is then transferred back to the 3-D space:

$$\phi(\vec{x}) = \frac{1}{4\pi\varepsilon}\frac{1}{\|\vec{x}-\vec{y}\|}q(\vec{y}) \tag{2.6}$$

where $\vec{x}$ is the so-called field point or evaluation point, and $\vec{y}$ is the source point. $\frac{1}{4\pi\varepsilon}\frac{1}{\|\vec{x}-\vec{y}\|}$ is the Green's function (or fundamental solution, or kernel) of the 3-D Laplace equation; it represents the response of the system due to a point charge disturbance. The Fourier transformation method will be given in detail when we calculate the more complicated Green's functions of the linearized compressible Stokes equation.

To derive integral equation (2.3) using the fundamental solution (2.6), we apply Green's second identity and neglect the double-layer integral which represents potential due to dipole distribution. Since the potential across the surface is continuous in our problem, there is no dipole distribution on the boundary. The potential in the space due to charge distribution on a surface is only a 3-D convolution of the charge density q' with the Greens' function. Equation (2.3) is called the single-layer integral equation of the Laplace problem.

To solve equation (2.6) at the discretized level, we chose to use the standard flat-panel piece-wise constant collocation approach. The surfaces are discretized using flat panels and the charge density distribution is assumed to be constant over the panels. Next, we generate the system equations by satisfying the voltage boundary condition at the centroid of each panel:

$$\phi(\vec{x}_i) = \sum_{j=1}^{\text{number of panels}} \left(\frac{1}{area(\text{panel j})} \int_{\text{panel j}} \frac{1}{4\pi\varepsilon} \frac{1}{\|\vec{x}_i - \vec{y}_j\|} ds(\vec{y}_j) \right) q_j \tag{2.7}$$

where q_j is the charge on the j^{th} panel. The discretized system equations can be rewritten in a matrix-vector form as:

$$\Phi = HQ$$

with (2.8)

$$H_{ij} = \frac{1}{area(\text{panel j})} \int_{\text{panel j}} \frac{1}{4\pi\varepsilon} \frac{1}{\|\vec{x} - \vec{y}\|} ds(\vec{y})$$

where Φ is the potential vector and Q is the vector of panel charges. Solving (2.8) yields the surface charges, which can then be used to calculate the capacitance. As mentioned before, a major advantage of BEM is that only the surface is discretized, generating many fewer elements than a volume discretization and making the system matrix much smaller. However, the integral equation method is a global approach; every point charge affects the potential distribution in the entire system. The BEM-generated system matrix H is accordingly a dense

matrix. Since there are efficient algorithms for solving the sparse matrices generated by FEM or FDM, it was believed that BEM was more expensive to use than FEM or FDM, even though there were many fewer unknowns. Accelerated approaches, which use the PFFT algorithm and the Fast Multi-pole algorithm, can solve the BEM-generated dense system matrix at low cost, and this has renewed interest in BEM. The next two sections describe one acceleration approach, the "GMRES+PFFT" approach, an $O(n \log(n))$ method .

2.2 GMRES

The GMRES (Generalized Minimal Residuals) algorithm [29, 36] is a Krylov subspace iterative solver. Given a linear system in matrix-vector format, $Ax = b$, where A is a nonsingular square matrix, GMRES starts from an initial guess x_0 and modifies the solution x_n at every step to minimize the norm of residual R_n

$$R_n = b - Ax_n \tag{2.9}$$

where x_n is in the Krylov subspace,

$$K_n = \left[b \middle| Ab \middle| \cdots \middle| A^{n-1}b \right] \tag{2.10}$$

A commonly used stable approach of doing the minimization is to generate a sequence of orthonormal vectors to form the matrix Q_n; these vectors also span the Krylov subspace K_n. The Arnoldi iteration starting from vector b is used in this orthogonalization step. The following relationship is given by the Arnoldi algorithm,

$$AQ_n = Q_{n+1}\hat{H}_n \tag{2.11}$$

where $\hat{H}_n$ is an $(n+1)\times n$ upper Hessenberg matrix. Letting $x_n = Q_n y_n$, we then minimize the norm of

$$Q_{n+1}^*(b - Ax_n) = Q_{n+1}^*(b - AQ_n y_n) = Q_{n+1}^*(b - Q_{n+1}\hat{H}_n y_n) = \|b\|e_1 - \hat{H}_n y_n \tag{2.12}$$

to yield y_n and $x_n = Q_n y_n$.

The most expensive computation of each GMRES iteration is the matrix-vector multiplication, which costs $O(n^2)$ operations. If the iteration reaches convergence criterion $R_m < Tolerance$ at the m^{th} steps, the total computational cost is $O(mn^2)$. Compared with the $O(n^3)$ direct Gaussian elimination method, the GMRES algorithm is much faster if $m << n$.

Several critical issues need to be discussed. First, we assume the matrix A is nonsingular, which is not always true. Dealing with a singular matrix is a major topic that will be discussed in Chapter 6. Second, the computational cost and memory requirement depends on m, but m is not guaranteed to be a small number. Reducing the number of iterations is important for a fast and memory-efficient solution scheme. A commonly used efficient approach is preconditioning the system matrix. Another way of reducing the number of iteration steps is to restart GMRES after a certain number of iterations. In such a case, the intermediate solution is used as an initial guess of the new iteration. Third, the GMRES algorithm minimizes the norm of the residual, which is guaranteed to be decreasing or unchanged at each iteration step. However, this doesn't guarantee that the solution converges to a finite meaningful value monotonically, although the solution usually converges to a meaningful value after a certain number of iterations. There are cases when the GMRES fails to converge, especially when the matrix is a singular matrix.

2.3 The Precorrected-FFT algorithm

The GMRES algorithm is a matrix-free method. It only needs one matrix-vector product in each iteration step, and it does not require the matrix to be explicitly stored and used [21]. This matrix-free property offers a potential way to save CPU time and memory. The accelerated BEM methods have been developed to accelerate the matrix-vector multiplication step by calculating the matrix-vector product approximately without using all entries in the system matrix.

In equation (2.8), every element in the system matrix represents the potential at a field point due to the charge distribution at the source point. This interaction is strong when the field point is close to the source point, and weak when the field point is far from the source point. Accordingly, the number of larger entries in the system matrix, which represent interactions between neighboring panels, is $O(n)$ if the number of neighboring panels is always smaller than a number k, which is true unless the panels are clustered into groups. Note that the BEM-generated system matrix is dense and the direct matrix-vector multiplication uses $O(n^2)$ operations. The weaker far-field interaction calculations must have cost $\left(O(n^2)-O(n)\right)\sim O(n^2)$ operations. The Precorrected-FFT (PFFT) algorithm described in this section uses an approximate approach rather than calculating matrix-vectors directly, so that the computational cost is reduced to O(nlog(n)).

The basic idea of the PFFT algorithm is as follows: separate the potentials due to nearby charges and far-field charges, and treat the far-field interactions in an approximate way using a uniform FFT grid and the FFT algorithm. Nearby interactions are usually strong due to the singular BEM kernels; they are computed directly using an analytical (or an accurate) kernel integration algorithm to avoid large numerical errors. The following are the four major steps of the PFFT algorithm:

a. Project the panel charges onto the FFT grid: $Q^{\text{grid}} = W^{\text{projection}} Q^{\text{panel}}$;

b. Compute grid voltages due to grid charges using the FFT. This step can be expressed as $\Phi^{\text{grid}} = IFFT\left(FFT(H^{\text{grid}}) \cdot FFT(Q^{\text{grid}})\right)$;

c. Interpolate the grid voltages back to panel voltages: $\Phi^{\text{panel}} = W^{\text{interpolation}} \Phi^{\text{grid}}$;

d. Directly compute nearby interactions and use the results to replace the inaccurate nearby interactions that go through the grid.

The cost of using the PFFT algorithm is dominated by the cost of the FFT step, which is $O(n\log(n))$. The projection and interpolation are local operations that cost only $O(n)$ floating-point operations. Hence, the total computational cost of the PFFT-accelerated BEM is $O(n\log(n))$. The four steps of the PFFT algorithm are illustrated by Figure 2.1.

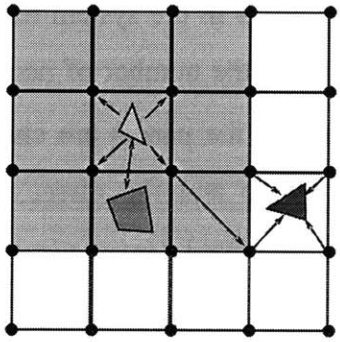

Figure 2.1: Four major steps of the PFFT algorithm

Projection

The purpose of introducing a uniform grid is to calculate approximately the far-field interactions; therefore the projection step must guarantee the accuracy of the far-field interactions. One way of projecting the panel charges to nearby grid point charges is through the help of far-field testing points. The charges at the grid points are calculated by forcing the potentials generated by the grid charges to equal the potentials due to the panel charges at those testing points.

Another straightforward way to calculate the projection operator is using the idea of local moment matching. For the sake of simplicity, we assume the closest grid point to the panel is picked as the origin, and let $G(\vec{x},\vec{y}) = \dfrac{1}{4\pi\varepsilon\|\vec{x}-\vec{y}\|}$. The potential at point $\vec{x}$ due to panel charge density q' is $\phi(\vec{x}) = q'(\vec{y})\int G(\vec{x},\vec{y})ds(\vec{y})$, which can be expanded in a Taylor series around the origin if the size of the panel is much smaller than the distance from the evaluation point $\vec{x}$ to the panel centroid.

$$\begin{aligned}\phi(\vec{x}) &= q'\int G(\vec{x},\vec{y})ds(\vec{y}) \\ &= q\left\{\frac{\int G(\vec{x},0)ds(\vec{y})}{\int ds(\vec{y})} + \sum_{j=1}^{3}\left[\frac{\partial(G(\vec{x},\vec{y}))}{\partial y_j}\right]_{\substack{\text{at}\\ \text{origin}}}\Phi_j + \frac{1}{2}\sum_{k=1}^{3}\sum_{j=1}^{3}\left[\frac{\partial^2(G(\vec{x},\vec{y}))}{\partial y_j\partial y_k}\right]_{\substack{\text{at}\\ \text{origin}}}\Phi_{jk} + O\left[\left(L/D\right)^3\right]\right\}\end{aligned} \tag{2.13}$$

where $q = \int q'ds(\vec{y})$, "at origin" means $y_i = 0$, and Φ_j and Φ_{jk} are the first and second-order moment around the origin, i.e.

$$\Phi_j = \frac{\int x_j ds(\vec{y})}{\int ds(\vec{y})} \tag{2.14}$$

L in (2.13) is the size of the panel, and D is the approximate distance from the origin to the evaluation point $\vec{x}$. If $L << D$, the above expansion only needs lower-order moment terms to be accurate. A similar expansion of the potential due to N point charges at the neighboring grid points is expressed as

$$\phi^{\text{grid}}(\vec{x}) = \sum_{m=1}^{N} G(\vec{x}, \vec{y}^{\,m}) c^m q$$

$$= q\left\{ G(\vec{x},0)\sum_{m=1}^{N} c^m + \sum_{m=1}^{N}\sum_{j=1}^{3}\left[\frac{\partial G(\vec{x},\vec{y})}{\partial y_j}\right]_{\text{origin}}^{\text{at}} c^m \Psi_j^m + \frac{1}{2}\sum_{m=1}^{N}\sum_{k=1}^{3}\sum_{j=1}^{3}\left[\frac{\partial^2 G(\vec{x},\vec{y})}{\partial y_j \partial y_k}\right]_{\text{origin}}^{\text{at}} c^m \Psi_{jk}^m + O\left[\left(L/D\right)^3\right]\right\}$$

(2.15)

where, $\Psi_j^m = y_j^m$ and $\Psi_{jk}^m = y_j^m y_k^m$ are the first and second-order moments of point m with respect to the origin, and c^m is the weight of the projection we wish to solve. A moment match leads to:

$$\begin{bmatrix} 1 & 1 & 1 & \cdots \\ \Psi_1^1 & \Psi_1^2 & \Psi_1^3 & \cdots \\ \Psi_2^1 & \Psi_2^2 & \Psi_2^3 & \cdots \\ \vdots & \vdots & \vdots & \ddots \end{bmatrix} \begin{bmatrix} c^1 \\ c^2 \\ c^3 \\ \vdots \end{bmatrix} = \begin{bmatrix} 1 \\ \Phi_1 \\ \Phi_2 \\ \vdots \end{bmatrix} \tag{2.16}$$

The unknown weight c^m can be solved and the projection of the panel charge onto the m^{th} grid point is $c^m \int q'(\vec{y}) ds(\vec{y})$.

Convolution

A convolution in the 3-D space can be easily calculated as a multiplication in the frequency domain, so Fourier transformations are used in this step. The Fourier transform of the grid kernel data $\widetilde{H} = FFT(H^{\text{grid}})$ is calculated once and stored, where $H^{\text{grid}}(i,j)$ expresses the relationship between the potential on grid point i and a point charge at grid point j, i.e., $\Phi^{\text{grid}}(i) = H^{\text{grid}}(i,j) q^{\text{grid}}(j) = \dfrac{1}{4\pi\varepsilon \left\| \vec{x}(i) - \vec{x}(j) \right\|} q^{\text{grid}}(j)$. The convolution is calculated using an FFT, followed by a multiplication, and an IFFT, i.e., $\Phi^{\text{grid}} = IFFT\left(\widetilde{H} \cdot FFT(Q^{\text{grid}})\right)$; so the

computational cost of this step is dominated by the cost of FFT, which is well known to be $O(n\log(n))$.

Interpolation

After the potentials at the grid points are calculated, we only need a local interpolation scheme to compute the approximate potential solution at the centroid of the panels. A polynomial interpolation is the simplest one: $\Phi^{\text{panel}} = W^{\text{interpolation}}\Phi^{\text{grid}}$, where $W^{\text{interpolation}}$ is the interpolation operator.

Pre-correction

From the discussions on the previous page, we know that the potential solution Φ^{panel} is only an inaccurate solution that contains two parts: the accurate potential solution due to far-field charge distributions, and the inaccurate potential solution due to neighboring panel charges. The pre-correction step replaces the inaccurate neighboring interactions with accurate ones.

Let $A^{\text{panel}}_{\text{k,neighbour}}$ be the accurate neighboring interactions calculated using a direct kernel integration algorithm; an analytical kernel integration algorithm for flat polygons is discussed in the next chapter. The inaccurate neighboring interactions are calculated again using the projection and interpolation operators. Note that both projection and interpolation are local operations, so we only need to calculate nearby grid potentials due to nearby grid charges in the second step. The potential on grid point i due to a point charge at grid point j is simply $\frac{q(j)}{4\pi\varepsilon\left\|\vec{x}(i)-\vec{x}(j)\right\|}$; FFT's are not necessary since we only calculate neighboring interactions.

Let $W^{\text{interpolation}}H^{\text{grid}}W^{\text{projection}}q^{\text{neighbour}}$ represent the inaccurate neighboring interactions; the following pre-correction step replaces the inaccurate nearby solutions:

$$\Phi_k^{panel} = \Phi_k^{panel} + A_{k,neighbour}^{panel} - W^{interpolation} H^{grid} W^{projection} q^{neighbour} \tag{2.17}$$

This step is also a local operation that costs $O(n)$. So the final computational cost of calculating the matrix-vector products using the PFFT algorithm is only $O(n\log(n))$.

Chapter 3

Kernel integration algorithms

Kernel integration is a critical step of the BEM; integration accuracy directly impacts the accuracy of the final solution. It is also a tricky step because there might be different algorithms that suit different BEM kernels at different levels of accuracy requirements. This is especially true in the singular case (or nearly-singular case), when the evaluation point is on (or close to) the panel. Therefore, developing a fast and accurate kernel integration subroutine often requires the combination of multiple algorithms. This section discusses several flat-panel kernel integration algorithms. A new curved-panel integration algorithm will be presented in Chapter 5.

The algorithms introduced here are mainly for the integration of $\frac{1}{r}$ type weakly-singular kernels, or strongly singular kernels and hypersingular kernels with Cauchy Principle Values (CPV).

3.1 Cubature method

The cubature (or quadrature) method is the easiest and usually the most efficient one if the integrand is a smooth function over the integration domain. An online Encyclopedia of Cubature Formulas can be found at:
http://www.cs.kuleuven.ac.be/~nines/research/topics/ecf.html.

However, the cubature method may not be accurate enough when applied to non-smooth functions, unless special cubature formulas are used. Increasing the cubature formula order

and number of cubature points does not help much. For example, certain cubature formulas do not work in integrating $\frac{1}{r}$ type kernels when the evaluation point is in the integration domain. The reason is that $r = 0$ if the evaluation point happens to be at the cubature point. And this often happens because the panel centroid is usually picked both as the evaluation point by the collocation scheme, and the cubature point by the cubature formulas. Changing cubature formulas or applying a subdivision approach (see Figure 3.1), may avoid divided-by-zero, but the accuracy is typically not good enough. Generally speaking, the cubature formulas developed for smooth functions do not work well when applied to singular integrands or nearly-singular integrands, because they cannot capture the singular behavior of the integrands. It is suggested that other methods such as the analytical method or the semi-analytical method be used in the singular case. Or a de-singularity approach (please refer to Section 3.4) be performed to remove the singularity before applying the cubature formulas.

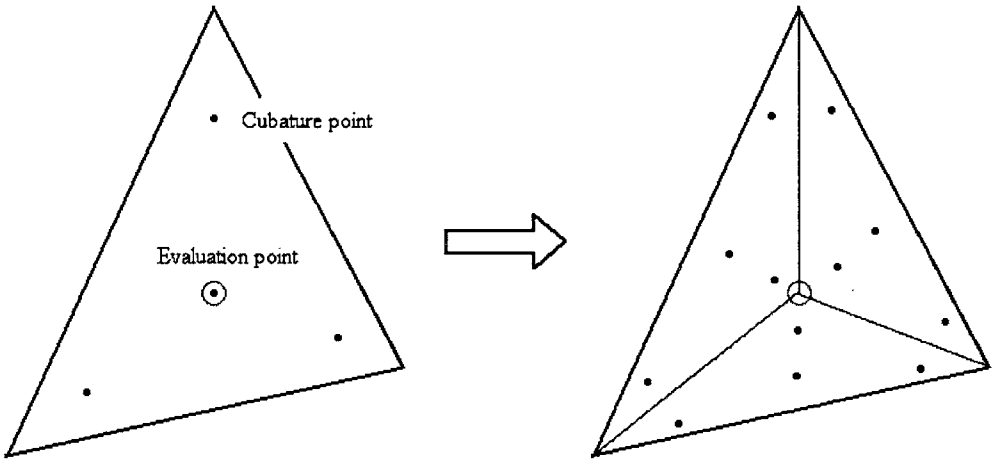

Figure 3.1: Subdivision of a triangle panel

3.2 Semi-analytical method

The semi-analytical method is a numerically more robust approach. This method calculates the inner integration of the two dimensional integration analytically, and uses a numerical

approach, such as Gaussian quadrature, to calculate the outer integral. It is numerically cheaper than the analytical method. A particularly attractive feature of this method is that a coordinate transformation (from Cartesian coordinate system to polar coordinate system) can remove the singularity of the 3-D $\frac{1}{r}$ type kernels.

Here we use the Helmhotz kernel, $\frac{e^{ikr}}{r}$, as an example to illustrate this method. The Helmhotz kernel is a weakly singular kernel with $\frac{1}{r}$ type singularity; it is similar to the unsteady Stokes kernel but much simpler.

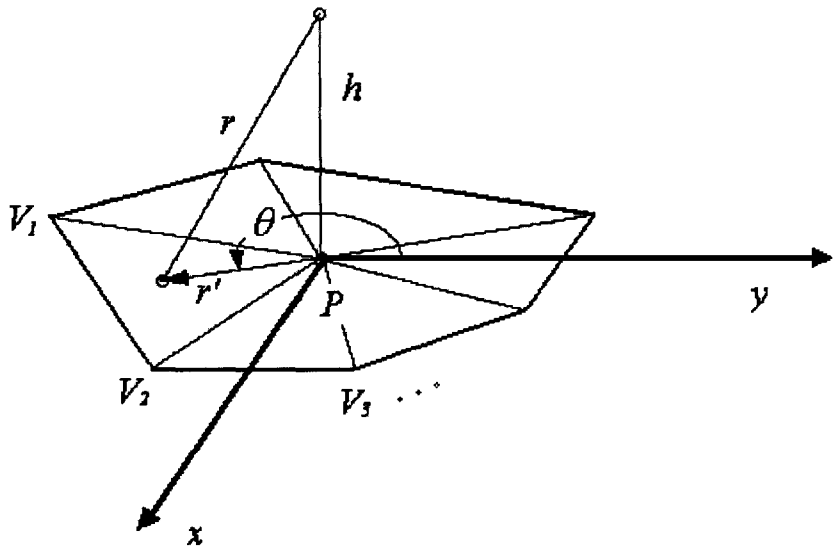

Figure 3.2: A flat panel in a polar coordinate system

Consider integrating the kernel in a polar coordinate system whose origin is the projection of the evaluation point on the flat panel plane (see Figure 3.2):

$$\iint \frac{e^{ikr}}{r} dxdy = \iint \frac{e^{ikr}}{r} r' dr' d\theta \tag{3.1}$$

with

$$r = \sqrt{r'^2 + h^2} \Rightarrow rdr = \frac{1}{2} d(r^2) = \frac{1}{2} d(r'^2) = r' dr'$$

where r, r', and h are defined in Figure 3.2. Then we have,

$$\iint \frac{e^{ikr}}{r} dxdy = \iint \frac{e^{ikr}}{r} r' dr' d\theta = \iint \frac{e^{ikr}}{r} rdrd\theta$$
$$= \sum_{j=1}^{\text{number of corners}} \left(\int_{\theta(V_j)}^{\theta(V_{j+1})} \frac{1}{ik} (e^{ikr(\theta)} - e^{ikh}) d\theta \right) \times sign\left(V_j \to V_{j+1} \to P \text{ is counterclockwise}\right) \tag{3.2}$$

If the projection of the evaluation point is outside the flat polygon, then some parts of the integration domain need to subtract. This can be decided by checking the sequence of the vertices V_j, V_{j+1}, and P, which form a triangle. Since the corners are named counterclockwise in Figure 3.2, define:

$$sign\left(V_j \to V_{j+1} \to P \text{ is counterclockwise}\right) = \begin{cases} 1 & true \\ 0 & V_j, V_{j+1}, P \text{ on the same line} \\ -1 & \text{false} \end{cases} \tag{3.3}$$

A shortcoming of this algorithm is that the outer integral will not be accurately calculated by the Gauss quadrature approach if P is very close to the edge. A subdivision can be applied to divide the line integral from $V_j \to V_{j+1}$ to two line integrals of $V_j \to P'$ and $P' \to V_{j+1}$, where P' is the projection of P on line $V_j \to V_{j+1}$. Increasing the number of quadrature points also helps.

3.3 Analytical method

Accuracy is the unbeatable advantage of the analytical methods. Although analytical kernel integration algorithm is not as fast as some numerical approaches, the kernel integration part in the PFFT-accelerated BEM solution scheme only costs $O(n)$, which is not the dominant cost. But from the accuracy point of view, the nearby interactions calculated by the kernel integration algorithm strongly affect the accuracy of the final solutions. So using the analytical kernel integration algorithm may significantly improve the accuracy at low cost.

In this section we discuss the analytical integration scheme for integrating

$$\iint_{\text{flat}} \frac{p(x,y,z)}{r^{2n+1}} ds \qquad n = 0,1,... \tag{3.4}$$

over flat polygons. Here r is the distance between the evaluation point and the panel, and p is a polynomial of x, y, and z. It is assumed that n in the above equation is an integer. This algorithm is based on Newman's analytical algorithm for $\iint_{\text{flat}} \frac{p(x,y,z)}{r} ds$ and $\iint_{\text{flat}} \frac{p(x,y,z)}{r^3} ds$. To make this algorithm even faster, we add a recursive scheme for solving the line integrations. We also generalize Newman's algorithm to calculate integrals in the form expressed by equation (3.4).

First, a local coordinate system (ξ,η,ζ) is set up such that the panel is put in the $\xi-\eta$ coordinate plane. Major computations are finished in the local coordinate system and the solutions are then transferred back to the global system. Newman's kernel integration algorithm starts from computing the potential due to constant dipole and source distributions. A recursive scheme with three recursive equations is then applied to extend the potential solutions due to lower-order source and dipole distributions to those due to the higher-order

ones. This recursive scheme, together with another recursive scheme we added to solve the line integrals, makes this analytical algorithm not only accurate but also fast, especially when multiple kernels need to be calculated simultaneously. Obviously, this makes the analytical kernel integration algorithm a good candidate for integrating the multiple Stokes kernels.

A detailed scheme for solving integrals in the form of $\int\frac{p(x,y,z)}{r}ds$ and $\int\frac{p(x,y,z)}{r^3}ds$ is given in the Appendix.

In general, if the integral is not in the form of the above source and dipole distributions but contains higher-order singularity, it is suggested that two sets of integrals $\int\frac{p(x,y,z)}{r^{2n+1}}ds$ and $\int\frac{p(x,y,z)}{r^{2n+3}}ds$ be calculated simultaneously using the given recurrence schemes. Some of the formulas in the Appendix need to be changed correspondingly. The following equation may be used to modify the formulas:

$$\frac{d}{dZ}\left(\iint\frac{p(\xi,\eta)}{r^{2n+1}}ds\right)=\frac{d}{dZ}\left(\iint\frac{p(\xi,\eta)}{\left((X-\xi)^2+(Y-\eta)^2+Z^2\right)^{\frac{2n+1}{2}}}ds\right)=-(2n+1)Z\iint\frac{p(\xi,\eta)}{r^{2n+3}}ds \tag{3.5}$$

where Z is the coordinate of the evaluation point in the panel normal direction in the local coordinate system.

The analytical solutions discussed here are accurate most of the time. However, one must be cautious with those equations. First, the solutions of some integrals are infinite when the evaluation point is on the panel, such as $\int\frac{1}{r^3}ds$. Second, the numerical stabilities of the above recurrence schemes need to be checked. If the evaluation points are far away from the panel, those integrals can be easily computed using cubature formulas. But the analytical algorithms

may not work well in such cases; some numerical instabilities have been noticed. We suggest using the cubature method whenever required accuracy is guaranteed.

3.4 De-singularity approach

As mentioned before, the cubature method is a cheap and efficient method for integrating smooth functions. Most 3-D kernels are smooth function when the evaluation point is far from the panel. The difficulty only happens when the evaluation point is on the panel or very close to the panel. Figure 3.3 shows the singular performance (the sharp spike) of the singular function, which is difficult to capture with the regular cubature method. For $\frac{1}{r}$ kernel, it is clear that the analytical method works fine. But so far there is no analytical method for $\frac{e^{ikr}}{r}$ kernel, and this may also be true for other weakly singular kernels with $\frac{1}{r}$ type singularity.

The de-singularity approach is a combined method. As we know $\lim_{r\to 0}\left(\frac{e^{ikr}}{r}-\frac{1}{r}\right)=0$, this motivates us to apply the cubature method to the non-singular part of the kernel, $\frac{e^{ikr}}{r}-\frac{1}{r}$, which is a smooth function, while the $\frac{1}{r}$ singular part can be calculated using the analytical method.

For kernels with other type of singularities, this de-singularity approach also works.

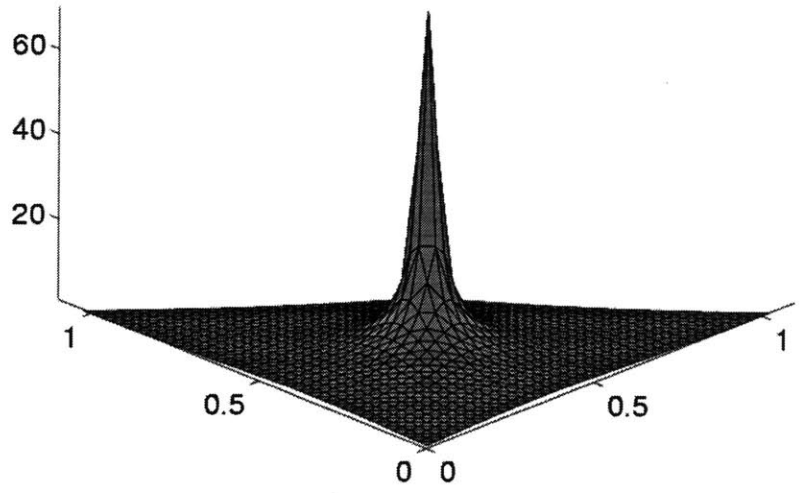

Figure 3.3: A singular function

3.5 The strongly singular and hypersingular kernels

Given a 3-D kernel with

$$\lim_{r\to 0}(\text{kernel}) = O(\frac{1}{r^s}) \tag{3.6}$$

the kernel is

$$\begin{cases} \text{weakly singular} & s = 1 \\ \text{strongly singular} & s = 2 \\ \text{hypersingular} & s \geq 3 \end{cases} \tag{3.7}$$

Weakly singular kernels are integrable kernels. Consider integrating a weakly singular kernel over a circle with radius a when the evaluation point is at the center:

$$\iint_{circle} \frac{1}{r} dxdy = \int_0^{2\pi} \int_0^a \frac{1}{r} r' dr' d\theta = 2\pi a \tag{3.8}$$

However, the integration of strongly singular kernels and hypersingular kernels in the singular case may not be a finite number. Although the integral equation is physically meaningful, it doesn't guarantee that any part of the kernel integration is mathematically finite. For example, as mentioned before, $\int \frac{1}{r^3} ds$ is infinity if the evaluation point is right on the panel. Fortunately, $\int \frac{1}{r^3} ds$ only appears as part of the double-layer kernel in the steady incompressible Stokes pressure integral equations. The double-layer integral is zero when the fluid motion is perturbed by the motion of rigid bodies.

If the kernel has Cauchy principle value, like the single-layer pressure kernel of the incompressible steady Stokes equation $\frac{x_i - y_i}{r^3}$, then the integration is a finite number if the evaluation point is on the panel but not on the edges. This can be proved by a similar integration as (3.8); the value of the integral is zero due to anti-symmetry. Solving this integration is easy if the analytical kernel integration algorithm is applied.

3.6 A conclusion

A good kernel integration scheme is crucial for a successful BEM solver. If multiple kernels are involved and a setup step is needed, a combinational approach that avoids repeating the setup step is usually the optimal solution. The integration scheme of the complicated linearized compressible Stokes kernels will be discussed in later chapters.

Chapter 4

The steady and unsteady incompressible FastStokes solvers

A fast 3-D Stokes equation solver has many applications in modeling MEMS devices. We have applied the PFFT-accelerated BEM to the Stokes integral equations and developed a FastStokes simulation program. Details about the PFFT accelerated BEM solution scheme have already been given in Chapter 2. This chapter covers some basic topics related to the Stokes flow and the FastStokes simulation program. Section 4.2 and 4.3 of this chapter summarize the critical aspects in deriving the steady and unsteady incompressible Stokes integral equations; detailed derivation procedure of more complicated linearized compressible Stokes integral equations is given in Chapter 8. Section 4.4 discusses a major modification of the PFFT algorithm when applied to the Stokes problem, which is a vector problem.

Before we touch the integral equations, let us briefly talk about Stokes flow and some key assumptions.

4.1 The Stokes flow and the Stokes equations

A viscous flow with a very small Reynolds number is called a Stokes flow or a creeping flow [13, 14, 24]. The Reynolds number is defined as $\mathrm{Re} = UL/\nu$, where U is the velocity, L is the characteristic length, and ν is the kinematic viscosity of the fluid. The Reynolds number is frequently used to determine which is dominant between the inertia and the viscous effects, since $\mathrm{Re} \propto \frac{\text{inertia force}}{\text{viscous force}}$. Given a fluid flow with a low Reynolds number (Re<<1), it is clear that the viscous force is dominant and the inertia force may be neglected to reduce the

complexity of analysis. Low Reynolds number flow has many applications, especially in modeling the slow fluid motions generated by small structures. Accordingly, many MEMS fluid problems may be modeled by the Stokes equation. The small feature size of the MEMS devices implies a small velocity even if the movable parts of the devices oscillate at high frequencies. Our goal is to develop a fast Stokes equation solver and apply it to modeling MEMS fluid problems.

Besides the low Reynolds number assumption, we have also assumed that the fluid is continuous when deriving the Stokes equation. This is an accurate assumption for liquids; it is also good for simulating gases in MEMS devices packaged around the ambient air pressure (1 atm). But this assumption needs to be checked carefully when applied to vacuum-packaged MEMS devices, because the mean free path increases inversely proportional to the pressure. If the mean free path of the gas molecules is large enough to be comparable to the feature size of the devices, the interactions between molecules and structure surfaces affect the motion of the molecules. In such cases, the flow should not be assumed to be continuous. The Knudsen number, defined as $K_n = \frac{\text{mean free path}}{\text{feature size}}$, is a non-dimensional number that is often used to justify the applicability of the continuity assumption. Generally speaking, the continuity assumption causes an error within engineering accuracy (5% error) if $K_n < 0.1 \sim 0.01$. For simple problems that can be modeled by 1-D or 2-D fluid models, adjusting parameters is an easy way to compensate for the gas rarefication effect. Such simple adjustments are usually also geometry-dependent, which makes it difficult to extend to general 3-D modeling [49].

In addition, we have also assumed that the device being modeled is in an isothermal condition. It is true that heating effects generated by viscous forces and electric currents may cause temperature rises, sometimes quite significantly, but the temperature gradients in MEMS devices are usually small due to the excellent thermal conductivity of silicon and polysilicon. And solution variations caused by temperature rises can be easily compensated for by adjusting the parameters accordingly.

4.2 Integral equations of the steady Stokes flow

The incompressible Stokes equation is derived from the incompressible Navier-Stokes equation by neglecting the nonlinear convective term. The momentum equation and the continuity equation of the unsteady Stokes flow are:

$$\begin{cases} \rho \dfrac{\partial \vec{u}}{\partial t} = -\nabla P + \mu \nabla^2 \vec{u} \\ \nabla \bullet \vec{u} = 0 \end{cases} \tag{4.1}$$

Here ρ is the density of the fluid, u is the velocity, μ is the viscosity, and P is the pressure. If the fluid is assumed to be steady, or if the fluid motion is purely perturbed by the motions of the micro-scale devices and not by the outside excitations (this guarantees that the velocity is small), scale analysis indicates that the acceleration term and the nonlinear convective term in the momentum equation are both negligibly small, and the result is the governing equation of the steady Stokes flow without body forces:

$$\begin{cases} -\nabla P + \mu \nabla^2 \vec{u} = 0 \\ \nabla \bullet \vec{u} = 0 \end{cases} \tag{4.2}$$

In order to derive the boundary integral equation, it is necessary to find the fundamental solutions or the Green's functions of the steady Stokes problem first. We assume there is a point force at $\vec{x}_0$ in the free space with strength $\vec{g}\delta(\vec{x}\text{-}\vec{x}_0)$. The corresponding governing equations are:

$$\begin{cases} -\nabla P + \mu \nabla^2 \vec{u} + \vec{g}\delta(\vec{x} - \vec{x}_0) = 0 \\ \nabla \bullet \vec{u} = 0 \end{cases} \tag{4.3}$$

Solving the above equations yields the fundamental solutions of velocity, pressure, and stress tensor due to the point force [14, 24]:

$$
\begin{aligned}
u_i(\vec{x}) &= \frac{1}{8\pi\mu} G_{ij}(\vec{x},\vec{x}_0) g_j, \\
P(\vec{x}) &= \frac{1}{8\pi} p_j(\vec{x},\vec{x}_0) g_j, \\
\sigma_{ik}(\vec{x}) &= \frac{1}{8\pi} T_{ijk}(\vec{x},\vec{x}_0) g_j,
\end{aligned}
\tag{4.4}
$$

where

$$
\begin{aligned}
G_{ij}(\vec{x},\vec{x}_0) &= \frac{\delta_{ij}}{r} + \frac{\hat{x}_i \hat{x}_j}{r^3}, \\
p_i(\vec{x},\vec{x}_0) &= 2\frac{\hat{x}_i}{r^3}, \\
T_{ijk}(\vec{x},\vec{x}_0) &= -6\frac{\hat{x}_i \hat{x}_j \hat{x}_k}{r^5}, \\
r &= |\hat{x}|, \quad \hat{\mathrm{x}} = \vec{x} - \vec{x}_0.
\end{aligned}
\tag{4.5}
$$

Note the repeated indices are Einstein summations. The stress σ_{ik} of the fluid is defined as:

$$
\sigma_{ik} = -P\delta_{ik} + \mu\left(\frac{\partial u_i}{\partial x_k} + \frac{\partial u_k}{\partial x_i}\right) \tag{4.6}
$$

and the surface force is $f_i = \sigma_{ik} n_k$.

Next, we derive the velocity boundary integral equation using the Lorentz reciprocal identity

$$
\frac{\partial}{\partial x_j}\left(u_i' \sigma_{ij} - u_i \sigma_{ij}'\right) = u_i' \frac{\partial \sigma_{ij}}{\partial x_j} - u_i \frac{\partial \sigma_{ij}'}{\partial x_j} = 0 \tag{4.7}
$$

where u_i and σ_{ij} are the velocity and stress tensor of flow A, and u_i' and σ_{ij}' are the velocity and stress tensor of flow B. It is assumed that both "flows" are solutions of the same fluid domain due to different boundary conditions. In addition, there is no force inside the fluid domain. The Lorentz reciprocal identity is also called the Green's second identity applied to the Stokes flow problem. It is a powerful tool because it gives information about an unsolved flow A (stress or velocity solutions), based on the solutions of another flow B and the boundary conditions of flow A. When applied to deriving the boundary integral equations, the velocity and stress solutions of flow B are the fundamental solutions of the Stokes equation. We integrate (4.7) over the entire 3-D fluid domain and apply the divergence theorem to convert the volume integral to a surface integral. The velocity integral equations for a point $\vec{x}_0$ outside the fluid domain, inside the fluid domain, and right on the fluid boundary D are

$$\begin{cases} \vec{x}_0 \text{ outside fluid} & -\int_D G_{ij}(\vec{x}_0,\vec{x}) f_i(\vec{x}) ds(\vec{x}) + \mu \int_D u_i(\vec{x}) T_{ijk}(\vec{x}_0,\vec{x}) n_k(\vec{x}) ds(\vec{x}) = 0 \\ \vec{x}_0 \text{ inside fluid} & u_j(\vec{x}_0) = -\frac{1}{8\pi\mu}\int_D G_{ij}(\vec{x}_0,\vec{x}) f_i(\vec{x}) ds(\vec{x}) + \frac{1}{8\pi}\int_D u_i(\vec{x}) T_{ijk}(\vec{x}_0,\vec{x}) n_k(\vec{x}) ds(\vec{x}) \\ \vec{x}_0 \text{ right on D} & u_j(\vec{x}_0) = -\frac{1}{4\pi\mu}\int_D G_{ij}(\vec{x}_0,\vec{x}) f_i(\vec{x}) ds(\vec{x}) + \frac{1}{4\pi}\int_D{}^{PV} u_i(\vec{x}) T_{ijk}(\vec{x}_0,\vec{x}) n_k(\vec{x}) ds(\vec{x}) \end{cases} \tag{4.8}$$

where f_j in the above equation is the surface force distribution, G_{ij} is the single-layer kernel of the velocity integral equation, T_{ijk} is the double-layer kernel, $\vec{x}_0$ is the position of the field point, and $\vec{x}$ is the position of the source point. Note that the integral at infinity is zero since velocity and stress decrease at the speed of r^{-1} and r^{-2}, respectively. So the above integral equations also work for problems in an infinite fluid domain. The superscript PV indicates the principal value when the point is right on the boundary and the double-layer jump is not considered. The double-layer jump on smooth surfaces is:

$$\frac{1}{8\pi}\int_D T_{ijk}(\vec{x}_0,\vec{x})n_k(\vec{x})ds(\vec{x}) = \delta_{ij}\times\begin{cases}1 & \text{inside D}\\ 1/2 & \text{right on D}\\ 0 & \text{outside D}\end{cases} \tag{4.9}$$

The double layer jump is a mathematical jump that describes the discontinuity due to dipole distributions. Equation (4.9) further indicates that the double-layer integral of a non-deforming closed surface (velocity is the same so that they can be taken out of the double layer integral sign of (4.8)) is zero if the evaluation point is in the fluid domain (outside the body surface D in (4.9)). If the evaluation point approaches the boundary from the side of the fluid domain (outside D) until it is "right on" the boundary D, we have to keep the solution continuous and apply the value of the "outside D" case, which is zero. So we simply neglect the double-layer integral since most of the problems we want to model are related to rigid bodies. Then we have a single-layer velocity integral equation that works both inside the fluid and "right on" the boundary defined by the rigid body:

$$\begin{aligned} u_i(\vec{x}_0) &= -\frac{1}{8\pi\mu}\int G_{ij}(\vec{x}_0,\vec{x})f_j(\vec{x})ds(\vec{x})\\ G_{ij}(\vec{x}_0,\vec{x}) &= \frac{\delta_{ij}}{r}+\frac{\hat{x}_i\hat{x}_j}{r^3}\\ r &= |\hat{x}|,\quad \hat{\mathrm{x}} = \vec{x}_0-\vec{x} \end{aligned} \tag{4.10}$$

The corresponding pressure integral equation is

$$\begin{aligned} P(\vec{x}_0) &= -\frac{1}{8\pi}\int p_i(\vec{x}_0,\vec{x})f_i(\vec{x})ds(\vec{x})+P_\infty\\ p_i(\vec{x}_0,\vec{x}) &= 2\frac{\hat{x}_i}{r^3} \end{aligned} \tag{4.11}$$

Again, the single-layer integral expresses velocity due to point force distribution f_j on the boundary. We only give the derivations for translational motions here; similar derivations are also applicable to the rotational motions. Therefore, the velocity integral equations above are

valid as long as the fluid motion is caused by the motions of rigid bodies. However, the double-layer integrals are not necessarily zero in the flexible body case.

4.3 Unsteady Stokes flow integral equations

In the unsteady case, the governing equations (4.1) are still linear. In order to meet the popular demand for frequency response solutions, we have chosen to create a frequency domain solver. We first assume the motion of the fluid is generated by the oscillating motion of the devices; we then apply a time-harmonic assumption to (4.1) and assume all variables are sinusoidal functions of the oscillation frequency. Given an oscillating point force at $\vec{x}_0$ with strength $\vec{g}e^{i\omega t}\delta(\vec{x}\text{-}\vec{x}_0)$, the velocity and pressure are assumed to be $\vec{u}e^{i\omega t}$ and $Pe^{i\omega t}$. Equation (4.1) in the frequency domain is:

$$\begin{cases} \rho_0 i\omega u_j = -\nabla_j P + \mu\nabla^2 u_j + g_j\delta(\vec{x}\text{-}\vec{x}_0) \\ \nabla_j u_j = 0 \end{cases} \tag{4.12}$$

Solving the above equations leads to the fundamental solutions of the unsteady incompressible Stokes flow. The integral equations of the unsteady Stokes equation are similar to those of the steady one, but with frequency-dependent kernels. A frequency parameter is defined as $\lambda^2 = {}^{-i\omega\rho}/_{\mu}$, where ω is the oscillation frequency of the object that induces the fluid motion, and $i = \sqrt{-1}$. The detailed derivations of the integral equations are not given in this section. We will focus on the derivation procedures in Chapter 8 when we derive the fundamental solutions of the linearized compressible Stokes integral equation. Here, we simply list the single-layer velocity integral equation. Discussions and comparisons with the linearized compressible Stokes equation will also be given in Chapter 8. The velocity integral equation of the unsteady Stokes flow is given by Pozrikidis [24]:

$$u_i(\vec{x}) = -\frac{1}{8\pi\mu}\int G_{ij}(\vec{x},\vec{x}_0) f_j(\vec{x}_0) ds(\vec{x}_0)$$

$$G_{ij}(\hat{x}) = A(R)\frac{\delta_{ij}}{r} + B(R)\frac{\hat{x}_i \hat{x}_j}{r^3}$$

where

$$A = 2e^{-R}(1+\frac{1}{R}+\frac{1}{R^2}) - \frac{2}{R^2}$$

$$\mathrm{B} = -2e^{-R}(1+\frac{3}{R}+\frac{3}{R^2}) + \frac{6}{R^2} \qquad (4.13)$$

$$R = \lambda r$$

The solution of surface forces for the unsteady Stokes equation is also a function of frequency, and it is usually complex. The real part of the surface forces is the damping force solution, while the imaginary part is the spring force solution. Because of the mass acceleration term in the momentum equation, the forces are not necessarily in phase with the velocity. The pressure integral equation and kernel are still the same.

A rather difficult issue in solving the unsteady Stokes integral equation is the kernel integration, because the kernel in (4.13) is a fast-decaying oscillating kernel. The kernel, though it looks like a mix of weakly singular kernels, strongly singular kernels, and hypersingular kernels, is actually just a weakly singular kernel because of the cancellations when $r \to 0$. This can be shown by a series expansion. However, on the other hand, the cancellation makes numerically integrating the kernel very difficult. Some numerical procedures are unstable. The kernel integration will be discussed in Chapter 8 when we compare it with the even more complicated velocity kernel of the linearized compressible Stokes equation.

The unsteady Stokes equation has more applications than the steady Stokes equation. But in general, the quasi-static assumption is accurate enough for many air-packaged MEMS devices. When the frequency is low and the mass acceleration term makes trivial

contributions, the real part of the unsteady Stokes solution converges to the solution of the steady Stokes equation. The imaginary part, as can be expected, is almost zero.

4.4 The PFFT algorithm for the Stokes problem

The PFFT algorithm described in the previous chapter works in both one-variable (scalar) and multi-variable (vector) problems. But the memory and CPU time usages will not be minimized without an optimal organization. This section focuses on a special topic of the PFFT algorithm when applied to the Stokes problem, that is, sequencing the FFTs and IFFTs.

The Stokes velocity integral equation has three velocity components, three force components, and nine kernels. Among the nine kernels, only six kernels are independent since $G_{ij} = G_{ji}$. Given U, the GMRES algorithm is applied in FastStokes to calculate F using the equation $U = GF$. The PFFT algorithm is further applied to accelerate the matrix-vector multiplications of GMRES. The FFTs of the grid kernels $\tilde{G}_{jk}^{\text{grid}} = FFT(G_{jk}^{\text{grid}})$ are calculated and stored so that they can be used repeatedly. The projection, interpolation, and pre-correction operators are still the same in the multi-variable problem; they are applied repeatedly to all variables.

The second step of the PFFT algorithm is a convolution step which includes 2 FFTs (to be more precise, one FFT and one IFFT) in the scalar case, and this is a computationally expensive step. A naive approach of duplicating the same idea and applying it in the Stokes problem (a vector case) can be described as

$$U_j^{\text{grid}} = \sum_{k=1}^{3} IFFT\left(\tilde{G}_{jk}^{\text{grid}} \cdot FFT\left(F_k^{\text{grid}}\right)\right) \tag{4.14}$$

A total of 18 FFTs are needed for every matrix-vector calculation step. A more efficient way that avoids repeating the calculations of $FFT(F_j^{\text{grid}})$ is to save the result of

$\tilde{F}_j^{\text{grid}} = FFT\,(F_j^{\text{grid}})$ in every GMRES iteration so that it can be used repeatedly. Also, only one IFFT is needed for every grid velocity calculation if the following scheme is used:

$$\begin{aligned} \tilde{F}_j^{\text{grid}} &= FFT(F_j^{\text{grid}}) \\ U_j^{\text{grid}} &= IFFT\left(\sum_{k=1}^{3} \left(\tilde{G}_{jk}^{\text{grid}} \cdot \tilde{F}_j^{\text{grid}}\right)\right) \end{aligned} \tag{4.15}$$

A total of only 6 FFTs (3FFTs in the first equation and 3 IFFTs in the second) are needed in each matrix-vector product calculation [49]. This is a dramatic computational cost reduction compared with 18 FFTs. This method is shown schematically in Figure 4.1.

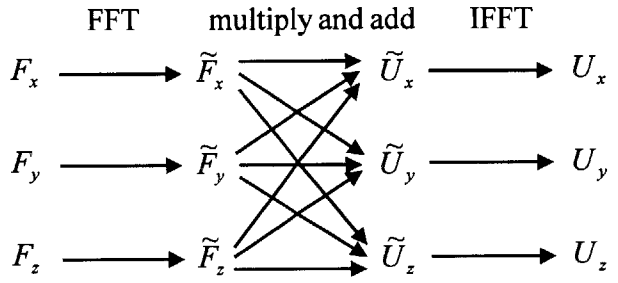

Figure 4.1: An efficient sequence of FFTs and IFFTs

Chapter 5

A curved panel integration algorithm

In some cases, discretizing curved surfaces using flat panels might cause dramatic discretization error. If curved panels are used, an accurate curved panel integration algorithm is needed to avoid erasing the accuracy gains of using curved panels in the discretization. This chapter presents a new approach to computing $\frac{1}{r}$ singularities over curved panels. By using carefully chosen mapping techniques, a curved panel with curved edges is mapped to a flat panel with straight edges. Analytical formulas for flat-panel integrals are then applied to give approximate curved-panel integration solutions. For those curved panels with reasonable curvatures and smooth edges, this method can efficiently achieve excellent accuracy (10^{-5}% error). A simple sphere example is given at the end of this chapter to show the advantage of using curved panels.

5.1 A mapping method

In this chapter, we consider the example of computing $\iint \frac{1}{r} ds$ over a curved triangular panel using a mapping technique, though other functions can be integrated in a similar way.

Since most geometrically well-defined curved panels with reasonably large curvatures can be accurately mapped to flat panels with straight edges, it is possible to introduce a reference flat panel and integrate a slightly different function over the flat panel. The key point here is how to construct a mapping that accurately maps the curved panel to a straight-edged flat panel. Because the $\frac{1}{r}$ integrand is a weakly singular function, approximations that are not accurate

around the evaluation point may lead to large numerical errors, especially when the evaluation point is on or close to the panel. A successful approximation should be accurate enough near the singularity; therefore, singular functions are ideal candidates for approximating other singular functions. We might be able to use the flat-panel solutions calculated using the analytical method described in Chapter 3 to calculate an approximate curved-panel integration solution.

Given a curved triangular panel, a straightforward choice of reference flat panel is the flat panel defined by the three corners, though it may not be a good choice. The integration over a curved panel may be approximately expressed as:

$$\int_{\text{curve}} \frac{1}{r_{\text{eval-curve}}} ds_{\text{curve}} = \int_{\text{flat}} \frac{1}{r_{\text{eval-flat}}} \left(\frac{r_{\text{eval-flat}}}{r_{\text{eval-curve}}} |J| \right) ds_{\text{flat}} \approx \int_{\text{flat}} \frac{P(\xi,\eta)}{r_{\text{eval-flat}}} ds_{\text{flat}} \tag{5.1}$$

where $r_{\text{eval-curve}}$ is the distance from the evaluation point to a point on the curved panel, $r_{\text{eval-flat}}$ is the distance form the evaluation point to the corresponding point on the flat panel, J is the Jacobian matrix, $P(\xi,\eta)$ is a polynomial approximation of $\frac{r_{\text{eval-flat}}}{r_{\text{eval-curve}}}|J|$, and $\int_{\text{flat}} \frac{P(\xi,\eta)}{r_{\text{eval-flat}}} ds_{\text{flat}}$ is known to be solvable using the analytical method described in Chapter 3. If $\frac{1}{r_{\text{eval-flat}}}$ is the only singular function and $\frac{r_{\text{eval-flat}}}{r_{\text{eval-curve}}}|J|$ is smooth near the singularity, it will be easy to find a polynomial $P(\xi,\eta)$ that accurately approximates $\frac{r_{\text{eval-flat}}}{r_{\text{eval-curve}}}|J|$. Therefore the ultimate goal is to find a reference flat panel that makes $\frac{r_{\text{eval-flat}}}{r_{\text{eval-curve}}}|J|$ as smooth as possible.

Figure 5.1 shows a curved panel and two reference flat panels. A tangent panel that touches the curved panel at a point closest to the evaluation point on the panel is an ideal choice. In the singular case, the evaluation point that is on the surface is also the tangent point. In the

near-singular case, the tangent point is chosen as shown in Figure 5.2; it is the point on the surface closest to the evaluation point.

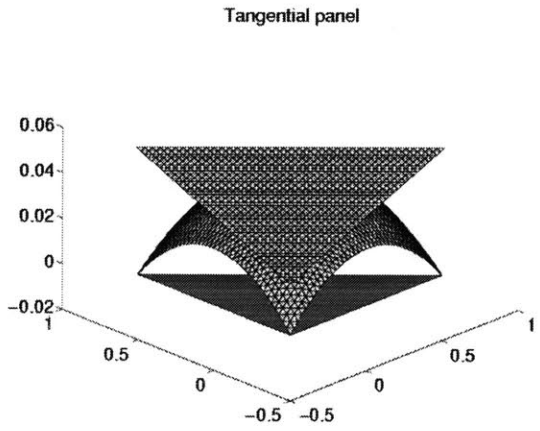

Figure 5.1: The tangent panel

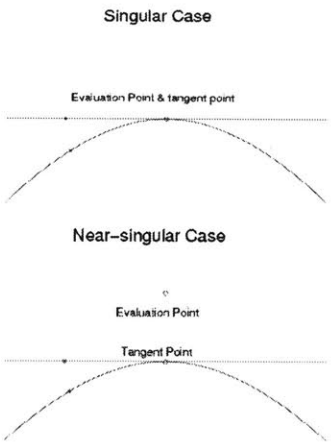

Figure 5.2: Evaluation points and tangent points

When the curved panel is mapped to the reference flat panel, the following limit exists when the point on the curved panel approaches the tangent point (the corresponding point on the reference flat panel moves together with it),

$$\lim_{\text{point}\to\text{tangent point}} \frac{r_{\text{eval-flat}}}{r_{\text{eval-curve}}} = 1 \tag{5.2}$$

Note that the determinant of the Jacobian of the mapping $|J|$ is usually a smooth function, so the ideal mapping we are looking for is a mapping between the curved panel and a tangent panel that coincides at the tangent point. This tangent panel is called the "ideal reference flat panel" in this chapter.

5.2 Polynomial approximation

A second problem is how to calculate the polynomial $P(\xi,\eta)$. We suggest using the reference panel to set up the local coordinate system. The mapping between the curved panel and the reference flat panel is defined as

$$x = x(\xi,\eta),\ y = y(\xi,\eta), \text{ and } z = z(\xi,\eta) \tag{5.3}$$

and the determinant of the Jacobian is

$$|J| = \sqrt{\left(\frac{\partial x}{\partial \xi}\frac{\partial y}{\partial \eta} - \frac{\partial y}{\partial \xi}\frac{\partial x}{\partial \eta}\right)^2 + \left(\frac{\partial y}{\partial \xi}\frac{\partial z}{\partial \eta} - \frac{\partial z}{\partial \xi}\frac{\partial y}{\partial \eta}\right)^2 + \left(\frac{\partial z}{\partial \xi}\frac{\partial x}{\partial \eta} - \frac{\partial x}{\partial \xi}\frac{\partial z}{\partial \eta}\right)^2} \tag{5.4}$$

In the global system, the evaluation point is (X',Y',Z') and the point on the curved panel is $(x(\xi,\eta), y(\xi,\eta), z(\xi,\eta))$. In the local system, the evaluation point is (X,Y,Z) and the point on the flat panel is $(\xi,\eta,0)$. For the i^{th} point on the curved panel,

$$P(\xi_i,\eta_i) = \frac{r_{\text{eval-flat}}}{r_{\text{eval-curve}}}|J| = \frac{\sqrt{(X-\xi_i)^2 + (Y-\eta_i)^2 + Z^2}}{\sqrt{(X'-x(\xi_i,\eta_i))^2 + (Y'-y(\xi_i,\eta_i))^2 + (Z'-z(\xi_i,\eta_i))^2}}|J| \tag{5.5}$$

$P(\xi,\eta)$ can be analytically expanded and truncated to finite terms if the curved surface has an analytical expression. Here we suggest using a cubature method. This method first finds cubature points (ξ_i,η_i) on the reference panel and its corresponding $\frac{r_{\text{eval-flat}}}{r_{\text{eval-curve}}}|J|$, then it calculates the coefficients of the polynomial by forcing the value of the polynomial to match $\frac{r_{\text{eval-flat}}}{r_{\text{eval-curve}}}|J|$ at those cubature points. If more cubature points are used, a least square method can be applied to compute the coefficients of the polynomial.

More precisely, assume

$$P(\xi,\eta)=c_1+c_2\xi+c_3\eta+c_4\xi\eta\ldots\ldots c_k\xi^m\eta^n, \tag{5.6}$$

and then calculate the polynomial coefficients by solving

$$\begin{bmatrix} 1 & \xi_1 & \eta_1 & \cdots & \xi_1^m\eta_1^n \\ 1 & \xi_2 & \eta_2 & \cdots & \xi_2^m\eta_2^n \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 1 & \xi_k & \eta_k & \cdots & \xi_k^m\eta_k^n \end{bmatrix} \begin{bmatrix} c_1 \\ c_2 \\ c_3 \\ \vdots \\ c_k \end{bmatrix} = \begin{bmatrix} P(\xi_1,\eta_1) \\ P(\xi_2,\eta_2) \\ \vdots \\ P(\xi_k,\eta_k) \end{bmatrix}. \tag{5.7}$$

Sometimes it is difficult to find the tangent panel analytically; a short iteration may be used in such cases. Mapping is the critical step of this curved-panel integration algorithm, so the curved surface must be uniquely mapped to the flat surface. Easily made mistakes are surface overlaps (parts of the curved surface can be mapped to two or more flat panels) and holes (parts of the curved surface cannot be mapped to any flat panels).

5.3 Curved-panel integration algorithm

The curved-panel integration algorithm for computing $\iint\frac{1}{r}ds$ over a curved panel can be summarized as four major steps:

1.0) Calculate tangent point and tangent flat panel.

2.0) Find a mapping between curved panel and ideal reference flat panel.

3.0) Compute cubature points.

4.0) Perform curved-panel integration:

4.1) Compute $\iint_{\text{flat}} \frac{1}{r} ds$, $\iint_{\text{flat}} \frac{\xi}{r} ds$, $\iint_{\text{flat}} \frac{\eta}{r} ds, \cdots \iint_{\text{flat}} \frac{\xi^m \eta^n}{r} ds$ analytically

4.2) Calculate polynomial expansion $P(\xi,\eta) = c_1 + c_2\xi + c_3\eta + c_4\xi\eta \ldots\ldots c_k\xi^m\eta^n$, for

$$\int_{\text{curve}} \frac{1}{r_{\text{eval-curve}}} ds_{\text{curve}} = \int_{\text{flat}} \frac{1}{r_{\text{eval-flat}}} \left(\frac{r_{\text{eval-flat}}}{r_{\text{eval-curve}}} |J| \right) ds_{\text{flat}} = \int_{\text{flat}} \frac{P(\xi,\eta)}{r_{\text{eval-flat}}} ds_{\text{flat}}$$

4.3) Compute

$$\int_{\text{flat}} \frac{P(\xi,\eta)}{r_{\text{eval-flat}}} ds_{\text{flat}} = c_1 \int_{\text{flat}} \frac{1}{r_{\text{eval-flat}}} ds_{\text{flat}} + c_2 \int_{\text{flat}} \frac{\xi}{r_{\text{eval-flat}}} ds_{\text{flat}} + c_3 \int_{\text{flat}} \frac{\eta}{r_{\text{eval-flat}}} ds_{\text{flat}} \ldots\ldots + c_k \int_{\text{flat}} \frac{\xi^m \eta^n}{r_{\text{eval-flat}}} ds_{\text{flat}}$$

This method is very accurate if the value of $\frac{r_{\text{eval-flat}}}{r_{\text{eval-curve}}} |J|$ is smooth enough.

5.4 Accuracy testing and mapping improvements

A simple curved triangle panel that is part of a sphere is used to test the accuracy of integrating $\iint_{\text{curve}} \frac{1}{r} ds$ using this mapping method. The radius of the panel varies from 2~7.5, and the distances between corners are 1, 1, and $\sqrt{2}$. Note that the larger the radius, the "flatter" the panel. First, the singular case (the evaluation point is the centroid of the curved panel) and the nearby case (the evaluation point is the centroid of a nearby panel) are tested. Figures 5.3 and 5.4 show that very good accuracy is achieved using this algorithm, even with a low-order polynomial approximation.

Second, the near-singular performance of the mapping method is tested. Evaluation points are chosen along the line that connects the center of the sphere and the circumcenter of the flat panel defined by the three corners. Figure 5.5 shows that the accuracy decreases and then

increases. This is caused by the mapping error. Figure 5.6 shows $\frac{r_{\text{eval-flat}}}{r_{\text{eval-curve}}} = 1$ at the near-singular point, and it is close to 1 all over the panel. But a scaling reveals a peak that cannot be easily fitted with a polynomial. Of course, the accuracy of the mapping method will not be bad, since $\frac{r_{\text{eval-flat}}}{r_{\text{eval-curve}}} \sim 1$ all over the panel, but it will not be very good unless a large number of cubature points and a high-order polynomial are used. In the singular case, $\frac{r_{\text{eval-flat}}}{r_{\text{eval-curve}}}$ strictly increases or decreases when a point on the panel moves farther away from the singularity. When the evaluation point is far away from the panel, $\frac{r_{\text{eval-flat}}}{r_{\text{eval-curve}}}$ is also smooth enough to be accurately fitted to a polynomial. The difficulty only occurs when the evaluation point is neither far from the panel nor very close to the panel. Figure 5.5 shows that using a large number of cubature points and a high-order polynomial can achieve good accuracy but at high cost.

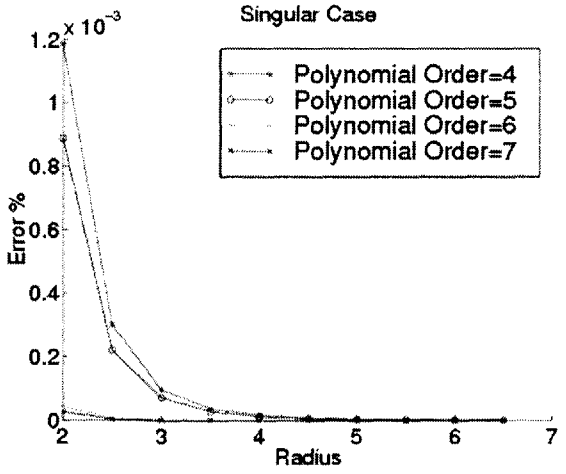

Figure 5.3: Singular case accuracy

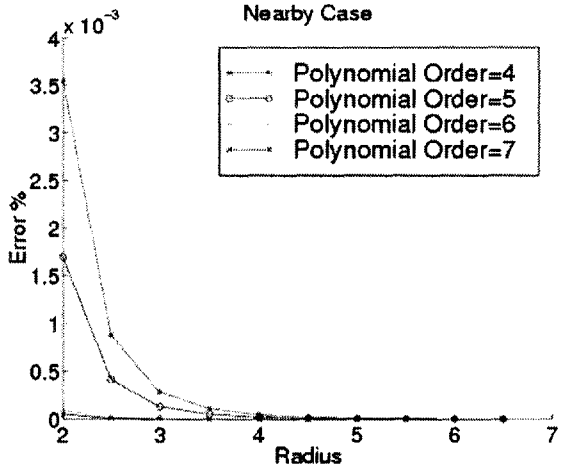

Figure 5.4: Nearby case accuracy

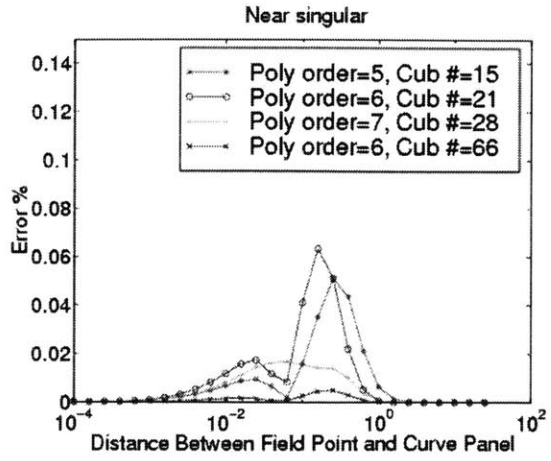

Figure 5.5: Near-singular accuracy

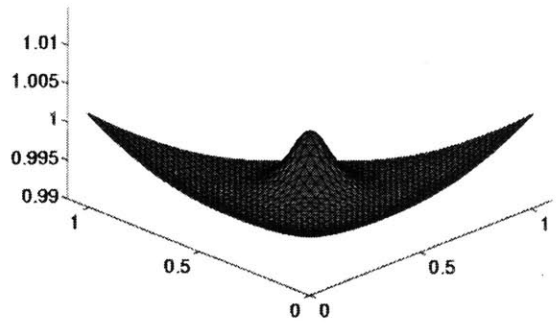

Figure 5.6: Value of $\frac{r_{\text{eval-flat}}}{r_{\text{eval-curve}}}$

The hat-shape of $\frac{r_{\text{eval-flat}}}{r_{\text{eval-curve}}}$ at the near-singular area reflects a problem in the mapping. Therefore, modifying the mapping may improve the accuracy. The method used here is keeping the tangent point fixed, while scaling (enlarging or shrinking) the reference flat panel. Figure 5.7 shows $r_{\text{eval-flat}} \neq r_{\text{eval-curve}}$ in the old mapping. A very short iteration procedure is used to perform the scaling: each time, the panel is enlarged so that the smallest $\frac{r_{\text{eval-flat}}}{r_{\text{eval-curve}}}$ at those cubature points increases to 1 (see Figure 5.8). Figure 5.9 shows that the modified mapping gets rid of the hat-shape shown in Figure 5.6. Figure 5.10 shows that the modification significantly increases accuracy.

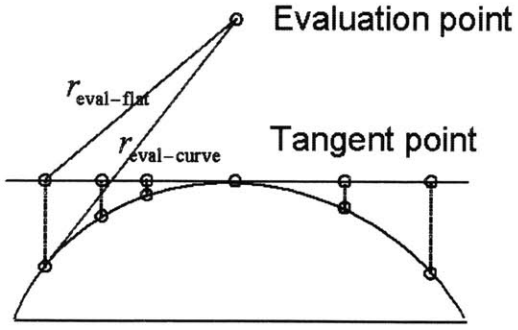

Figure 5.7: Old mapping

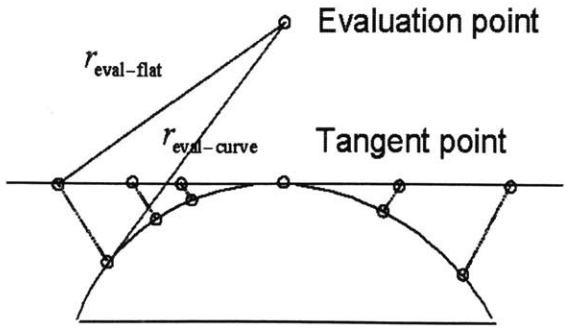

Figure 5.8: Modified mapping

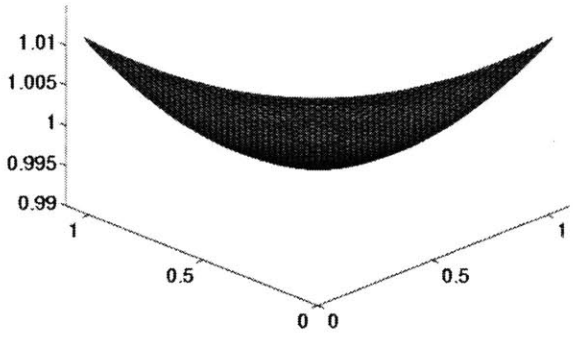

Figure 5.9: Hat-shape disappears

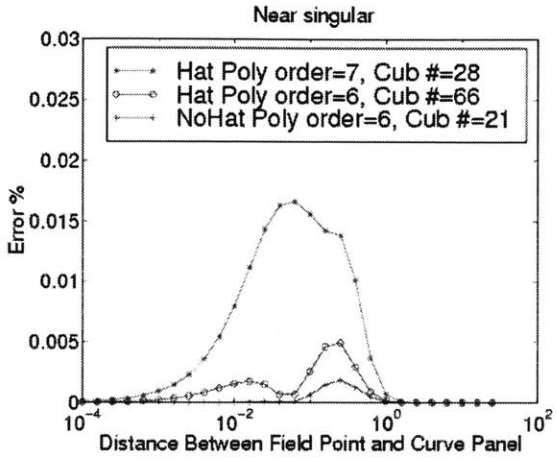

Figure 5.10: No-hat increases accuracy

5.5 Discretization error reductions of using curved panels

To demonstrate the advantage of using curved panels, consider computing the capacitance of a sphere when the charge distribution is uniform. Table 5.1 compares the capacitance computed using a flat-panel geometric approximation with that using curved panels and our new integration method. As is clear from the table, the curved panels yield fifty times the accuracy with a tenth of the panels.

Table 5.1: A simple example: capacitance of a sphere

Panels used	Error
48 Curved Panel	0.0127%
48 Flat Panel	8.659%
768 Flat Panel	0.679%

Chapter 6

Null space of the incompressible Stokes integral equation

The Stokes equation only has a pressure derivative term, which implies that a constant pressure can be added to the solution. Therefore, the equation does not have a unique solution without a proper pressure boundary condition. The corresponding singular mode of this equation is the constant-pressure zero-velocity solution, which does not affect the total force on a single rigid body. However, the singularity can impact the results produced by a numerical procedure, especially in multi-body problem simulations [44]. In addition, the detailed fluid forces will not be computed correctly unless the singular mode is treated properly.

We start this chapter by introducing the null space problem of a single rigid body, followed by detailed discussions. A two-step method is presented in later sections: the first step solves the null-space-free solution, and the second step recovers the correct solution from the null-space-free solution by adjusting the magnitudes of the null-space components. With this approach, the final solution is unique and correct.

6.1 Null space of a single rigid body

For a single rigid body, constant pressure around it generates zero total force, and therefore, no induced velocity. The forces produced by constant pressure are only nonzero in the normal direction, that is,

$$\vec{f} = -P\vec{n} \tag{6.1}$$

where $\vec{n}$ is the surface out normal. Substituting (6.1) into the single-layer integral equation (4.10) and integrating over the surface yields:

$$\int_{\text{surface}} G_{ij} n_j ds = 0 \tag{6.2}$$

If the single-layer integral in (4.10) is viewed as an operator applied to the surface force, the surface normal is the singular mode of this integral operator. At the discretized level, the integral operator is the G matrix and $0 = GN$ exists for every closed surface, where N is the surface normal vector. Obviously, N is the null-space vector of the G matrix. If a direct matrix inverse is applied to calculate the surface force vector F from $U = GF$, numerical error is unavoidable since G is a singular matrix. The following is the solution of F using a direct matrix inverse

$$F = F^{\text{correct solution}} + XN + \varepsilon \tag{6.3}$$

where X represents a large number generated by the numerical procedure, and ε is the numerical error. As can be imagined: if X is too big, noise ε may be big enough to obliterate the correct solution $F^{\text{correct solution}}$, and then recovering an accurate solution $F^{\text{correct solution}}$ from $F = F^{\text{correct solution}} + XN + \varepsilon$ would be impossible.

For now, we simply assume that there is an accurate numerical approach that generates a noise-free solution $F = F^{\perp} + YN$. We change the notation here because the null space is physically a piece of missing information which is not given by the velocity integral equation, so one should not expect to have an $F^{\text{correct solution}}$ by solving the velocity integral equation. A possible solution is $F^{\perp}$, which assumes the magnitude of the null-space component to be zero. In other words, $F^{\perp}$ is perpendicular to the null space.

6.2 Multi-body Problem

The missing pressure boundary condition is only one problem we need to face, and the solution is adding a proper boundary condition to the integral equations. Therefore, we need the pressure integral equation that has been largely forgotten by many researchers. In addition, since two integral equations are derived from the Stokes partial differential equation, a correct solution needs to satisfy not only both integral equations, but also the constraints that describe the relationships between the variables, such as $\vec{f}(\vec{x}) = -P(\vec{x})\vec{n}(\vec{x})$ on the surface of a rigid body. Such a relationship is the key to success, because the velocity integral equation and the pressure integral equation are relatively decoupled, and also because both the velocity integral operator and the pressure integral operator are singular.

Fortunately, we know the null spaces of the integral operators and we know that the solution is in a format $F = F^{\perp} + \sum_{i=1}^{\text{number of objects}} Y_i N_i$ if the numerical noise is well under control, where N_i is the null-space vector of the i^{th} rigid body. Calculating the null-space-free solution $F^{\perp}$ can be a starting point of our solution scheme. For now, we need to further explore some key points of this topic to provide a clearer picture.

6.2.1 The null space of the pressure integral operator

Surprisingly, the pressure integral operator is also a singular operator with the same null space, which is the surface normal. The definition of surface force for incompressible flow is

$$\begin{aligned} f_{3\times 1} &= \sigma_{3\times 3} n_{3\times 1} \\ \sigma_{3\times 3} &= -PI_{3\times 3} + \mu\left[\nabla\vec{u} + (\nabla\vec{u})^T\right] \end{aligned} \tag{6.4}$$

where $f_{3\times1}$ is the surface force vector with three components in $x, y,$ and z directions respectively, and $\sigma_{3\times3}$ is the stress tensor of the fluid. Equation (6.4) implies that $\vec{f} = -P\vec{n}$ if $\vec{u} = 0$. Note $\vec{f}$ does not appear explicitly in the Stokes equation, but it is a variable of the Stokes integral equations.

Suppose the entire domain bounded at infinity is filled with fluid, with an imaginary closed boundary defined by the surface of a rigid body (see Figure 6.1). The single-layer velocity and pressure integral equation represent velocity and pressure due to point force distribution on the boundary. Suppose we put normal direction point force distribution with value $-P$ on the boundary, that is $\vec{f} = -P\vec{n}$. As can be imagined and calculated from the velocity integral equation: the velocity is zero everywhere. The point forces put on the boundary only act on the inside fluid, which is incompressible, so the total effect of the point force distribution is zero velocity everywhere, with a constant pressure P inside the boundary and a zero pressure outside the boundary. The pressure jump across the boundary is $-P$. This layer jump is almost the same as the jump in the double-layer velocity integral; both kernels are strongly singular kernels. Mathematically, we can imagine that every point force is evenly split into two halves with the boundary right in the middle; so the pressure right on the boundary is $\frac{P}{2}$.

Apply the pressure integral equation:

$$P = -\frac{1}{8\pi}\int p_i f_i ds = -\frac{1}{8\pi}\int p_i P n_i ds$$

$$\rightarrow \quad -\frac{1}{8\pi}\int p_i n_i ds = \begin{cases} 0 & \text{outside D} \\ 0.5 & \text{right on D} \\ 1 & \text{inside D} \end{cases} \tag{6.5}$$

The double-layer jump makes the pressure non-continuous across the surface. As mentioned before, a well-accepted approach is to neglect the undesired side and keep the value on the boundary continuous with the other side. This is also physically meaningful because the right boundary is not "right on" the boundary but rather infinitely close to the boundary from the

fluid side. Therefore, the value of the double layer jump is zero because the fluid is outside the boundary defined by the surfaces of the rigid bodies. To avoid making the final equations too complicated, we handle the layer jump numerically by modifying the kernel integration algorithm slightly. By doing so, we will not have to discuss the double-layer jump topic again, since it is numerically incorporated in the kernel integration algorithm. Equation (6.5) can be proven both numerically and analytically.

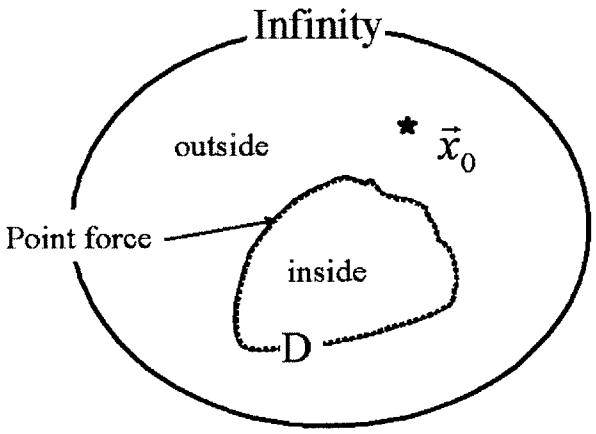

Figure 6.1: Fluid domain with a boundary D defined by the surface of a rigid body

After taking the double-layer jump into consideration, we conclude that the pressure integral operator is also a singular operator, and the null-space vector is also the surface normal.

6.2.2 The relationship between the velocity and pressure integral operators

The velocity and pressure integral operators are both singular with the same null space, and they both operate on the surface force distribution. A common approach to calculate the surface force is to use the velocity integral equation first, since both velocity and force have three components. The pressure integral equation is then applied to calculate the pressure, which can be used to modify the force solution.

Suppose we use the velocity integral equation to calculate a force solution. Because of the uncertainty of the null-space component, there are infinite solutions once a proper velocity boundary condition is given (the velocity boundary condition has to satisfy the continuity equation). If those force solutions are substituted into the pressure equation to calculate the pressure, the pressure solution will be correct and unique. This can be proved as follows.

> *There is a correct force solution, and this force solution will generate a unique and correct pressure solution, as can be proved by the definition of the pressure fundamental solution and linear superposition. The only difference between the correct force solution and the wrong solutions that satisfies the velocity integral equation is the null-space component, which is also the null-space component of the pressure integral operators. Since this null-space component will not affect the pressure solution of the pressure integral equation, any force solution that satisfies the velocity integral equation will give a unique pressure solution, which is the correct one.*

6.2.3 The pressure boundary condition

As mentioned before, we need to add a pressure boundary condition to the integral equation. This is the only way to select a correct solution for the incompressible Stokes boundary value problem. We know from the discussion in the previous section that the pressure variation is uniquely determined once the velocity field is given. Because the fundamental pressure solution due to a point force has a r^{-2} type singularity, the pressure variation decreases to zero at infinity at a speed of r^{-2}. Thus, the pressure boundary condition must be an adjustment to this unique pressure variation that is zero at infinity. Either the pressure at a point or the pressure at infinity is a sufficient pressure boundary condition. The correct pressure integral equation with a given ambient pressure P_∞ is

$$P = -\frac{1}{8\pi}\int p_i f_i ds + P_\infty \tag{6.6}$$

If the given pressure boundary condition is not the ambient pressure P_∞, but rather $P_{\text{at a certain point}}$, then the above equation may be applied to calculate the pressure at infinity $P_\infty = \left(P + \frac{1}{8\pi} \int p_i f_i ds \right)\Big|_{(\text{at a certain point})}$.

6.2.4 The relationship between the correct force solution $F^{\text{correct solution}}$ and the null-space-free force solution $F^{\perp}$

Now the picture is almost clear, but we have not given the relationship between the correct solution $F^{\text{correct solution}}$ and the null-space-free force solution $F^{\perp}$. It is easy to prove that the null-space-free force solution $F^{\perp}$ is not the correct force solution $F^{\text{correct solution}}$ using the example shown in Figure 6.2.

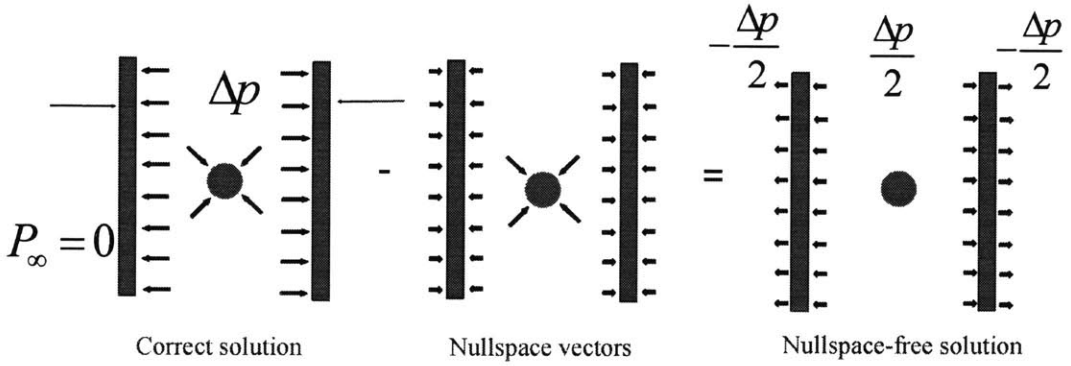

Figure 6.2: The correct force solution and the null-space-free force solution

In this example it is assumed that two long parallel plates are slowly approaching each other so that a high pressure region is generated in the middle; the pressure elsewhere is negligibly small. For the sake of simplicity, we assume the pressure is uniform in the middle. There is also a sphere right in the center. The correct surface force solution is plotted in the left picture in Figure 6.2. It is easily seen that the correct solution is not a null-space-free solution. To make it more dramatic, we subtract the null-space components (in the middle picture) to

reveal the null-space-free solution (in the right picture). The null-space-free solution not only gives the wrong force distributions on the two plates but also indicates zero force on the sphere, and this is clearly non-physical. This example also shows that a null-space-free solution, although it bears significant force discontinuities across the surfaces, may be modified to recover the correct solution. That is,

$$F^{\text{correct solution}} = F^{\perp} + \sum_{i=1}^{m} C_i N_i \tag{6.7}$$

where m is the number of objects in the system and N_i is the corresponding null-space vector. The factors C_i may not have the same values. In the above example, the corresponding $C_{\text{left plate}}, C_{\text{sphere}}$, and $C_{\text{right plate}}$ are $-\frac{\Delta p}{2}, -\Delta p$, and $-\frac{\Delta p}{2}$ respectively if the surface outward normals are used as null-space vectors.

Therefore, the factors C_i can be used to adjust the normal components of the force vector to make $\vec{f} = -P\vec{n}$ at the surfaces of the rigid objects. After this adjustment step, the solution should be correct and unique.

In a single-object system, the force solution is always continuous since there is only one closed surface. The null-space-free solution is not correct, but it is still sound although it may indicate a wrong pressure at infinity, which violates the pressure boundary condition. The difference between the null-space-free solution and the correct solution is only CN.

In a multi-object system, the null-space-free force solution is not necessarily continuous; the m-independent null-space vectors correspond individually to the surface normal on one object and zero on the others. As can be seen from Figure 6.2, the forces on the sphere are zero, while those on the inner side of the plate are not. These jumps across surfaces are clearly undesired, and may lead to confusion. This must be avoided by doing a pressure adjustment.

6.2.5 Strategy and a short summary

The missing pressure boundary condition, the null space of the singular integral operators, and the surface force that serves as a bridge between the two integral equations make it difficult to compute the correct and unique surface force solution. As a brief summary, the following strategy is presented to handle the singular Stokes BEM operator problem:

a. Apply a numerically robust algorithm to calculate the accurate null-space-free surface force solution. This solution is a finite solution.
b. For each object, apply the pressure integral equation to calculate the pressure at one point on the surface. Then use this pressure solution to adjust the null-space-free solution by adding or subtracting the null-space component so that $f_n = -P$.

Note that the null space of the singular integral operator is the surface normal only, which does not affect the total force on the object. Hence, the first step is good enough if only the net body forces, and not the detailed surface force, are all that are of interest. The null-space components can simply be eliminated in the computation procedure of the first step. Otherwise, it is necessary to perform the second step.

6.3 The modified GMRES algorithm

If the numerical aspect is not treated carefully, solving the singular matrix problem $U = GF$ may cause large numerical errors. The solution may be badly contaminated by the null-space components and numerical noise. A numerically robust procedure is crucial to the success of the two-step strategy presented in the previous subsection.

If a Krylov-subspace-based method, such as GMRES, is applied to solve $U = GF$, then removing the null space of the G matrix can be performed by removing the null space from every Krylov subspace vector since the GMRES algorithm searches the solution iteratively in the Krylov subspace.

$$\text{Krylov Subspace} \quad K_n = [U, GU, G^2U, G^3U ....., G^{n-1}U] \tag{6.8}$$

An easy approach is to perform the null space removing procedure after every matrix-vector product step of GMRES. This guarantees that the null-space vectors will not contaminate the orthogonal vectors generated by the modified Gram-Schmidt orthogonalization step.

But why is the Krylov subspace not perpendicular to the null space? Applying the divergence theorem to the continuity equation, we have

$$\iiint \nabla \bullet \vec{u} dv = \iint \vec{u} \bullet \vec{n} ds = 0 \Rightarrow UN = 0 \tag{6.9}$$

Therefore, the first Krylov subspace vector is perpendicular to the null-space vector. $U = GF$ also indicates that every vector in the range of the G matrix is a valid velocity distribution; hence, the product of the G matrix with a vector is also perpendicular to the null-space vector. This leads us to conclude that the Krylov subspace $\kappa = [U, GU, G^2U, G^3U .....]$ is automatically perpendicular to the null-space vectors. However, this is not true. A numerically calculated G matrix bears non-negligible numerical errors from the numerical procedures such as the kernel integration algorithm. The matrix vector product used in the GMRES algorithm is only approximately calculated using the PFFT algorithm, which also introduces numerical errors. So, the Krylov subspace is not perfectly perpendicular to the null space of the G matrix.

The null space removing procedure is a simple procedure that removes the null-space vector N_i from every Krylov subspace vector (the matrix-vector product) k_j. It is computationally cheap and easy to implement.

$$k_j = k_j - \frac{k_j{}^T N_i}{N_i{}^T N_i} N_i \tag{6.10}$$

In many cases, the numerical errors contaminate the orthogonal vector space generated by the modified Gram-Schmidt orthogonalization step and cause slow convergence. Thus, this null-space removal modification is necessary. It not only leads to a perfect null-space-free solution, but also makes the GMRES algorithm converge faster. In Figure 6.3, the red line labeled "Without Nullspace Remover" represents the norm of the residual of the GMRES iteration when the Null-Space Remover was turned off. The blue line labeled "With Nullspace Remover" represents the norm of residual when the Null-Space Remover was on; it converged much faster after about 30 iterations. Thus, having a null-space remover that is easy to implement is a good choice.

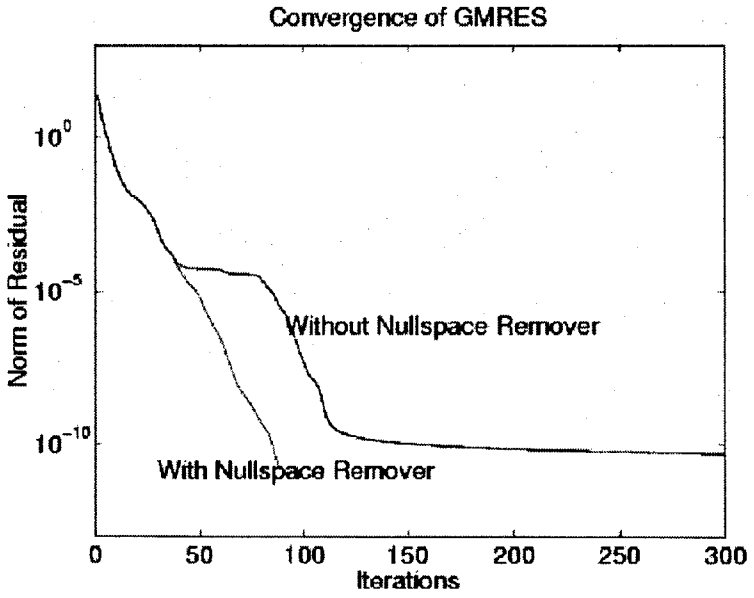

Figure 6.3: GMRES converges faster with the null-space remover.

6.4 The "Regularization" Method

Other than modifying the GMRES algorithm, modifying the G matrix can also prevent inverting a singular matrix. Once we know the null space of the G matrix, it is not difficult to reconstruct another nonsingular matrix G' so that the null-space-free solution is the unique solution. This is the "regularization" method. In this method, m rank-one-matrices are used to reconstruct the singular G matrix of an m-object problem.

$$G' = G + \sum_{i=1}^{m} c_i N_i N_i^T \tag{6.11}$$

Where $c_i N_i N_i^T$ is a rank-one-matrix, c_i is the coefficient. Then, $G' N_i \neq 0$ and G' is no longer singular. $G' F = U$ has a unique solution $F^{\perp}$:

$$G' F^{\perp} = GF^{\perp} + \sum_{i=1}^{m} c_i N_i N_i^T F^{\perp} = GF^{\perp} + 0 = U \tag{6.12}$$

It is clear from the above equation that the null-space-free solution is also the solution of the "regularization" method, it can be solved directly from the equation $G' F = U$.

The "regularization" method is also a good method, but we have chosen to use the modified GMRES algorithm in FastStokes. This is not only because the fast convergence of the modified GMRES algorithm is attractive, but also because the uncertainty of c_i in the "regularization" method raises suspicions that improper c_i might cause problems. Large c_i might overwhelm G and smaller c_i does not help very much. A well-conditioned G' depends on good methods to pick the c_i, while this problem does not exist if the modified GMRES

method is used. One idea for picking the c_i is to make the nonzero elements of the rank-one matrix comparable to the diagonal elements of the G matrix.

6.5 Pressure Pinning Method

The null-space-free solution is obviously non-physical and confusing, as indicated by Figure 6.2. If detailed surface forces are desired, the null-space component must be properly added and the pressure must be taken into account. This is the second step, which we call the pressure pinning method.

As mentioned before, a pressure boundary condition is necessary to compute a unique solution, either a P_∞ or a $P_{\text{at a certain point}}$ is needed. Assuming P_∞ is given, the pressure integral equation (6.6) may be applied to calculate pressure in the entire fluid domain. The pressure solution, as proved in the previous sections, is correct and unique. Since the only uncertainty between the null-space-free force solution and the correct one is the magnitude of the null-space vector, we can calculate the pressure on the surface and apply the relationship $f_n = -P$ to figure out this magnitude. Only the pressure at one point needs to be calculated because there is only one variable C_i for each closed surface.

$$F^{\text{right solution}} = F^{\perp} + \sum_{i=1}^{m} C_i N_i \tag{6.13}$$

N_i in the above equation is the discretized form of the surface outward normal. Equation (6.6) is applied to calculate the pressure at this point on the surface. Note that the relationship $f_n = -P$ is true on the surface of the rigid body, so let

$$C_i = \left(-P - F_n^{\perp}\right)\Big|_{\text{at a certain point on object } i} \tag{6.14}$$

and substitute it into (6.13) to calculate the final solution that satisfies both the Stokes equations and the pressure boundary condition for an m-object system.

Theoretically, C_i should be the same at any point on one closed surface, but discretization error and other numerical error may affect the accuracy of C_i. A straightforward conclusion is that the numerical solutions of C_i at points far from corners and edges are more accurate.

6.6 Numerical Results

Results from applying the two-step method to a two-plate problem are shown in Figures 6.4 and 6.5. In the two-plate example, the top plate is moving up at a velocity $V_z = 1$ while the bottom plate is fixed. The sizes of the two plates are $100\mu m \times 100\mu m \times 15\mu m$. Figure 6.4 shows the result of ignoring the singularity: the solution is badly contaminated by the null-space vectors. Figure 6.5 shows the result of using the modified GMRES and the pressure pinning method described above. Although it is not obvious from the figures, both methods give the correct total forces, but it is quite clear that only the second one gives the correct surface force distribution.

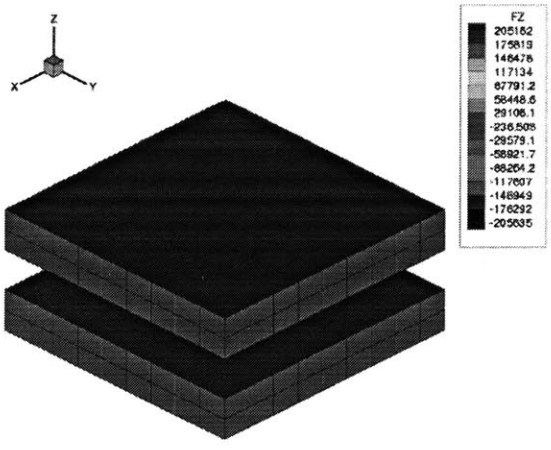

Figure 6.4: A wrong solution without considering the singular BEM operators

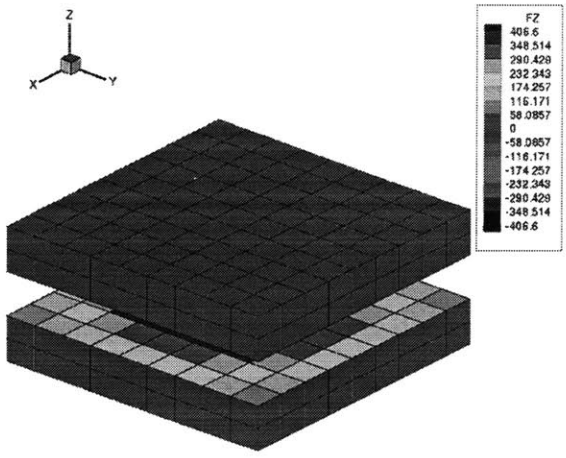

Figure 6.5: The correct solution of the two parallel plate example

Chapter 7

Numerical simulation examples using the steady incompressible FastStokes solver

The previous chapters have covered the major theoretical aspects of the incompressible FastStokes simulation program. We give four simulation examples in this chapter to show the effectiveness of the steady incompressible FastStokes solver. A sphere, a resonator, a micro-mirror, and ADXL76 accelerometer are simulated. The simulation results of real devices are further compared with testing results to show the accuracy of the steady incompressible FastStokes solver.

7.1 A translating sphere

For a simple spherical geometry, an analytical solution of the Stokes equation exists. Given the radius of the sphere R_0 and constant velocity $\vec{U}$, the drag force on the sphere is given by:

$$\vec{F} = 6\pi\mu R_0 \vec{U} \tag{7.1}$$

In this computational experiment, it is assumed that $\mu = 1, R_0 = 1, U_z = 1$, and FastStokes is used to calculate the z-direction drag forces numerically. Figure 7.1 shows the surface force distribution. The red stars in Figure 7.2 are the relative errors that decrease linearly with respect to the number of panels in a log-log plot. The blue circles are the errors of the total surface area due to the flat panel discretization. Note that the blue line is parallel, and very close, to the red line. This is because the error mainly comes from the geometrical error of using a flat panel discretization, and this geometrical error is reflected in the error of total surface area. The CPU times of using the $O(n\log(n))$ FastStokes solver and the traditional

$O(n^3)$ Gaussian elimination method (LU decomposition) are compared in Figure 7.3. If 5,000 panels are used, FastStokes is about 3,000 times faster than Gaussian elimination. The memory usage of Gaussian elimination is $O(n^2)$ while that of FastStokes is much less (about $O(n) \sim O(n^{1.5})$); the comparisons are shown in Figure 7.4. A 500-Mhz dual-processor computer running the Alpha-Linux system is used in these simulations.

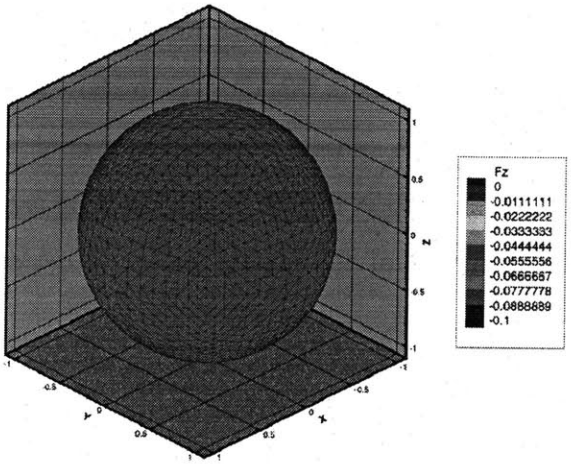

Figure 7.1: A translating sphere

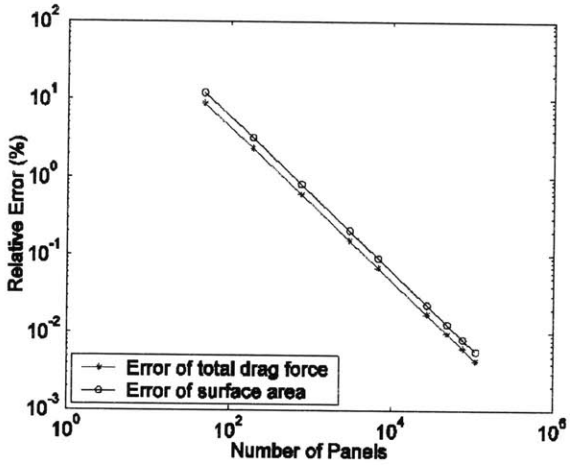

Figure 7.2: Relative error of the sphere vs. the number of panels

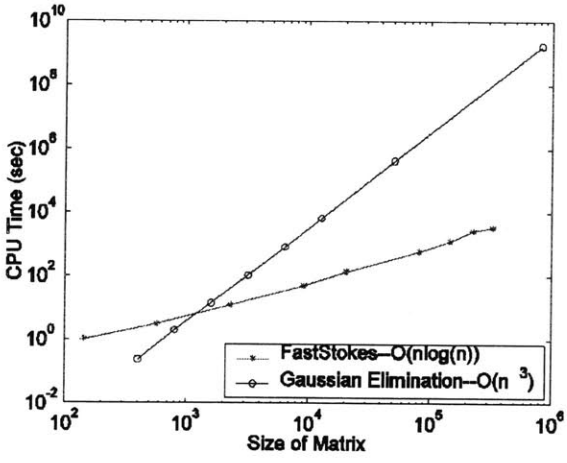

Figure 7.3: CPU times of FastStokes and Gaussian elimination

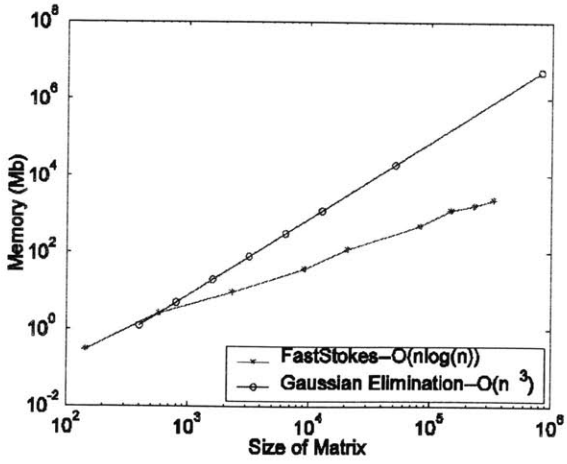

Figure 7.4: Memory usages of FastStokes and Gaussian elimination

7.2 Comb-drive resonator

A lateral comb-drive resonator is shown in Figure 7.5. The test structure was fabricated using the MUMPS process at MCNC (now Cronos Integrated Microsystems Inc., Research Triangle Park, North Carolina). Some important dimensions are given in Table 7.1. The movable comb-drive was set into motion in air at atmospheric pressure using an electrical stimulus to one static comb-drive. The magnitude and phase of the resulting motions were measured using the Computer Micro Vision technology [50]. The measured resonant frequency of the lateral motion is 19.2 kHz and the quality factor is 27.

Table 7.1: Resonator dimensions

	Dimensions (μm)
Finger gap	2.88
Finger length	39.96
Finger overlap	19.44
Tether length	151
Tether width	1.1
Thickness	1.96
Substrate gap	2

A discretization using 16,544 panels is shown in Figure 7.5. The air viscosity used in the calculation is $\mu = 1.843 \times 10^{-5}\, kg/m \cdot s$ [17]. FastStokes was used to simulate the device, and the lateral direction surface force solution is shown in Figure 7.6. Using the rigid-body assumption and a second-order spring-mass-damper system as a macro-model, we calculate the damping coefficient b from the simulation result and then further calculate the quality factor Q:

$$m_{eff}\ddot{x} + b\dot{x} + kx = F_{\text{electrostatic}}$$
$$Q = \frac{\sqrt{km_{eff}}}{b} \tag{7.2}$$

The effectiveness mass $m_{eff} = m_m + \frac{12}{35}m_t = 5.14 \times 10^{-11} kg$ is calculated using Rayleigh's method [35], where m_m and m_t are the masses of the movable comb-drive and the tethers individually calculated from geometry. Stiffness k can be calculated from the measured resonance frequency and the effective mass using $k = (2\pi f_0)^2 m_{eff} = 0.748 N/m$. The simulation result is compared with the experimental result in Table 7.2. The steady incompressible FastStokes solver gives a numerical solution that is very close to the experimental results, while simple approaches such as using the Couette flow model or 1-D Stokes flow model fail. A major reason is that those simple fluid models cannot model 3-D effects, which are important in this resonator case. This is reflected by the inaccurate inter-finger and end forces.

The solution convergence with respect to different discretizations is shown in Figure 7.7; the solution is accurate even if a coarse mesh with 4,868 quadrilateral panels is used. The CPU time is shown in Figure 7.8; a very fine discretization with 59,280 panels takes a little more than an hour's time.

Table 7.2: Comb-drive resonator simulation and measurement results

	Drag forces (nN) / (U_x=1 m/s)					Q
	Total	Bottom	Top	Inter-finger	End and others	
Couette Flow	123.7	108.9		14.8		50.1
1-D Stokes	137.1	108.9	13.5	14.8		45.2
FastStokes	223.7	123.0 **(55%)**	26.8 **(12%)**	73.8 **(33.0%)**		27.7
Experiment						27

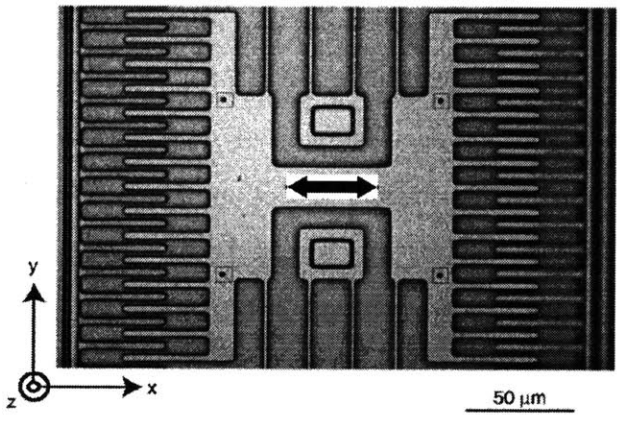

Figure 7.5: SEM of a lateral resonator

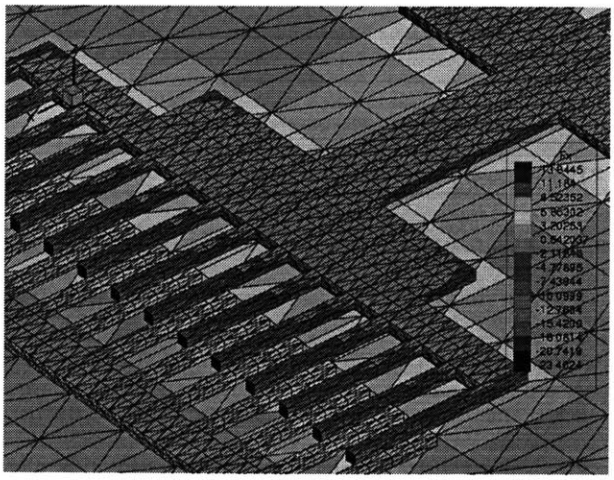

Figure 7.6: Detailed drag force on a lateral resonator using the incompressible Stokes model

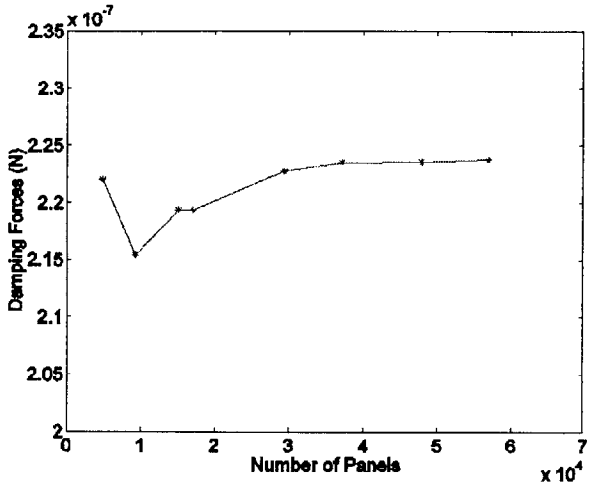

Figure 7.7: Convergence of the drag forces of the comb-drive resonator simulation

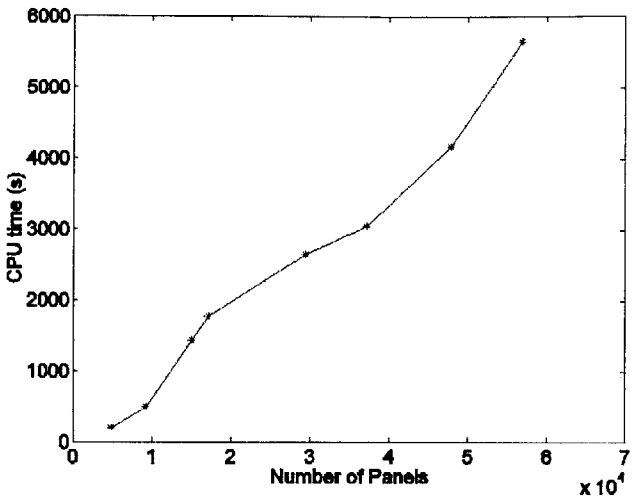

Figure 7.8: CPU times of the comb-drive resonator simulation

7.3 Micro-mirror

An electrostatically actuated micro-mirror was simulated using FastStokes [46]. The micro-mirror was fabricated and tested in the Micromachined Product Division of Analog Devices Inc. (Cambridge, MA). The air-packaged micro-mirror is the critical part of an optical switch, and its dynamic performance is strongly affected by the viscous drag forces. Testing data have shown that the mirror is heavily damped, with a quality factor around 2 for certain designs. Two major modes, the "mirror only" rotation mode and "mirror + gimbal" rotation mode, are simulated here. Table 7.3 compares the simulation results and experimental results of two different designs:

Table 7.3: Quality factors of the micro-mirror simulations and measurements

		Measured Q	Simulated Q	Error (%)
Mirror 1	Mirror+gimbal	2.31	2.36	2.16
	Mirror	3.45	3.14	8.99
Mirror 2	Mirror+gimbal	4.27	4.69	9.84
	Mirror	10.63	10.16	4.42

The simulated and measured quality factors match within 10%. Again, the small differences prove the accuracy of the FastStokes program. Figure 7.9 shows the Z-direction surface force on the mirror when both mirror and gimbal rotate. Only half of the mirror is plotted in Figure 7.9 in order to give a clear view of the force distribution. When the mirror rotates counterclockwise, as we expected, the viscous drag force on the mirror surface is against this motion. Figure 7.10 shows that the simulation solution quickly converges as the discretization is refined. Figure 7.11 shows the CPU time. The simulation was finished in less than an hour when 42,340 panels were used.

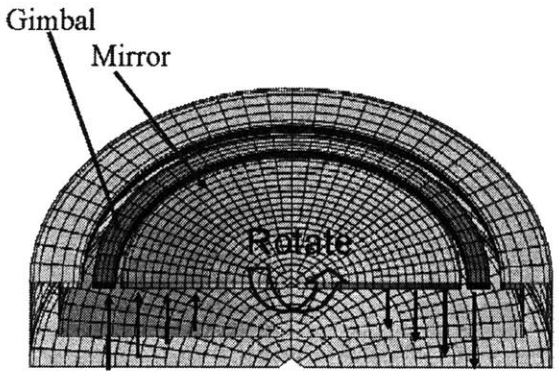

Figure 7.9: Z-direction force on a micro-mirror

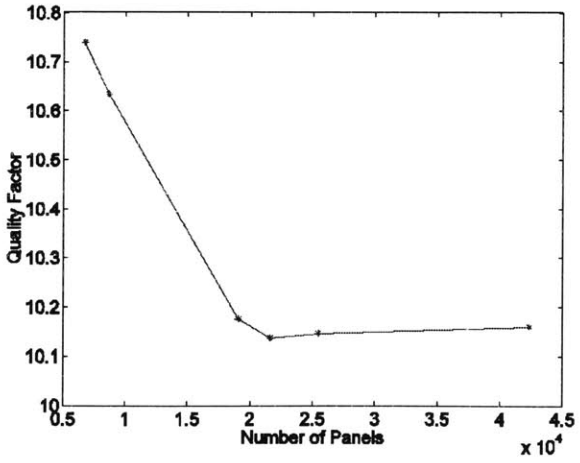

Figure 7.10: Convergence of the micro-mirror simulation

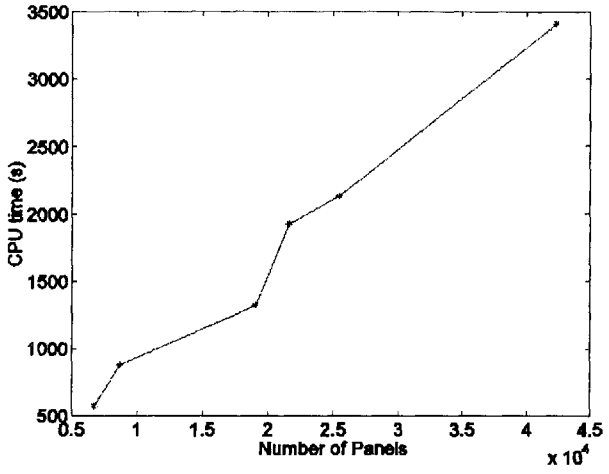

Figure 7.11: CPU times of the micro-mirror simulation

7.4 ADXL76

A picture of the ADXL76 accelerometer is shown in Figure 7.12. The device is fabricated by the Micromachined Product Division of Analog Devices using the *i*MEMS process. Some basic dimensions are listed in the following table.

Table 7.4: Key dimensions of ADXL 76

Finger overlap	104 μm
Air gap between fingers	1.3 μm
Air gap between substrate	1.6 μm
Number of cells	28

It is very clear that the aspect ratio of air gaps between the comb fingers is large enough to generate strong viscous damping forces. This is indeed the dominant source of viscous damping forces in ADXL76. The substrate is also very close to the finger and proof mass, hence the shear

damping forces between the substrate and beam (movable comb) should not be neglected. In such cases, a simple model based on a semi-analytical approach will not yield very accurate results. However, full 3-D simulation of the entire device is very difficult even for a fast solver like FastStokes. Those very close fingers cannot be accurately simulated without using a very fine discretization, and this generates large numbers of unknowns. To solve this problem efficiently and to figure out a good method for modeling even more complicated accelerometers, cells instead of the whole devices are simulated (see Figure 7.13). Simulation results show that the damping forces increase linearly with the number of cells (Figure 7.14). An extrapolation yields a quality factor of 6.46, which is very close to the tested quality factor of the devices (about 6.0).

Using the FastStokes solver, we also simulated the Q-factor drifts of ADXL76 due to geometry variations such as beam curvatures and positional offsets. Testing a single device or a batch of wafers does not yield many useful results, since curvatures and offsets are usually coupled, and they vary from device to device. But the simulation results clearly indicate the percentage of damping changes due to geometry variations. Therefore, a fast full 3-D simulation program, such as FastStokes, plays a very important role in studying the damping effect of ADXL76.

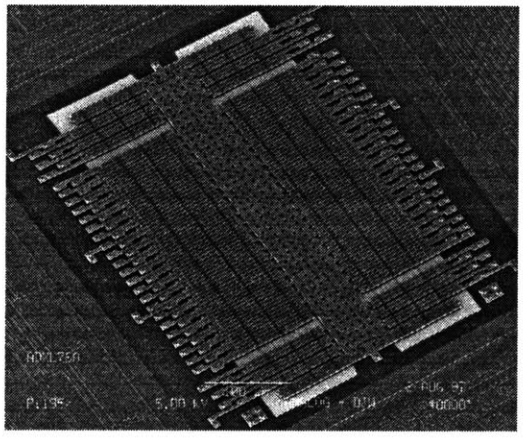

Figure 7.12: ADXL76 accelerometer

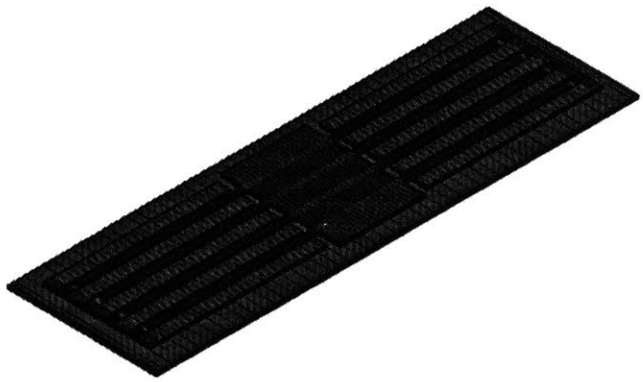

Figure 7.13: 4 cells used in ADXL76 simulation

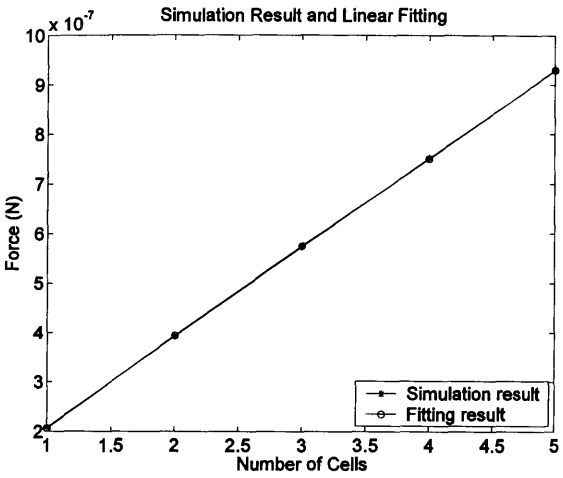

Figure 7.14: Drag forces on cells and linear data fitting

Chapter 8

Compressible Stokes Flow

In the past twenty years, MEMS has been one of the most exciting fields. Many novel design ideas have revealed the great potential of the micro-scale devices, and this has attracted ever-increasing attention to exploring new phenomena. Squeeze film damping has been studied for decades in the lubrication industry. The discovery of strong squeeze film damping in micromachined devices has renewed the interest in further study of the low Reynolds number flow. Unlike the oil lubrication problems people studied before, the squeeze film effects in MEMS are usually related to weak air compression.

Using the thin-film surface-micromachining technologies, it is easy to fabricate complicated 2-D surface structures. Hence many MEMS devices have a large thin proof mass separated by a narrow gap from the substrate. When the device is packaged in air and the out-of-plane motion of the proof mass is important, air damping must be taken into consideration. Because of the large aspect ratio of the thin air film trapped between the proof mass and the substrate, the squeeze film effect may be very strong. In such cases, viscous drag force along the narrow passage obstructs the motion of air, and this causes the generation of a high (or low) pressure region between the proof mass and the substrate. The damping force is very strong due to the viscous force. In addition, there is also a strong spring force because air is compressed (or expanded).

Many researchers have studied the squeeze film effect; experimental results show significant air compression when the squeeze film effect is strong. This is particularly reflected by the large spring forces generated by the air compression. A large damping force usually leads to a small quality factor; but a large spring force may change the resonance frequency by orders of

magnitude. If the device is designed as a resonator or a sensor, the compression effect may change the dynamic performance of the device dramatically. Therefore, compression effect must be taken into consideration.

A well-developed method for handling the squeeze film problem is solving the Reynolds equation [10]. The Reynolds equation is derived from the compressible Navier-Stokes equation, based on special simplifications applicable to the thin fluid film between two close surfaces. Those assumptions, though accurate when applied to thin liquid films, are not applicable to air in complicated general 3-D geometries.

The linearized compressible Reynolds equation is a simplified version of the nonlinear compressible Reynolds equation. It is computationally friendly; there are even analytical solutions for simple cases such as thin films between two square or circular plates [4]. The linearized compressible Reynolds equation has been applied extensively in studying the air squeeze film effect in MEMS devices. Researchers have already achieved exciting simulation results that prove the importance of modeling the air compression effect.

Unfortunately, comparisons between experimental results, numerical simulation results using FEM, and numerical solutions using the linearized compressible Reynolds equation also show the shortcomings of this approach [12, 40]. The Reynolds equation was developed for thin fluid films; it is not capable of handling the increasingly complex geometries of the new MEMS designs. Some modifications can improve the accuracy of the linearized Reynolds equation; they can even model the edge effect that may cause large errors if neglected. But such modifications are usually based on simple analytical approaches whose applications are also limited. Obviously, it is difficult to extend the Reynolds equation approach to general devices, such as the micro mirror discussed in the previous chapter. A computationally friendly 3-D fluid model is needed if accurate solutions are desired.

We have successfully developed a 3-D compressible Stokes flow model and a corresponding PFFT-accelerated BEM solution scheme [45]. This 3-D fluid model is derived from 3-D

Navier-Stokes model based on the low pressure variation assumption only; therefore it is applicable to general 3-D geometry. In this chapter we begin by deriving the PDEs of the compressible Stokes equation, followed by the derivations of boundary integral equations.

In addition to presenting the new fluid model, we further explore the differences between the incompressible Stokes flow model and the compressible Stokes flow model in this chapter. Numerical results and further discussion of calibrating fluid compression effects in MEMS devices are given in Chapter 9.

8.1 The compressible Stokes equation

Assuming an isothermal condition, the governing equations of the isothermal compressible flow are

$$\begin{cases} \rho\dfrac{\partial \vec{u}}{\partial t} + \rho\vec{u}\bullet(\nabla\vec{u}) = -\nabla P + \mu\nabla^2\vec{u} + \left(\kappa + \dfrac{1}{3}\mu\right)\nabla(\nabla\bullet\vec{u}) \\ \dfrac{\partial \rho}{\partial t} + \nabla\bullet(\rho\vec{u}) = 0 \\ P/\rho = RT = \text{constant} \end{cases} \tag{8.1}$$

where the first equation is the momentum equation (Navier-Stokes equation), and the second equation is the continuity equation. We also add the ideal gas equation to relate pressure P with density ρ. Again, μ is the viscosity, u is the velocity, T is the temperature, and $R \sim 287\ J/(kg\cdot K)$ is a constant. The third term on the right-hand side of the momentum equation is the so-called second viscosity term, the bulk coefficient of viscosity $\kappa = 0$ for monatomic gases. As mentioned before, the fluid motions in MEMS devices are usually generated by the oscillations of the moving parts. Given a feature size L and an oscillation frequency $\dfrac{\omega}{2\pi}$, the velocity is on the order of $L\omega$. If we introduce non-dimensional variables

$\vec{u}' = \vec{u}/U, \vec{x}' = \vec{x}/L$ and $P' = PL/\mu U$, the momentum equation can be non-dimensionalized and the result is:

$$\frac{L^2\omega}{\nu}\frac{\partial \vec{u}'}{\partial t} + \frac{UL}{\nu}\vec{u}'\bullet(\nabla'\vec{u}') = -\nabla' P' + \nabla'^2\vec{u}' + \frac{1}{3}\nabla'(\nabla'\bullet\vec{u}') \tag{8.2}$$

Because of the small feature size of the MEMS devices, the Reynolds number $\mathrm{Re} = \frac{UL}{\nu} \sim \frac{L^2\omega}{\nu}$ is small in a wide frequency range. It is also clearly shown in (8.2) that the first term and the second term of the momentum equation are of the same order of magnitude if $U \sim L\omega$ is true.

The incompressible Stokes equation is derived from the incompressible Navier-Stokes equation with the small Reynolds number assumption. By following the same rule, we neglect the nonlinear convective term to obtain the compressible Stokes equation. In addition, the density in the first and second equations is replaced with $\rho = \frac{P}{RT}$:

$$\begin{cases} \frac{P}{RT}\frac{\partial \vec{u}}{\partial t} = -\nabla P + \mu\nabla^2\vec{u} + \frac{1}{3}\mu\nabla(\nabla\bullet\vec{u}) \\ \frac{\partial P}{\partial t} + \nabla\bullet(P\vec{u}) = 0 \end{cases} \tag{8.3}$$

We refer to the above equations as the governing equations of the compressible Stokes flow. They are capable of modeling the slow motions of compressible fluids such as air or lubrication oil with tiny bubbles inside. The first term in the above equation may be neglected, but let us keep it for a moment.

Equation (8.3) is a nonlinear equation that is not easy to solve using BEM. What we prefer is an easier linear model that allows the fast simulations of geometrically complicated devices. Fortunately, an accurate linearization of equation (8.3) is possible.

It has been noticed that the pressure variations in the MEMS devices are usually small, although the pressure gradients are large. Many MEMS devices are fabricated with thin-film surface micromachining technologies; this makes generating large pressure variations almost impossible. Even if the device is not very thin, a large pressure variation on the order of 1 atm does not happen frequently unless lots of energy is consumed. Therefore, we linearize the pressure in those nonlinear terms around the ambient air pressure:

$$\begin{cases} \rho_0 \dfrac{\partial \vec{u}}{\partial t} = -\nabla P + \mu \nabla^2 \vec{u} + \dfrac{1}{3}\mu \nabla(\nabla \bullet \vec{u}) \\ \dfrac{\partial P}{\partial t} + P_0 \nabla \bullet \vec{u} = 0 \end{cases} \tag{8.4}$$

where P_0 is the ambient air pressure, and $\rho_0 = \dfrac{P_0}{RT}$. Equations in (8.4) are the linearized compressible Stokes equations. By "Stokes equations" we mean both the momentum equation and the continuity equation. If the first term in the momentum equation is also neglected, the following equations are named the simplified linearized compressible Stokes equations.

$$\begin{cases} 0 = -\nabla P + \mu \nabla^2 \vec{u} + \dfrac{1}{3}\mu \nabla(\nabla \bullet \vec{u}) \\ \dfrac{\partial P}{\partial t} + P_0 \nabla \bullet \vec{u} = 0 \end{cases} \tag{8.5}$$

The pressure time derivative term in the continuity equation must be kept to allow fluid compression, so both (8.4) and (8.5) are capable of modeling weak fluid compression effects generated by oscillating motions.

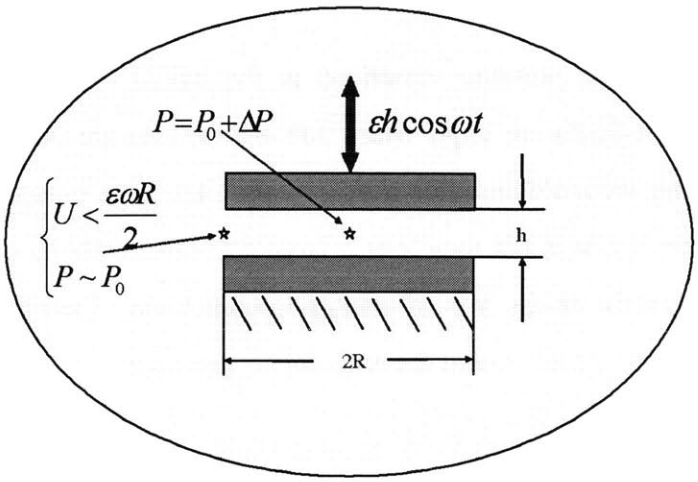

Figure 8.1: Two circular parallel plates

The linearization of the velocity time derivative term in the momentum equation will not cause much error, since this term itself is negligible. But the effect of linearizing the continuity equation is not very straightforwardly understandable without using an example. Consider two circular plates with radius R and an air gap h in between. The bottom plate is fixed and the top plate oscillates up and down with a sinusoidal displacement $\varepsilon h\sin\omega t$ (see Figure 8.1), where ε is a constant number much smaller than 0.1. If we temporarily assume the air is incompressible and apply the continuity equation for incompressible flow, the average amplitude of radial air velocity around the edge of the gap is $U=\frac{\varepsilon\omega R}{2}$. Thus, $U<\frac{\varepsilon\omega R}{2}$ because the fluid is compressible. We further assume the pressure at the edge of the gap is zero. This is not accurate, but it gives an approximation $\frac{\partial P}{\partial x}\sim\frac{\Delta P}{R}$, where ΔP is the largest pressure variation that is at the center of the circular air gap. Next, we expand the nonlinear continuity equation of the compressible Stokes flow:

$$\frac{\partial P}{\partial t}+\nabla\bullet\left(P\vec{u}\right)=\frac{\partial P}{\partial t}+(\nabla P)\bullet\vec{u}+P(\nabla\bullet\vec{u}) \tag{8.6}$$

We assume $P = P_0 + \widetilde{P}$, and apply the two conclusions we obtained from the two-circular-plate example to the following scale analysis

$$\begin{aligned}
&\frac{\partial P}{\partial t} + \nabla P \bullet \vec{u} + P(\nabla \bullet \vec{u}) \\
&\sim \omega \widetilde{P} + \frac{\partial P}{\partial x} U + \left(P_0 + \widetilde{P}\right)(\nabla \bullet \vec{u}) \\
&\sim \omega \widetilde{P} + \frac{\Delta P}{R} \frac{\varepsilon \omega R}{2} + P_0 (\nabla \bullet \vec{u}) \\
&= \omega \widetilde{P} + \frac{\varepsilon \omega \Delta P}{2} + P_0 (\nabla \bullet \vec{u})
\end{aligned} \tag{8.7}$$

First, the third term on the right-hand side of (8.7) is linearized based on the assumption $\widetilde{P} << P_0$. If $\varepsilon \le 0.1$, and therefore the amplitude of the plate motion is much smaller than the air gap, the second term on the right-hand side of (8.7) is much smaller than the first term and it may be neglected. Generally speaking, neglecting the second term is accurate when $\frac{\partial P}{\partial t} >> \nabla P \bullet \vec{u} \Rightarrow \frac{U}{L} << \omega \Rightarrow \frac{U}{\omega L} << 1$, where U is the maximum velocity, and L is the length over which the most significant pressure drop happens. We have $\frac{U}{\omega L} \approx \frac{\varepsilon}{2}$ in the two plate example. $\frac{U}{\omega L}$ is, therefore, equivalent to the nondimensional oscillation amplitude, and it can also be used as a nondimensional number to justify the validity of the above linearization.

From the above analysis, we conclude that the linearization approach is accurate if the oscillation amplitude is small ($\varepsilon \le 0.1$) and the pressure variation is small ($\widetilde{P} << P_0$). Then the solution of the linearized compressible Stokes equations should be very close to the solution of the nonlinear compressible Stokes equations.

Equations (8.4) and (8.5) are linear equations that allow a PFFT-accelerated BEM approach. A major part of this chapter focuses on the derivations of the corresponding boundary integral equations. Before we start to discuss BEM, the next section compares equation (8.5) with the linearized compressible Reynolds equation.

8.2 The compressible Reynolds equation

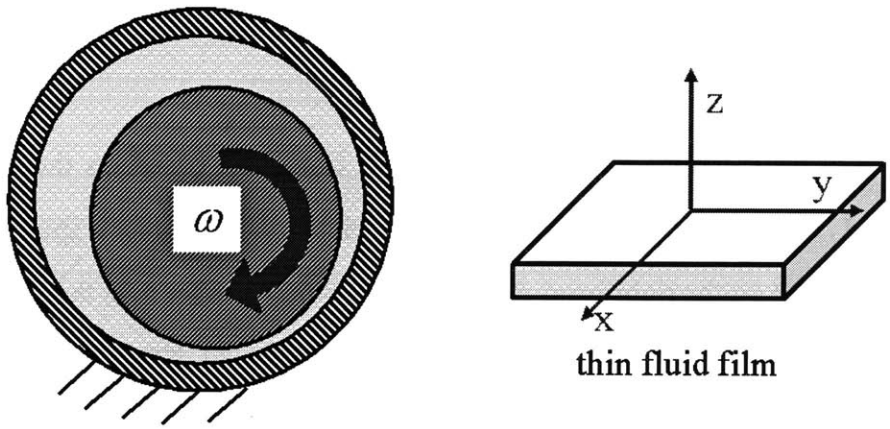

Figure 8.2: A fluid lubrication problem

The compressible Reynolds equation is derived from the isothermal compressible Navier-Stokes equations with key assumptions that are applicable to thin fluid films. The basic equations used in deriving the compressible Reynolds equation are

$$\begin{cases} \dfrac{\partial P}{\partial x} = \mu \dfrac{\partial^2 u_x}{\partial z^2}; \quad \dfrac{\partial P}{\partial y} = \mu \dfrac{\partial^2 u_y}{\partial z^2} \\ \dfrac{\partial \rho}{\partial t} + \nabla \bullet (\rho \vec{u}) = 0 \\ P/\rho = RT = \text{constant} \end{cases} \tag{8.8}$$

The fluid is assumed to be a laminar Newtonian flow with constant viscosity. In addition, the above equations also assume:

a. The Reynolds number is small. Hence, the acceleration term and convective term in the Navier-Stokes equation can be neglected.

b .The second viscosity term is negligibly small.

c. $u_z = 0,\ \dfrac{\partial P}{\partial z} = 0.$

Next, the first two equations in (8.8) are integrated to give approximate velocity profiles and an integration of the continuity equation across the fluid film yields the compressible Reynolds equation for thin film with thickness h. Note that we have also assumed the top surface of the thin film is moving up and down and the bottom surface is fixed.

$$\nabla \bullet \left(h^3 \nabla P\right) = 12\mu \frac{\partial(\rho h)}{\partial t} \tag{8.9}$$

The above Reynolds equation can be further linearized to the linearized compressible Reynolds equation with oscillation frequency ω. The non-dimensional linearized compressible Reynolds equation is:

$$\nabla^2 \Theta - \sigma \frac{\partial \Theta}{\partial \tau} = \sigma \frac{\partial \theta}{\partial \tau}$$
$$\text{with} \quad \tau = \omega t; \theta = \frac{\delta}{h}\cos \omega t; \Theta = \frac{\Delta P}{P_0}; \sigma = \frac{12\mu L^2 \omega}{P_0 h_0^2} \tag{8.10}$$

where σ is the squeeze number. If applied to the air film between two parallel square plates, L is the length of the plate. The linearized compressible Reynolds equation is a computationally friendly model. Analytical solutions have already been solved for thin films between parallel rectangular and circular plates [4]. Because of its simplicity, the linearized compressible Reynolds equation has been used extensively in modeling MEMS air squeeze

film effects. However, more and more researchers have started to question the accuracy of such an over-simplified approach. A significant shortcoming of this model is the neglect of the edge effect, since the equation is only applicable to fluid films. It has been found that the linearized Reynolds equation may be very inaccurate when the air film is not "thin" enough or the aspect ratio is small, where the aspect ratio of the thin film is defined as L/h. If the aspect ratio of the air film is between 5~20, a 20~75% or even larger error in damping force calculation should be expected [40].

The inaccuracy of the linearized Reynolds equation is mostly due to neglecting 3-D effects. There are some modifications that help reduce errors, but such modifications are usually based on analytical formulas whose applications are also limited. A full 3-D fluid model like the linearized compressible Stokes equation is obviously a superior one.

We only give a brief derivation of the linearized compressible Reynolds equation in this section so as to avoid listing too many details of this well-studied model. Because the derivation procedure is based on thin-film applications and many specific assumptions have been applied, it is very difficult to compare the performance of the linearized compressible Reynolds equation with that of the linearized compressible Stokes equation simply by examining the derivation procedures. We compare only the major assumptions in Table 8.1. Again, the linearized compressible Reynolds equation is only applicable to thin fluid films; but the linearized compressible Stokes equation is a true 3-D fluid model applicable to the entire fluid domain. The last assumption in Table 8.1 is clearly not good for general 3-D geometry; this strongly limits the application of the linearized compressible Reynolds equation. Comparisons of numerical simulation results will be given in the next chapter.

Table 8.1: The assumptions used by the two linearized fluid models

Linearized Compressible Reynolds Equation	Linearized Compressible Stokes Equation
Low Reynolds number	
Isothermal	
Small motion amplitude	
Small pressure variation	
$\rho\frac{\partial\vec{u}}{\partial t}=0$	
$u_z=0,\frac{\partial P}{\partial z}=0$	

8.3 The fundamental solutions of the compressible Stokes equation

Solving the fundamental solutions or the Green's functions is the first step in deriving boundary integral equations. The fundamental solutions represent responses of the fluid due to a point force perturbation at the source point. If we assume there is an oscillating point force $\vec{g}e^{i\omega t}\delta(\vec{x}\text{-}\vec{x}_0)$ at point $\vec{x}_0$, where $\omega=2\pi f$ is the angular velocity of the oscillation, $i=\sqrt{-1}$, and $\vec{g}$ is the strength of the point force, then velocity, pressure, and stress responses are also sinusoidal functions that can be expressed as $u_k e^{i\omega t}, Pe^{i\omega t}$, and $\sigma_{lk}e^{i\omega t}$, and the solutions of the corresponding velocity, pressure, and stress fields are:

$$
\begin{aligned}
u_k(\vec{x}) &= \frac{1}{8\pi\mu} G_{kj}(\vec{x},\vec{x}_0) g_j \\
P(\vec{x}) &= \frac{1}{8\pi} p_j(\vec{x},\vec{x}_0) g_j \\
\sigma_{lk}(\vec{x}) &= \frac{1}{8\pi} T_{ljk}(\vec{x},\vec{x}_0) g_j
\end{aligned}
\tag{8.11}
$$

Note that u_k, P, and σ_{lk} are complex numbers because of the time derivative terms in the linearized compressible Stokes equation. $G_{kj}(\vec{x},\vec{x}_0)$, $p_j(\vec{x},\vec{x}_0)$, and $T_{ljk}(\vec{x},\vec{x}_0)$ are the fundamental solutions (Green's functions, or BEM kernels). The stress of the compressible Navier-Stokes flow is defined as [20]:

$$
\sigma_{lk} = \left(-P + (\kappa - \frac{2}{3}\mu)\nabla_j u_j \right)\delta_{lk} + \mu\left(\frac{\partial u_l}{\partial x_k} + \frac{\partial u_k}{\partial x_l} \right) \tag{8.12}
$$

where $\kappa = 0$. Substituting $u_k e^{i\omega t}, Pe^{i\omega t}$, and $\sigma_{lk} e^{i\omega t}$ into equation (8.4) and adding the point force term $\vec{g}e^{i\omega t}\delta(\vec{x}\text{-}\vec{x}_0)$ yields the system equations with the amplitudes as unknowns:

$$
\begin{cases}
\rho_0 i\omega u_j = -\nabla_j P + \mu\nabla^2 u_j + \frac{1}{3}\mu\nabla_j(\nabla_k u_k) + g_j\delta(\vec{x}\text{-}\vec{x}_0) = \nabla_k \sigma_{jk} + g_j\delta(\vec{x}\text{-}\vec{x}_0) \\
i\omega P + P_0\nabla_j u_j = 0
\end{cases}
\tag{8.13}
$$

The unknowns are the Green's functions we wish to solve. As mentioned before, solving the above equation is relatively easy if we do it in the Fourier domain and then transfer the solutions back to the 3-D space. The 3-D Fourier transformations of velocity, pressure, and two frequently used functions are

$$
\begin{aligned}
u_k &= \frac{1}{(2\pi)^{\frac{3}{2}}}\int_{-\infty}^{+\infty}\tilde{u}_k e^{i\vec{\alpha}\bullet\vec{x}}d\vec{\alpha} \\
P &= \frac{1}{(2\pi)^{\frac{3}{2}}}\int_{-\infty}^{+\infty}\tilde{P} e^{i\vec{\alpha}\bullet\vec{x}}d\vec{\alpha} \\
\delta(\vec{x}-\vec{y}) &= \frac{1}{(2\pi)^{3}}\int_{-\infty}^{+\infty} e^{i\vec{\alpha}\bullet(\vec{x}-\vec{y})}d\vec{\alpha} \\
\frac{1}{4\pi|\vec{x}-\vec{y}|} &= \frac{1}{(2\pi)^{3}}\int_{-\infty}^{+\infty}\frac{e^{i\vec{\alpha}\bullet(\vec{x}-\vec{y})}}{a^2}d\vec{\alpha}
\end{aligned}
\tag{8.14}
$$

where $\vec{\alpha} = \alpha_1 i + \alpha_2 j + \alpha_3 k$ is the frequency parameter, and $a^2 = \alpha_j a_j$. Substituting (8.14) into (8.13) gives the corresponding governing equations in the frequency domain,

$$
\begin{cases}
(\rho_0 i\omega + \mu\alpha^2)\tilde{u}_1 + i\alpha_1\tilde{P} + \dfrac{1}{3}\mu\alpha_1(a_1\tilde{u}_1 + a_2\tilde{u}_2 + a_3\tilde{u}_3) = \dfrac{1}{(2\pi)^{\frac{3}{2}}}g_1 \\
(\rho_0 i\omega + \mu\alpha^2)\tilde{u}_1 + i\alpha_2\tilde{P} + \dfrac{1}{3}\mu\alpha_2(a_1\tilde{u}_1 + a_2\tilde{u}_2 + a_3\tilde{u}_3) = \dfrac{1}{(2\pi)^{\frac{3}{2}}}g_2 \\
(\rho_0 i\omega + \mu\alpha^2)\tilde{u}_3 + i\alpha_3\tilde{P} + \dfrac{1}{3}\mu\alpha_3(a_1\tilde{u}_1 + a_2\tilde{u}_2 + a_3\tilde{u}_3) = \dfrac{1}{(2\pi)^{\frac{3}{2}}}g_3 \\
\dfrac{\omega}{P_0}\tilde{P} + (a_1\tilde{u}_1 + a_2\tilde{u}_2 + a_3\tilde{u}_3) = 0
\end{cases}
\tag{8.15}
$$

Let us define:

$$
b = \rho_0\omega, \quad \lambda_1^2 = -\frac{\dfrac{\omega b}{P_0}}{1+\dfrac{4\omega\mu i}{3P_0}}, \quad \lambda_2^2 = \frac{bi}{\mu}, \quad C_1 = \frac{1+\dfrac{\omega\mu i}{3P_0}}{\left(1+\dfrac{4\omega\mu i}{3P_0}\right)}
\tag{8.16}
$$

and solve the velocity and pressure fundamental solutions in the frequency domain:

$$\tilde{u}_k = \frac{1}{(2\pi)^{\frac{3}{2}}}\left(\frac{\delta_{kj}}{\mu(\alpha^2+\lambda_2^2)} - C_1\frac{\alpha_k\alpha_j}{\mu(\alpha^2+\lambda_1^2)(\alpha^2+\lambda_2^2)}\right)g_j$$
$$\tilde{P} = -\frac{1}{(2\pi)^{\frac{3}{2}}}\frac{1}{\left(1+\frac{4\omega\mu i}{3P_0}\right)}\frac{\alpha_j i}{(\alpha^2+\lambda_1^2)}g_j \tag{8.17}$$

After taking an inverse Fourier transformation of (8.17), we obtain the fundamental solutions in the 3-D space:

$$u_k(\vec{x}) = \frac{1}{8\pi\mu}G_{kj}(\vec{x},\vec{x}_0)g_j, \text{and } P(\vec{x}) = \frac{1}{8\pi}p_j(\vec{x},\vec{x}_0)g_j$$

with

$$p_j = \frac{2\hat{x}_j}{\left(1+\frac{4\omega\mu i}{3P_0}\right)}\left(\frac{1}{R^3}+\frac{\lambda_1}{R^2}\right)e^{-\lambda_1 R}$$
$$G_{kj} = 2\delta_{kj}\left[\frac{e^{-\lambda_2 R}}{R} - C_0\left(\frac{\left(e^{-\lambda_1 R}-e^{-\lambda_2 R}\right)}{R^3}+\frac{\left(\lambda_1 e^{-\lambda_1 R}-\lambda_2 e^{-\lambda_2 R}\right)}{R^2}\right)\right]+$$
$$+2C_0\hat{x}_k\hat{x}_j\left[3\frac{\left(e^{-\lambda_1 R}-e^{-\lambda_2 R}\right)}{R^5}+3\frac{\left(\lambda_1 e^{-\lambda_1 R}-\lambda_2 e^{-\lambda_2 R}\right)}{R^4}+\frac{\left(\lambda_1^2 e^{-\lambda_1 R}-\lambda_2^2 e^{-\lambda_2 R}\right)}{R^3}\right]$$
$$C_0 = \frac{1+\frac{\omega\mu i}{3P_0}}{\left(1+\frac{4\omega\mu i}{3P_0}\right)\left(\lambda_2^2-\lambda_1^2\right)}, R = |\vec{x}_0-\vec{x}|, \hat{x}_j = x_{0j}-x_j \tag{8.18}$$

The above fundamental solutions have two frequency parameters λ_1 and λ_2 in the exponential part, and one frequency coefficient C_0. The kernels are fast-decaying oscillating kernels; they are obviously very complicated.

8.4 Simplifying the fundamental solutions of the compressible Stokes equations

Integrating complicated kernels is a difficult job, especially when both good accuracy and high speed are desired. The kernels in (8.18) are much more complicated than these frequently used $\frac{1}{r}$ and $\frac{e^{ikr}}{r}$ kernels, particularly the velocity kernel. Without talking about integrating, even calculating the value of the velocity kernel is difficult. Very often, part of the kernel, which is a large number, cancels with large numbers from other parts, and the cancellation leaves only a very small number badly contaminated by numerical errors. We discuss simplifying the linearized compressible Stokes kernels in this section.

The linearized compressible Stokes model is a more general fluid model that takes both the compression effect and the transient effect into account. From another point of view, the simplified linearized compressible Stokes flow model, the unsteady incompressible Stokes flow model, and the steady incompressible Stokes flow model are just simplified versions of the linearized compressible Stokes flow model. This section shows how to derive these simple kernels directly by simplifying the kernels of the linearized compressible Stokes equations.

8.4.1 The kernels of the simplified linearized compressible Stokes equations

The simplified linearized compressible Stokes kernels are presented in this subsection. To compare the differences, we use the results in the previous section. Because ρ_0 only appears in the acceleration term of the momentum equation in (8.4), (8.4) becomes (8.5) if we mathematically assume $\rho_0 = 0$. Consequently, the two frequency parameters of the linearized compressible Stokes kernels are all zero:

$$b = \rho_0 \omega = 0, \quad \lambda_1^2 = -\frac{\frac{\omega b}{P_0}}{1 + \frac{4\omega\mu i}{3P_0}} = 0, \quad \lambda_2^2 = \frac{bi}{\mu} = 0 \tag{8.19}$$

Next, we substitute (8.19) into equation (8.17) and take an inverse Fourier transform to yield the fundamental solutions of the simplified linearized compressible Stokes equations:

$$G_{ij} = \frac{\delta_{ij}}{R} + C_1 \frac{\hat{x}_i \hat{x}_j}{R^3}, \quad p_j = \frac{2}{\left(1 + \frac{4\omega\mu i}{3P_0}\right)} \frac{\hat{x}_j}{R^3}$$
$$\text{where} \quad C_1 = \frac{1 + \frac{\omega\mu i}{3P_0}}{1 + \frac{4\omega\mu i}{3P_0}} \tag{8.20}$$

A comparison between (8.20) and (8.18) reveals very dramatic simplifications; the kernels of the simplified linearized compressible Stokes equation are very similar to those of the steady incompressible Stokes equation, which are simple and easy to integrate. Clearly, the kernels are still frequency-dependent, but they are no longer fast-decaying oscillating kernels because all those exponential terms are gone.

8.4.2 The kernels of the unsteady incompressible Stokes equations

Once we remove the fluid compression effect from the linearized compressible Stokes equations (8.4), it becomes the unsteady incompressible Stokes equation (4.1). A mathematical simplification that makes the compressible flow (8.4) "incompressible" is to assume $P_0 \to \infty$ so that the pressure variation becomes trivial when compared with ambient air pressure. Meanwhile, the density is kept unchanged. Accordingly, we have $\lambda_1 = 0$, and

$C_0 = \frac{1}{\lambda_2^2}$. Defining $\lambda = \lambda_2 = \sqrt{i\omega/\nu}$, equation (8.18) is simplified to the fundamental solutions of the unsteady incompressible Stokes equations:

$$u_k(\vec{x}) = \frac{1}{8\pi\mu} G_{kj}(\vec{x},\vec{x}_0) g_j, \quad \text{and } P(\vec{x}) = \frac{1}{8\pi} p_j(\vec{x},\vec{x}_0) g_j$$

where

$$p_j = \frac{2\hat{x}_j}{R^3}, \quad R = |\vec{x}_0 - \vec{x}|, \quad \hat{x}_j = x_{0j} - x_j \tag{8.21}$$

and $$G_{kj} = 2\frac{\delta_{kj}}{R}\left[e^{-\lambda R} - \frac{\left(1-e^{-\lambda R}\right)}{\lambda^2 R^2} + \frac{e^{-\lambda R}}{\lambda R}\right] + 2\frac{\hat{x}_k \hat{x}_j}{R^3}\left[3\frac{\left(1-e^{-\lambda R}\right)}{\lambda^2 R^2} - 3\frac{e^{-\lambda R}}{\lambda R} - e^{-\lambda R}\right]$$

8.4.3 The kernels of the steady incompressible Stokes equations

The steady incompressible Stokes equation only models quasi-static phenomena. Therefore, the kernels of the simplified linearized compressible Stokes equation can be simplified to the kernels of the steady incompressible Stokes equation by substituting $\omega = 0$ into equation (8.20), and this leads to

$$G_{ij} = \frac{\delta_{ij}}{R} + \frac{\hat{x}_i \hat{x}_j}{R^3}, \quad p_j = 2\frac{\hat{x}_j}{R^3} \tag{8.22}$$

8.5 The Lorentz reciprocal identity

The boundary integral equation can be derived from the Lorentz reciprocal identity, which is also called the Green's second identity applied to the Stokes flow problem. Before using it, we first derive the Lorentz reciprocal identity and prove that it can be applied to the linearized compressible Stokes flow.

The reciprocal identity is a powerful tool for understanding the solutions of the Stokes flow. Its format is simple and suitable for generating boundary integral equations. The most

important aspect is that it offers information about an unknown flow based on solutions of a known flow. We assume there are two flows bounded by the same boundary, flow A and flow B. To be more precise, A and B are just solutions of the same fluid domain due to different boundary conditions. B is the known solution and A is the unknown solution we wish to solve. It is also assumed that external forces are only applied to the boundaries and not to the interior fluid domain. Let us define u and u' to be the velocities of flows A and B, respectively, and σ and σ' to be the stresses of flows A and B, respectively. In addition, the oscillation frequencies of the external forces that perturb the motions of the fluid are also assumed to be the same. The Lorentz reciprocal identity states:

$$\frac{\partial}{\partial x_j}\left(u_k'\sigma_{kj} - u_k\sigma_{kj}'\right) = 0 \tag{8.23}$$

which can be used in combination with the divergence theorem to generate boundary integral formulas. To prove this, we first rewrite the following equation

$$u_k'\frac{\partial\sigma_{kj}}{\partial x_j} = \frac{\partial}{\partial x_j}\left(u_k'\sigma_{kj}\right) - \sigma_{kj}\frac{\partial u_k'}{\partial x_j} = \frac{\partial}{\partial x_j}\left(u_k'\sigma_{kj}\right) - \left[\left(-P - \frac{2}{3}\mu\nabla_l u_l\right)\delta_{kj} + \mu\left(\frac{\partial u_k}{\partial x_j} + \frac{\partial u_j}{\partial x_k}\right)\right]\frac{\partial u_k'}{\partial x_j}$$
$$= \frac{\partial}{\partial x_j}\left(u_k'\sigma_{kj}\right) - \left[\left(-P + \frac{2}{3}\mu\omega i\frac{P}{P_0}\right)\delta_{kj} + \mu\left(\frac{\partial u_k}{\partial x_j} + \frac{\partial u_j}{\partial x_k}\right)\right]\frac{\partial u_k'}{\partial x_j} \tag{8.24}$$

and apply the continuity equation to replace the divergence of velocity with a pressure term:

$$u_k'\frac{\partial\sigma_{kj}}{\partial x_j} = \frac{\partial}{\partial x_j}\left(u_k'\sigma_{kj}\right) - \left((1 - \frac{2\mu\omega i}{3P_0})P\right)\omega i\frac{P'}{P_0} - \mu\left(\frac{\partial u_k}{\partial x_j} + \frac{\partial u_j}{\partial x_k}\right)\frac{\partial u_k'}{\partial x_j} \tag{8.25}$$

Interchanging the roles of the two flows in the above equation leads to another equation

$$u_k \frac{\partial \sigma'_{kj}}{\partial x_j} = \frac{\partial}{\partial x_j}\left(u_k \sigma'_{kj}\right) - \left((1 - \frac{2\mu\omega i}{3P_0})P'\right)\omega i \frac{P}{P_0} - \mu\left(\frac{\partial u'_k}{\partial x_j} + \frac{\partial u'_j}{\partial x_k}\right)\frac{\partial u_k}{\partial x_j} \tag{8.26}$$

Then, subtracting (8.25) from (8.24), and applying the assumption that there is no force inside the fluid domain, we get the final form of the Lorentz reciprocal identity:

$$\frac{\partial}{\partial x_j}\left(u'_k \sigma_{kj} - u_k \sigma'_{kj}\right) = u'_k \frac{\partial \sigma_{kj}}{\partial x_j} - u_k \frac{\partial \sigma'_{kj}}{\partial x_j} = u'_k u_k i\omega\rho_0 - u_k u'_k i\omega\rho_0 = 0 \tag{8.27}$$

Note that the reciprocal identity (8.23) is applicable only to the fluid domain with no external force inside.

Since the above derivation procedure already uses the stress tensor of the compressible Stokes flow, we conclude that the reciprocal identity is also applicable to the compressible Stokes flow.

8.6 The rigid body motions

We proved the double-layer jump identity in (4.9) in this section. This identity leads to some useful results about the rigid body motions, which we will need when deriving the single-layer integral equations.

First, rewrite the momentum equation in (8.3) using the stress tensor

$$\rho_0 i\omega u_j = \nabla_k \sigma_{jk} + g_j \delta(\vec{x}\text{-}\vec{x}_0) \tag{8.28}$$

Substituting the fundamental solutions into (8.28), the left-hand side and right-hand side of (8.28) become

$$\rho_0 i\omega u_j = \rho_0 i\omega \frac{1}{8\pi\mu} G_{jl} g_l$$
$$\nabla_k \sigma_{jk} + g_j \delta(\vec{x}-\vec{x}_0) = \frac{1}{8\pi} \frac{\partial}{\partial x_k} (T_{jkl} g_l) + g_j \delta(\vec{x}-\vec{x}_0) \tag{8.29}$$

Substituting (8.29) into (8.28) and rearranging the equations, we have

$$\frac{1}{8\pi} \frac{\partial}{\partial x_k} (T_{jkl} g_l) + g_j \delta(\vec{x}-\vec{x}_0) = \rho_0 i\omega \frac{1}{8\pi\mu} G_{jl} g_l$$
$$\rightarrow \frac{\partial T_{jkl}}{\partial x_k} = -8\pi\delta_{jl}\delta(\vec{x}-\vec{x}_0) + \rho_0 i\omega \frac{1}{\mu} G_{jl} \tag{8.30}$$

Next, we integrate the second equation in (8.30) over the closed surface defined by the boundary of a rigid body and apply the divergence theorem:

$$\iiint \frac{\partial T_{jkl}}{\partial x_k} dv = \iint T_{jkl} n_k ds = \iiint \left(-8\pi\delta_{jl}\delta(\vec{x}-\vec{x}_0) + \rho_0 i\omega \frac{1}{\mu} G_{jl} \right) dv \tag{8.31}$$

Then, we rearrange the above equation to yield:

$$\iint T_{jkl} n_k ds = \iiint -8\pi\delta_{jl}\delta(\vec{x}-\vec{x}_0) dv + \iiint \rho_0 i\omega \frac{1}{\mu} G_{jl} dv \tag{8.32}$$

The first volume integration on the right-hand side represents the effect of the point force distribution; the second term represents the effect due to the fluid inertia inside the boundary of the rigid body. But actually, there *is* no fluid inside the boundary of the rigid body. What is there is the mass of the rigid body that will be considered when analyzing the dynamic performance of the whole system. It is only for the simplicity of our derivation that we assume the fluid to be everywhere in the entire domain, including the volume occupied by the rigid body. Hence, it is not bad to simply assume the fluid inside the rigid body boundary to have zero density, since it does not really affect the motion of fluid outside the boundary when the

motion is "rigid." (The fluid inside the rigid body boundary moves like a rigid body; fluid motion is almost zero.) Another way to look at it is that the squeeze film effect we are studying is caused by the relative motion of the nearby surfaces, such as two plates approaching each other. We can imagine that the two plates have very thin rigid metal shells on the boundary with the interior filled with fluid, then it is easy to understand that the inside fluid affects almost nothing. In addition, if the above identity (8.32) is applied to the simplified linearized Stokes equation, this second term in (8.32) is exactly zero since the first term on the left-hand side of (8.28) is neglected. Thus we achieve a simplified identity that states:

$$-\frac{1}{8\pi}\iint T_{jkl}n_k\,ds \approx -\frac{1}{8\pi}\iiint -8\pi\delta_{jl}\delta(\vec{x}\text{-}\vec{x}_0)\,dv = \delta_{jl}\begin{cases}1 & \text{inside}\\ 1/2 & \text{right on}\\ 0 & \text{outside}\end{cases} \tag{8.33}$$

Where $\vec{n}$ is the outward normal. Note that the meaning of "inside" means "inside the boundary of the rigid body", which never happens if we deal with exterior problems only. "Outside" case is zero means that the contribution of this point force to the stress integration in (8.33) is zero, if the motion is perturbed by a point force outside the boundary. From the physical point of view, (8.33) describes force balance, where the left-hand side is the fluid force on the boundary, and the right-hand side is the external point force applied to the rigid body. If we add a constant velocity term to the integral of (8.33)

$$-\frac{1}{8\pi}\iint u_l T_{jkl}n_k\,ds \approx -u_l\frac{1}{8\pi}\iiint -8\pi\delta_{jl}\delta(\vec{x}\text{-}\vec{x}_0)\,dv = u_l\begin{cases}1 & \text{inside}\\ 1/2 & \text{right on}\\ 0 & \text{outside}\end{cases} \tag{8.34}$$

This implies that the double-layer integral due to a translating motion of the rigid object is zero if $\vec{x}_0$ is outside the boundary. Similarly, the double-layer integral due to the rotational motion of a rigid body is also zero if $\vec{x}_0$ is outside the boundary. Thus we reach a conclusion

that the double-layer integral makes zero contribution when the evaluation point is outside, if the fluid motion is perturbed solely by the rigid body motions.

8.7 The integral equations of the linearized compressible Stokes equation

Now let us derive the boundary integral equation using the identities (8.23) and (8.33). It is assumed that flow B is the fundamental solution due to point force $\vec{g}e^{i\omega t}\delta(\vec{x}\text{-}\vec{x}_0)$ at point $\vec{x}_0$. The amplitudes of velocity and stress are:

$$u'_k(\vec{x}) = \frac{1}{8\pi\mu} G_{kl}(\vec{x},\vec{x}_0)g_l$$
$$\sigma'_{kj}(\vec{x}) = \frac{1}{8\pi} T_{jkl}(\vec{x},\vec{x}_0)g_l \tag{8.35}$$

Substituting (8.35) into the reciprocal identity (8.23), we have

$$\frac{\partial}{\partial x_j}\left(G_{kl}(\vec{x},\vec{x}_0)\sigma_{kj} - \mu u_k T_{jkl}(\vec{x},\vec{x}_0)\right) = 0 \tag{8.36}$$

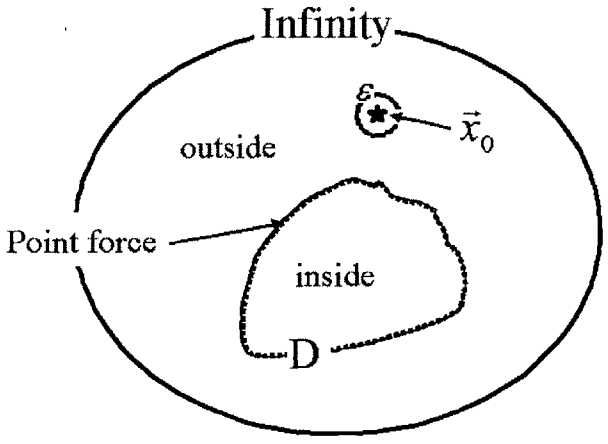

Figure 8.3: Fluid domain with a boundary D defined by the surface of a rigid body

Next we integrate (8.36) over the volume bounded by the boundary of D shown in Figure 8.3. Since the point force of flow B is only outside D, the reciprocal identity can be applied to the fluid inside D. Integration of (8.36) leads to:

$$\iiint_{\text{inside D}} \frac{\partial}{\partial x_j}\left(G_{kl}(\vec{x},\vec{x}_0)\sigma_{kj} - \mu u_k T_{jkl}(\vec{x},\vec{x}_0)\right)dv = \iint_D \left(G_{kl}(\vec{x},\vec{x}_0)\sigma_{kj}n_j - \mu u_k T_{jkl}(\vec{x},\vec{x}_0)n_j\right)ds = 0 \tag{8.37}$$

then let $f_k^{in} = \sigma_{kj}n_j$ be the external surface forces applied to the interior fluid only, and apply (8.33). Note that the point $\vec{x}_0$ is outside the boundary of the rigid body.

$$\iint_D G_{kl}(\vec{x},\vec{x}_0)f_k^{in}ds = \iint_D \mu u_k T_{jkl}(\vec{x},\vec{x}_0)n_j ds = 0 \tag{8.38}$$

Surprisingly, the integral is zero. This is because we have applied (8.33), which is based on the assumption that the double-layer integral of the internal fluid makes vanishing contributions to points outside the rigid surface. As a result, we get the following relationship from (8.38),

$$f_k^{in} = 0 \tag{8.39}$$

which is true if we assume the internal fluid density is zero. In other words, the rigid body motions of the boundary can move the internal fluid with zero forces because the internal fluid density is zero. Consequently, the contribution of the internal fluid inertia force to the outside point is zero.

Second, we investigate the fluid outside the rigid boundary D, extending all the way to infinity. As proved in the previous chapters, the surface integrals at infinity are zero. Hence

we can simply neglect them. But the point force is inside this fluid domain and it makes the reciprocal identity (8.23) invalid. This point force, therefore, has to be excluded from the fluid domain. In order to do so, we draw a small sphere with radius ε to exclude this point force (see Figure 8.3). Then the fluid domain is Ω = Infinity – D – sphere$_\varepsilon$. The volume integration in the following equation is converted to surface integration using the divergence theorem:

$$\iiint_{\Omega} \frac{\partial}{\partial x_j}\left(G_{kl}(\vec{x},\vec{x}_0)\sigma_{kj} - \mu u_k T_{jkl}(\vec{x},\vec{x}_0)\right)dv = 0 = \iiint_{\text{Infinity}} \cdots dv - \iiint_{\text{D}} \cdots dv - \iiint_{\text{sphere}_\varepsilon} \cdots dv$$
$$\Rightarrow 0 = \iint_{D}\left(G_{kl}(\vec{x},\vec{x}_0)\sigma_{kj}n_j - \mu u_k T_{jkl}(\vec{x},\vec{x}_0)n_j\right)ds + \iint_{\text{sphere}_\varepsilon}\left(G_{kl}(\vec{x},\vec{x}_0)\sigma_{kj}n_j - \mu u_k T_{jkl}(\vec{x},\vec{x}_0)n_j\right)ds \tag{8.40}$$

Both σ_{kj} and u_k are solutions of flow A, which is a constant in the sphere if ε is small enough:

$$\begin{aligned}
&\lim_{\varepsilon\to 0}\iint_{\text{sphere}_\varepsilon}\left(G_{kl}(\vec{x},\vec{x}_0)\sigma_{kj}n_j - \mu u_k T_{jkl}(\vec{x},\vec{x}_0)n_j\right)ds \\
&\approx \sigma_{kj}\lim_{\varepsilon\to 0}\iint_{\text{sphere}_\varepsilon} G_{kl}(\vec{x},\vec{x}_0)n_j ds - u_k \lim_{\varepsilon\to 0}\iint_{\text{sphere}_\varepsilon} \mu T_{jkl}(\vec{x},\vec{x}_0)n_j ds \\
&= 0 + 8\pi\mu u_l(\vec{x}_0) = 8\pi\mu u\ (\vec{x}_0)
\end{aligned} \tag{8.41}$$

(8.40) is then rewritten as:

$$\begin{aligned}
&\iiint_{\Omega}\frac{\partial}{\partial x_j}\left(G_{kl}(\vec{x},\vec{x}_0)\sigma_{kj} - \mu u_k T_{jkl}(\vec{x},\vec{x}_0)\right)dv = 0 \\
&= \iint_{D} G_{kl}(\vec{x},\vec{x}_0)\sigma_{kj}n_j ds - \mu\iint_{D} u_k T_{jkl}(\vec{x},\vec{x}_0)n_j ds + 8\pi\mu u_l(\vec{x}_0) = 0
\end{aligned} \tag{8.42}$$

Let $f_k^{out} = \sigma_{kj}n_j$ be the external surface forces applied to the rigid body surface, and we subtract (8.37) from (8.42) to remove the double-layer integral. In addition, we assume $f_k = f_k^{out} - f_k^{in}$:

$$u_l(\vec{x}_0) = -\frac{1}{8\pi\mu}\iint_D G_{kl}(\vec{x},\vec{x}_0) f_k ds \tag{8.43}$$

(8.43) is the single-layer integral equation we are looking for. Another way to prove (8.43) is simply to apply (8.33) to (8.42) and note that the point $\vec{x}_0$ is again outside the fluid domain $\Omega = \text{outside D} - \text{sphere}_\varepsilon$, making the second integral zero. Then we add $f_k^{in} = 0$ to yield the final form (8.43).

Equation (8.43) is derived by assuming the evaluation point is outside the boundary D; the same concept is also applicable to the inside of the fluid and the boundary. For an "interior" problem bounded by boundary D with $\vec{x}_0$ inside the boundary, the fluid domain is $\Omega = \text{D} - \text{sphere}_\varepsilon$. We further apply (8.33) to simplify the second integral:

$$\begin{aligned}
&\iiint_\Omega \frac{\partial}{\partial x_j}\left(G_{kl}(\vec{x},\vec{x}_0)\sigma_{kj} - \mu u_k T_{jkl}(\vec{x},\vec{x}_0)\right)dv = \iiint_D \cdots dv - \iiint_{\text{sphere}_\varepsilon} \cdots dv = 0 \\
&= \iint_D \left(G_{kl}(\vec{x},\vec{x}_0)\sigma_{kj} n_j - \mu u_k T_{jkl}(\vec{x},\vec{x}_0) n_j\right) ds - \iint_{\text{sphere}_\varepsilon} \left(G_{kl}(\vec{x},\vec{x}_0)\sigma_{kj} n_j - \mu u_k T_{jkl}(\vec{x},\vec{x}_0) n_j\right) ds \\
&= \iint_D G_{kl}(\vec{x},\vec{x}_0)\sigma_{kj} n_j ds - \mu \iint_D u_k T_{jkl}(\vec{x},\vec{x}_0) n_j ds - 8\pi\mu u_l(\vec{x}_0) \\
&= \iint_D G_{kl}(\vec{x},\vec{x}_0)\sigma_{kj} n_j ds + 8\pi\mu u_l^D - 8\pi\mu u_l(\vec{x}_0) \\
&\rightarrow u_l(\vec{x}_0) - u_l^D = \frac{1}{8\pi\mu}\iint_D G_{kl}(\vec{x},\vec{x}_0) f_k ds
\end{aligned} \tag{8.44}$$

The above equation (8.44) states that external forces applied to the interior fluid exist only when there are relative motions inside. u_l^D is the velocity of the translating motion of the rigid boundary.

To derive the integral equation when the point force of flow B is right on the boundary, imagine that the tiny sphere is split into two halves, so an equation similar to (8.42) is

$$\begin{aligned}&\iiint_{\Omega} \frac{\partial}{\partial x_j}\left(G_{kl}(\vec{x},\vec{x}_0)\sigma_{kj} - \mu u_k T_{jkl}(\vec{x},\vec{x}_0)\right)dv = 0\\ &= \iint_D G_{kl}(\vec{x},\vec{x}_0)\sigma_{kj}n_j ds - \mu \iint_D u_k T_{jkl}(\vec{x},\vec{x}_0)n_j ds + 4\pi\mu u_l(\vec{x}_0) = 0\end{aligned} \tag{8.45}$$

Then assuming $\vec{x}_0$ is right on the boundary and applying (8.33) to (8.45) to simplify the second integral:

$$\begin{aligned}&\iiint_{\Omega} \frac{\partial}{\partial x_j}\left(G_{kl}(\vec{x},\vec{x}_0)\sigma_{kj} - \mu u_k T_{jkl}(\vec{x},\vec{x}_0)\right)dv = 0\\ &= \iint_D G_{kl}(\vec{x},\vec{x}_0)\sigma_{kj}n_j ds + 4\pi\mu u_l(\vec{x}_0) + 4\pi\mu u_l(\vec{x}_0) = 0\\ &\rightarrow u_l(\vec{x}_0) = -\frac{1}{8\pi\mu}\iint_D G_{kl}(\vec{x},\vec{x}_0)f_k ds\end{aligned} \tag{8.46}$$

Since most of the simulation problems we are facing are exterior problems, we only care about the boundary and exteriors. We will stay with the single-layer integral equation (8.43) from now on.

Equation (8.43) implies that the double-layer integral, whose physical meaning is the velocity due to surface stress distribution, makes vanishing contributions when the boundaries move like rigid bodies. Similar derivations are also extendable to the pressure integral equation, and the single-layer pressure integral equation is:

$$P(\vec{x}_0) = -\frac{1}{8\pi}\iint_D p_k(\vec{x},\vec{x}_0)f_k ds \tag{8.47}$$

Both (8.43) and (8.47) are also applicable to the simplified linearized compressible Stokes problems.

8.7 Kernel integrations

Kernel integration is an important part of the BEM. If the kernel integration is not accurate, or numerically unstable, there is little hope that the final solution can be accurate. Of course, accuracy depends on not only the kernel integration algorithms but also the nature of the kernels. We discuss integrating the linearized compressible Stokes kernels in this section.

a. Integrating the G_{jk} kernel

The G_{jk} kernel is a fast-decaying oscillating kernel. The panel size should, therefore, be much smaller than the wave length of the kernel; this is usually satisfied at the discretization step. When the evaluation point is right on the panel, the semi-analytical method discussed in Section 3.2 can be applied. This method is accurate because the evaluation point is the centroid of the panel.

$$\iint G_{kj} ds = \iint G_{kj} r dr d\theta = \int \tilde{G}_{kj} d\theta$$

where

$$G_{kj} = 2\delta_{kj}\left[\frac{e^{-\lambda_2 R}}{R} - C_0\left(\frac{\left(e^{-\lambda_1 R} - e^{-\lambda_2 R}\right)}{R^3} + \frac{\left(\lambda_1 e^{-\lambda_1 R} - \lambda_2 e^{-\lambda_2 R}\right)}{R^2}\right)\right] +$$
$$+ 2C_0 \hat{x}_k \hat{x}_j \left[3\frac{\left(e^{-\lambda_1 R} - e^{-\lambda_2 R}\right)}{R^5} + 3\frac{\left(\lambda_1 e^{-\lambda_1 R} - \lambda_2 e^{-\lambda_2 R}\right)}{R^4} + \frac{\left(\lambda_1^2 e^{-\lambda_1 R} - \lambda_2^2 e^{-\lambda_2 R}\right)}{R^3}\right] \tag{8.48}$$

and

$$\tilde{G}_{kj} = 2\delta_{kj}\left[\frac{1 - e^{-\lambda_2 R}}{\lambda_2} - C_0\left((\lambda_1 - \lambda_2) + \frac{\left(e^{-\lambda_1 R} - e^{-\lambda_2 R}\right)}{R}\right)\right] +$$
$$+ 2C_0 \frac{\hat{x}_k \hat{x}_j}{R^2}\left[-3\frac{\left(e^{-\lambda_1 R} - e^{-\lambda_2 R}\right)}{R} - \left(\lambda_1 e^{-\lambda_1 R} - \lambda_2 e^{-\lambda_2 R}\right) - 2(\lambda_1 - \lambda_2)\right]$$

When the evaluation point is not on the panel, a serial expansion is applied:

$$G_{ij} = 2\frac{\delta_{ij}}{R}\left\{\left[1 - C_0\left(-\frac{\lambda_1^2 - \lambda_2^2}{2}\right)\right] + \left[-\lambda_2 - C_0\left(\frac{\lambda_1^3 - \lambda_2^3}{3}\right)\right]R + \left[\frac{\lambda_2^2}{2} - C_0\left(-\frac{\lambda_1^4 - \lambda_2^4}{8}\right)\right]R^2 + O(R^3)\right\}$$
$$+ 2C_0\frac{\hat{x}_i\hat{x}_j}{R^3}\left\{\left[-\frac{\lambda_1^2 - \lambda_2^2}{2}\right] + \left[\frac{\lambda_1^4 - \lambda_2^4}{8}\right]R^2 + O(R^3)\right\} \tag{8.49}$$

Next we define:

$$\overline{G_{ij}} = G_{ij} - \left\{2\frac{\delta_{ij}}{R}\left(1 - C_0\left(-\frac{\lambda_1^2 - \lambda_2^2}{2}\right)\right) + 2C_0\frac{\hat{x}_i\hat{x}_j}{R^3}\left(-\frac{\lambda_1^2 - \lambda_2^2}{2}\right)\right\} \tag{8.50}$$

The analytical flat panel kernel integration algorithm is applied to the terms in the curly braces, those terms are the weakly singular part; the remaining part $\overline{G_{ij}}$ is a smoother function that can be integrated using a high-order cubature formula. Since the kernel is a fast-decaying oscillating kernel, we use this approach to avoid large numerical errors when evaluation points are close to the panel. If the evaluation point is very far from the panel, a cubature method may be applied to directly integrate the G_{jk} kernel.

b. Integrating the pressure kernel

A similar combined approach is applied to the pressure kernel, which is

$$p_j = \frac{2\hat{x}_j}{\left(1 + \frac{4\omega\mu i}{3P_0}\right)}\left(\frac{1}{R^3} + \frac{\lambda_1}{R^2}\right)e^{-\lambda_1 R} \tag{8.51}$$

However, two important issues must be taken care of. The first is the so-called double-layer jump described in (6.5). A special procedure is added to the kernel integration algorithm to

adjust the layer jump so that the pressure is continuous when the evaluation point approaches the boundary from the exterior.

The second issue is the difficulty associated with the semi-analytical approach when the evaluation point is right on the panel. Since analytical formulas have not been found for $\int \frac{e^{-\lambda_1 R}}{R} dR$, we divide the kernel into two parts and apply different methods. The first part is a smooth function without singularity; it is integrated using a cubature method:

$$p_j^1 = \frac{2\hat{x}_j}{\left(1+\frac{4\omega\mu i}{3P_0}\right)} \frac{e^{-\lambda_1 R}-1}{R^3} \tag{8.52}$$

The remaining part is integrated using the semi-analytical method:

$$p_j^2 = \frac{2\hat{x}_j}{\left(1+\frac{4\omega\mu i}{3P_0}\right)} \left(\frac{1}{R^3}+\frac{\lambda_1}{R^2}e^{-\lambda_1 R}\right)$$

and

$$\int p_j^2 dR = \frac{2\hat{x}_j}{\left(1+\frac{4\omega\mu i}{3P_0}\right)R} \int \left(\frac{1}{R^2}+\frac{\lambda_1}{R}e^{-\lambda_1 R}\right) R dR = \frac{2\hat{x}_j}{\left(1+\frac{4\omega\mu i}{3P_0}\right)R} \left(\log R + 1 - e^{-\lambda_1 R}\right) \tag{8.53}$$

Note that, because of anti-symmetry,

$$\int_0^{2\pi} \frac{2\hat{x}_j}{\left(1+\frac{4\omega\mu i}{3P_0}\right)R} \log 0 d\theta = \int_0^{2\pi} \frac{2\cos(\theta+\psi_j)}{\left(1+\frac{4\omega\mu i}{3P_0}\right)} \log 0 d\theta = 0 \tag{8.54}$$

with $\frac{\hat{x}_j}{R} = \cos(\theta+\psi_j)$

if the evaluation point is right on the panel but not on the edges.

The unsteady incompressible Stokes kernels can be integrated using techniques similar to those we discussed above. Integrations of the simplified linearized compressible Stokes kernels are calculated using the analytical flat panel integration algorithm, since the kernels are much simpler.

8.8 The numerical robustness of the simplified linearized compressible Stokes integral equation

The complicated fast-decaying oscillating kernels of the linearized compressible Stokes equation have caused many numerical difficulties. However, the worst thing about these kernels is not the difficulties in kernel integration, but rather the performances related to the oscillating nature of the kernels. In Figure 8.4 we plot the absolute values of the real and imaginary parts of the six independent velocity kernels. The horizontal coordinate is the distance between the evaluation point and the source point. The oscillation frequency is assumed to be 1 MHz, and the corresponding frequency parameters are:

$$\begin{aligned} \lambda_1 &= 1.6323\times 10^{1} + 2.1418\times 10^{4} i \\ \lambda_2 &= 4.4792\times 10^{5} + 4.4792\times 10^{5} i \end{aligned} \tag{8.55}$$

In order to achieve good global accuracy, the size of the panel should be less than $\frac{1}{10} \sim \frac{1}{20}$ of the shortest wave length, that is, $\frac{2\pi}{\mathrm{Im}(\lambda_2)} \sim 14\mu m$. If the oscillation frequency is higher and the size of the panel is picked as $0.2\mu m$, it takes more than 20,000 quadrilateral panels to discretize a $20\mu m \times 20\mu m \times 1\mu m$ plate. The wave length calculation result is reflected by the real parts of the kernels plotted in Figure 8.4. It is clear that there is a sign change, pointed by

the right arrow, when the distance between the evaluation point and the source point is about $2\mu m$.

However, the left arrow points to another sign change that indicates the shortest wave length to be shorter than 1 nm. What happens is that the kernel is so complicated that parts of it cancel with other parts in the same kernel, and this causes the sign changes of some imaginary parts at a shorter distance.

Clearly, this bad performance severely limits the applications of the linearized compressible Stokes kernels, and makes them useful only for simulating nanometer-scaled devices. We have tried to calculate using those complicated kernels but were only able to get sound results when simulating devices with feature sizes on the order of 1 nm. But the good news is that the simplified linearized compressible Stokes kernels are not oscillating kernels. Hence, they are not limited by the wave length, and we can expect good results using the simplified linearized compressible Stokes integral equations.

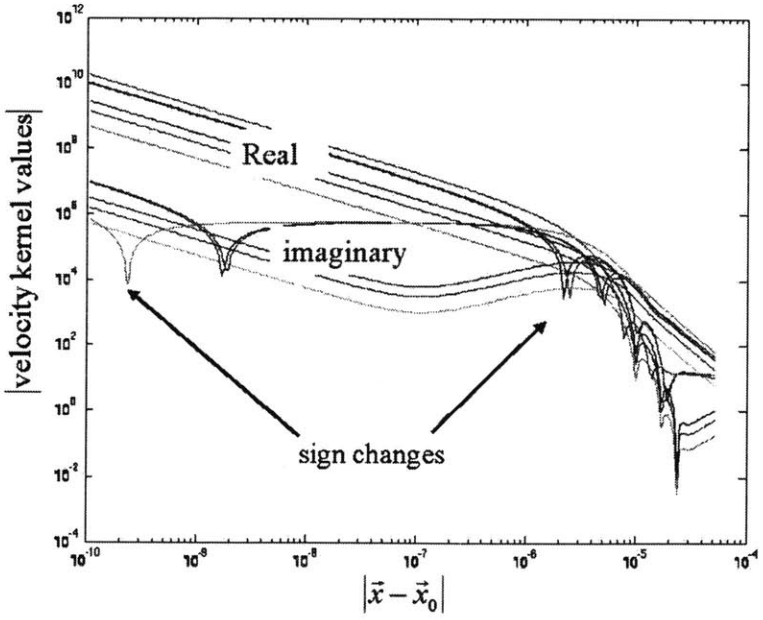

Figure 8.4: The absolute values of the real and imaginary parts of the linearized compressible Stokes velocity kernels

Figure 8.5 shows the absolute values of the unsteady incompressible Stokes velocity kernels, which are also fast-decaying oscillating kernels. The oscillations of the imaginary parts at short distances are suspicious, but the values are too low to cause bad effects. It is possible that the oscillations are caused by the numerical errors we frequently encounter. Generally speaking, the performances of the unsteady incompressible Stokes velocity kernels are much better, although, the panel size is still limited by the shortest wave length.

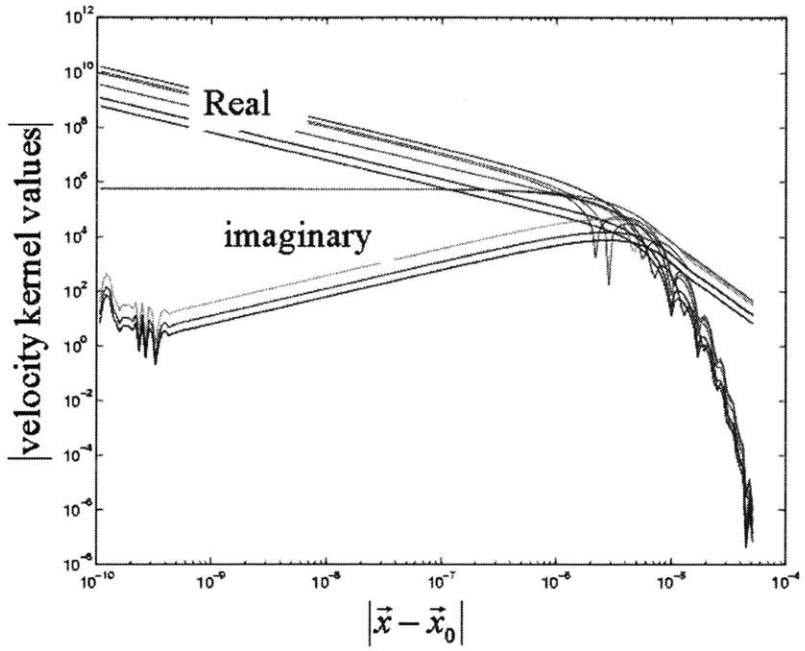

Figure 8.5: The absolute values of the real and imaginary parts of the unsteady incompressible Stokes velocity kernels

8.9 The null space issue of the linearized compressible Stokes BEM operators

Fortunately, the linearized compressible Stokes BEM operators are full-rank. The following are the linearized compressible Stokes equations:

$$\begin{cases} \rho i\omega\vec{u} = -\nabla P + \mu\nabla^2\vec{u} + \frac{1}{3}\mu\nabla(\nabla\bullet\vec{u}) \\ i\omega P + P_0\nabla\bullet(\vec{u}) = 0 \end{cases} \tag{8.56}$$
$$P_{total} = P_0 + Pe^{iwt}$$

The pressure oscillation amplitude already appears in the continuity equation, so it is not necessary to add extra boundary conditions since the solution is uniquely defined. The null space issue only arises when (8.56) is used to calculated quasi-steady cases (frequency equals to zero). Then the above equation is reduced to the steady incompressible Stokes equation.

The linearized compressible Stokes equation describes the behavior of a compressible flow at a certain ambient air pressure due to perturbations. It is different from the incompressible Stokes equation in which constant pressure can be added to the pressure term as desired. In the compressible case, a constant pressure may be added to the total pressure in the third equation of (8.56), but this changes the ambient air pressure P_0 in the second equation. Once the equation is given, such addition is not allowed and no pressure boundary condition is needed. Therefore the oscillation amplitude of the pressure is uniquely defined by the system equation (8.56), if the given velocity boundary condition is proper.

8.10 Fluid compression due to geometry

Significant air compression at low speed is often due to confined geometry, as with a piston in a cylinder with small orifices (see Figure 8.6). If the orifices are small, part of the air is trapped inside when the piston moves rapidly. Now, consider removing the second viscosity term in the simplified momentum equation using the linearized continuity equation:

$$\begin{cases} 0 = -\nabla P + \mu\nabla^2\vec{u} + \frac{1}{3}\mu\nabla(\nabla \bullet \vec{u}) \\ i\omega P + P_0\nabla \bullet (\vec{u}) = 0 \end{cases} \qquad (8.57)$$
$$\rightarrow 0 = -(1 + \frac{i\omega\mu}{3P_0})\nabla P + \mu\nabla^2\vec{u}$$

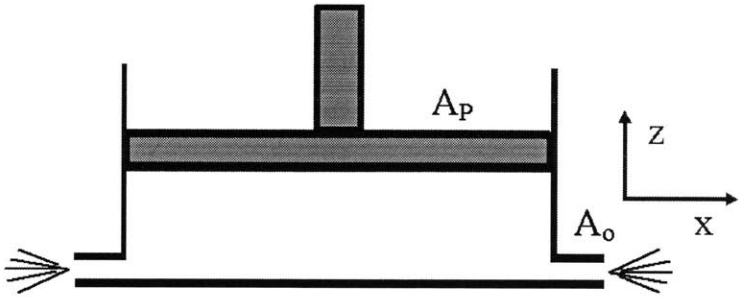

Figure 8.6: Air compression due to confined geometry

Accordingly, the pressure losses across narrow pipes can be expressed as:

$$(1 + \frac{i\omega\mu}{3P_0})\nabla P = \mu\nabla^2\vec{u}$$
$$(1 + \frac{i\omega\mu}{3P_0})\frac{\Delta P}{L} = \mu\frac{u_x}{z^2} \quad \Rightarrow \quad \frac{\Delta P}{L} \sim \frac{\mu}{C_1}\frac{u_x}{z^2} \qquad (8.58)$$

where L is the length of the pipe, u_x is the velocity along the pipe, $C_1 = 1 + \frac{i\omega\mu}{3P_0}$, and z is the dimension of the cross-section. The pressure drop can be estimated as $\frac{\Delta P}{P_0} \sim \frac{\mu}{C_1}\frac{Lu_x}{P_0 z^2}$. In the two circular plates example shown in Figure (8.1), we have $u_x = \pi f R\varepsilon$, $C_1 \sim 1$, $z = h$, and

$L = R$. The relative pressure drop $\frac{\Delta P}{P_0}$ due to nondimensional oscillation amplitude ε is the best indication of the strength of the compression effect:

$$\frac{\Delta P}{P_0} = \frac{\mu}{C_1}\frac{Lu_x}{P_0 z^2} = \frac{\mu R\,\pi f R\,\varepsilon}{P_0 h^2} = \frac{\mu\pi R^2 f}{h^2 P_0}\varepsilon \tag{8.59}$$

Clearly, the coefficient $\frac{\mu\pi R^2 f}{h^2 P_0}$ is equivalent to the squeeze number $\frac{24\,\pi\mu f R^2}{P_0 h^2}$. Therefore, we should use the nondimensional number $\frac{\Delta P}{P_0} \sim \frac{\mu}{C_1}\frac{Lu_x}{P_0 z^2}$ or the squeeze number to check the strength of the compression effect generated by the oscillating motions.

Chapter 9

The compressible FastStokes solver and numerical simulation results

The compressible FastStokes solver solves the linearized compressible Stokes equation and the simplified linearized compressible Stokes equation using the PFFT-accelerated BEM approach. A major numerical difference between the compressible FastStokes and the incompressible FastStokes is the complex number operations. The compressible FastStokes uses complex numbers because of the $i\omega$ terms in (8.57). As a result, both the computational cost and the memory requirement of the compressible FastStokes increase.

When solving the simplified linearized compressible Stokes equation, the compressible FastStokes solver is reasonably fast and numerically stable. Also, the discretization is not limited by wave lengths; this avoids quite a lot of trouble. Most numerical results presented in this chapter are simulated using the simplified linearized compressible Stokes model.

In addition to showing the effectiveness of the FastStokes simulation program, we will also discuss calibrating air compression effect in micromachined devices and show the approximate validity ranges of the incompressible assumption in air-packaged MEMS devices. Such calibration has been unavailable in the literature because of the lack of computationally friendly 3-D fluid models and fast simulation approaches.

9.1 A thin air film between two square parallel plates

Let us consider two square parallel plates: the top plate oscillates up and down and the bottom plate is fixed. The sizes of the two plates are $10\mu m \times 10\mu m \times 1\mu m$ with an air gap of $1\mu m$ in between. The device is assumed to be packaged at room temperature and oscillates at 10

MHz. Some important parameters are: $P_0 = 1.013 \times 10^5$ pa, $\rho = 1.177\,\text{kg/m}^3$, and $\mu = 1.843 \times 10^{-5}\,\text{N}\cdot\text{s/m}^2$. In the following pictures we show the FastStokes-simulated damping forces and spring forces on the bottom side of the top plate using the simplified linearized compressible Stokes model. It is clear that the simplified linearized compressible Stokes model captures the weak fluid compression effect. If the steady incompressible Stokes model is used, the spring force solution is zero and the damping force solution is not accurate.

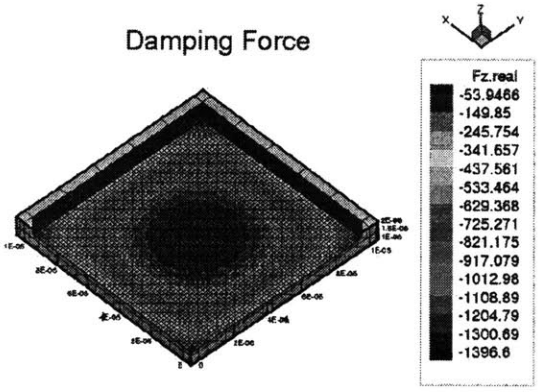

Figure 9.1: The damping force distribution on the bottom side of the top plate

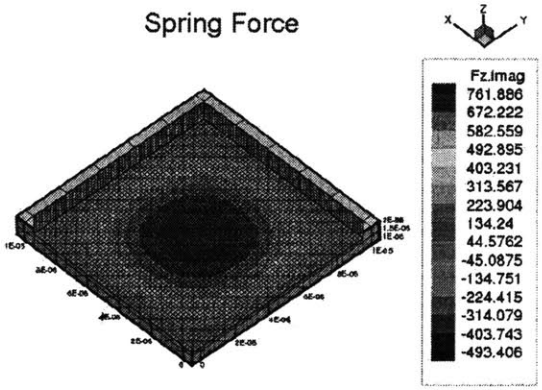

Figure 9.2: The spring force distribution on the bottom side of the top plate

9.2 Modeling the air compression effect in squeeze film

It has been reported that the air compression in squeeze film structures causes significant spring forces, which might shift the resonance frequencies substantially. The questions are, how to accurately model the damping forces and spring forces, and when to use a compressible model or an incompressible model? These are tough questions that have not been answered. The reason has been the lack of fast and accurate 3-D fluid solvers.

In this section we compare the numerical solutions of the incompressible FastStokes solver and the compressible FastStokes solver (which solves the simplified linearized compressible Stokes equation) with the analytical solutions of the linearized compressible Reynolds equation in the square-plate case. Again, we assume that these two square plates are packaged in air at 1atm, room temperature, and that the bottom plate is fixed and the top plate oscillates up and down. The oscillation frequency and aspect ratio are varied to change the strength of the compression effect. It is assumed that the air gap and the thickness of the plates are both $h = 1\mu m$ and the length of the square plate L is varied to change the aspect ratio $\frac{L}{h}$. A non-dimensional number called squeeze number, defined as $\frac{24\pi\mu f}{P_0}(Aspect\ Ratio)^2$ in the two square plate case, is commonly used to indicate the strength of the squeeze film effect, where f is the oscillation frequency. A larger squeeze number indicates a stronger squeeze film effect, and thus a larger spring force. The following two figures show the non-dimensional force on the top plate, which is defined as $\frac{Fh}{AP_0\varepsilon'}$, where F is the force, $A = L^2$ is the area of the thin air film, and $\varepsilon = \frac{\varepsilon'}{h}$ is the non-dimensional oscillation amplitude.

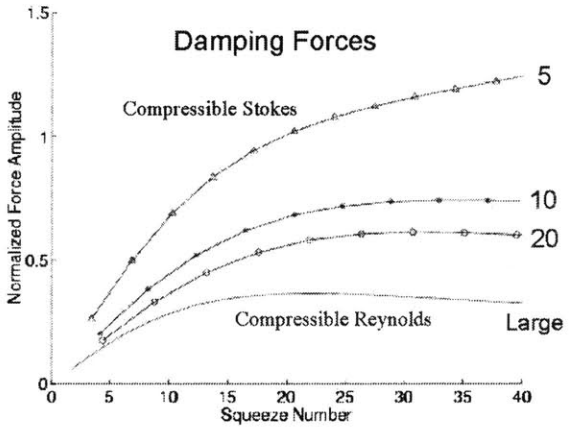

Figure 9.3: The non-dimensional total damping forces on the top plate

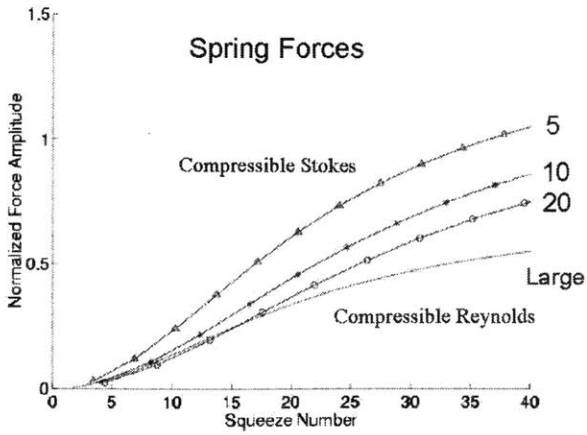

Figure 9.4: The non-dimensional total spring forces on the top plate

The "Compressible Stokes" in the above figures means the simplified linearized compressible Stokes equations; the "Compressible Reynolds" means the linearized compressible Reynolds equation. Note that the analytical Reynolds equation solution neglected the edge effect (or the entrance effect), which make it accurate only when the aspect ratio is extremely large. Some researchers have added adjustments to the analytical Reynolds equation solution to compensate for the edge effect, and they found that the original analytical solutions are very inaccurate. The original analytical solution predicts about 65% less damping force when aspect ratio = 5, 25% less when aspect ratio = 20, and 10% less when aspect ratio = 100. In addition to the neglected edge effect, the Reynolds equation solution also neglects other 3-D effects, such as the force on the top surface of the top plate.

In Figure 9.3 and Figure 9.4, it is easy to see that the damping forces and the spring forces predicted by the simplified linearized compressible Stokes equation are more meaningful. When the aspect ratio gets larger, both damping force solutions and spring force solutions converge to the solutions of the linearized compressible Reynolds equation.

In addition to the above solutions, the damping force solutions of the steady incompressible Stokes equation (labeled as "incompressible Stokes") are added to Figure 9.5. When the fluid compression effect is small, say squeeze number < 5, the solutions of the steady incompressible Stokes equation and those of the simplified linearized compressible Stokes equation match very well. This indicates that the incompressible fluid model is good enough for the damping force calculation. However, when the squeeze film effect is stronger, say, squeeze number > 10, a compressible fluid model is a must if an accurate damping force solution is desired. Again, the solution of the linearized compressible Reynolds equation is not accurate unless the aspect ratio is large.

Many researchers have a wrong impression that only compressible fluid models give good solutions when modeling squeeze film damping. The results we get here clearly indicate that if only the damping force is desired, an incompressible fluid model is capable of producing a good result when the squeeze film effect is not too strong.

In addition, the above simulation results show that the compressible FastStokes solver is a much more powerful tool for exploring the fluid damping and spring effects in MEMS. Whenever the fluid compression effect is not clear, it is suggested that the compressible FastStokes solver be used.

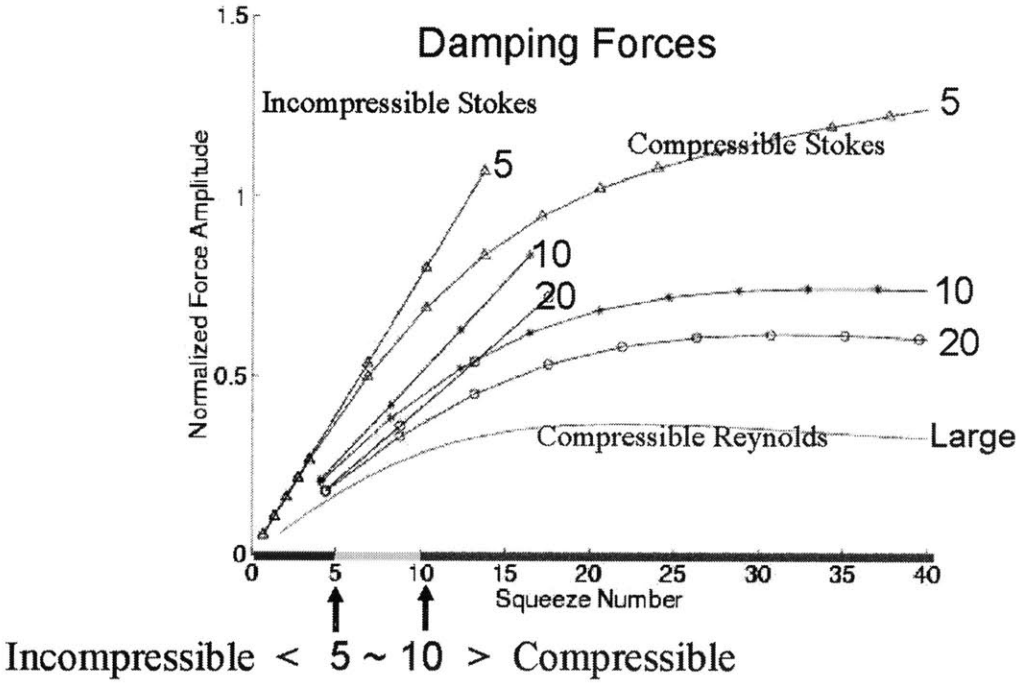

Figure 9.5: The non-dimensional total damping forces of the three fluid models

9.3 The spring forces

To further understand the importance of accurate spring force simulations and the applications of the compressible FastStokes solver, this section discusses resonance frequency shifts due to the spring forces generated by air compression. Such frequency shift is one critical aspect in design and it must be considered.

Again, we use the two square plate example used in the previous section (see Figure 9.6). The top plate is assumed to be attached to the substrate by tethers, and the resonance frequency is calculated as $f_0 = \frac{1}{2\pi}\sqrt{\frac{K}{m}}$ before air compression is considered, where K is the stiffness of the tethers. We also assume the top plate oscillates at the frequency f_0 and that the device is packaged in air. Then the spring force of the squeeze film contributes a $K_{\text{squeeze film}}$ that shifts the actual resonance frequency. The comparison of $K_{\text{squeeze film}}$ and K gives a good understanding of the effects of fluid compression. Again, the horizontal coordinate of Figure 9.7 is the squeeze number at the designed resonance frequency (without air spring force). The designed resonance frequency of the device is varied by changing the stiffness of the tethers.

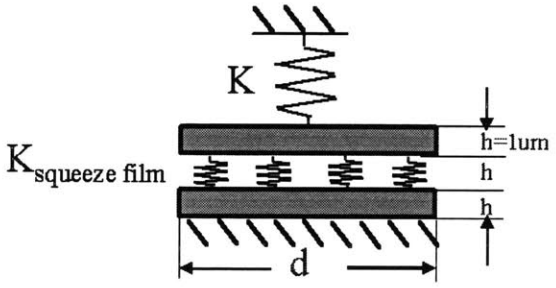

Figure 9.6: The two parallel square plates

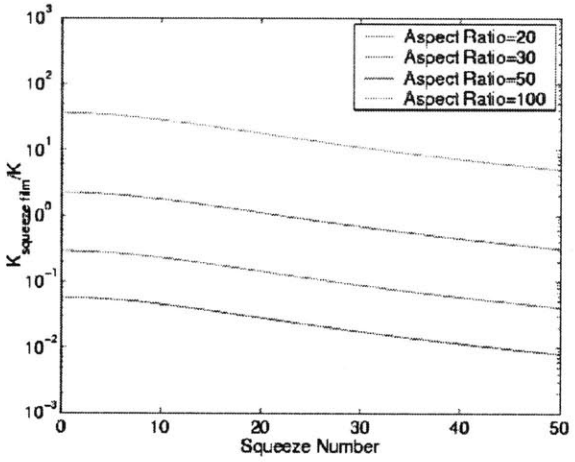

Figure 9.7: The stiffness shift due to spring forces

The result shown in Figure 9.7 reflects what people have observed experimentally: the stiffness might be shifted by orders of magnitude due to the spring forces generated by air compression, especially when the aspect ratio is large. An even more interesting conclusion drawn from Figure 9.7 is that the stiffness shifts converge when resonance frequency approaches zero. This can be explained by the following equations:

$$K = 4\pi^2 f_0^2 m = 4\pi^2 f_0^2 \rho_{si} Ah$$

$$F_{spring} = \text{non dimensional spring force;} \quad \text{squeeze number } \sigma = \frac{24\pi\mu f_0}{P_0}\left(\frac{d}{h}\right)^2 \tag{9.1}$$

$$K_{squeeze\ film} = \frac{AP_0}{h} F_{spring} \approx \frac{AP_0}{h}\left\{\frac{64\sigma^2}{\pi^8}\sum_{\substack{m,n \\ odd}} \frac{1}{(mn)^2\left(\left[m^2 + (n/\beta)^2\right]^2 \sigma^2/\pi^4\right)}\right\}$$

The last equation in (9.1) is the analytical spring force solution of the linearized compressible Reynolds equation, which shows that $K_{\text{squeeze film}} \propto \sigma^2$ and $K_{\text{squeeze film}} \propto f_0^2$ when the squeeze number is small. Since the stiffness of the tether is also proportional to the square of the oscillation frequency, i.e. $K \propto f_0^2$, Therefore, when resonance frequency is low, changing the stiffness of the tether does not change the relative resonance frequency shift.

Of course, all conclusions above are drawn from the simple two-square-plate example. If more conclusions are needed for general complicated geometries, the compressible FastStokes solver is the best solution to difficult 3-D simulation problems.

9.4 More complicated examples

In this section we study the damping and spring forces on irregular structures using the compressible FastStokes solver. An etch hole example and a high-aspect-ratio comb-drive example are given.

9.4.1 Etch holes on a large proof mass

Etch holes are frequently used to help the etchant reach underneath the structural layers, so that the sacrificial layers can be quickly removed and the structural layers can be released. If the etchant also attacks useful structures, carefully designed etch holes can reduce the length of the etching period and lower the side effects. Figure 7.12 shows arrays of etch holes on the proof mass of ADXL76. Clearly, the existence of etch holes affects the damping performance

of the structures. In this example we consider two devices: the first device is our favorite two-parallel-plate oscillator; the second device is similar but with a square etch hole on the top plate. We assume all plates are $20\mu m \times 20\mu m \times 1\mu m$, and the air gap is $1\mu m$. The top plates are assumed to be oscillating at 2 MHz, with a velocity amplitude 0.1 m/s. The squeeze number of the two-parallel-plate oscillator is 11, which means the compression effect is reasonably strong. Therefore, we use the simplified linearized compressible Stokes flow model. Figures 9.8 and 9.9 show the damping force distribution on the bottom side of the top plate. We use the same scale in both figures, so that the force reduction in the etch hole case is clearly seen.

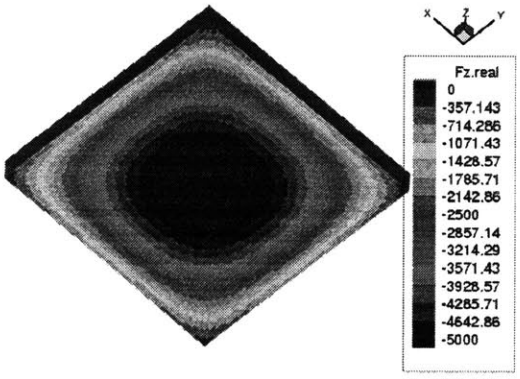

Figure 9.8: The damping force on the bottom side of the top plate

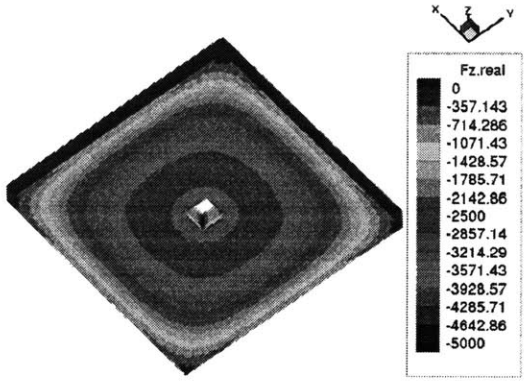

Figure 9.9: The damping force on the bottom side of the top plate, with a square etch hole at the center

Table 9.1 lists damping forces and spring forces on the top plates of both devices. For the second device, we calculated two cases: one in which the size of the etch hole is $2\ \mu m \times 2\ \mu m$ and one in which the size of the etch hole is $4\ \mu m \times 4\ \mu m$.

Table 9.1: Damping and spring force reductions due to the existence of an etch hole

Etch Hole	Forces	Plate only	With etch hole	Area reduction	Forces reduction
2 μm	Damping	127 nN	106 nN	1 %	17%
	Spring	46 nN	30 nN		35%
4 μm	Damping	127 nN	77 nN	4 %	39%
	Spring	46 nN	15 nN		68%

It is clear that the etch holes substantially reduce the damping and spring forces. A $2\ \mu m \times 2\ \mu m$ etch hole reduces the area of the top plate by only 1%, but the damping force is reduced by 17%; and the spring force is reduced even more significantly, by 35%. The $4\ \mu m \times 4\ \mu m$ etch hole reduces the damping force by 39%, and it reduces the spring force by 68%.

9.4.2 High aspect-ratio comb-drives

The recent development in bulk-micromachining technologies makes fabricating thick structures, such as high aspect-ratio comb-drive structures, easier. It is almost certain that the lateral motion of the thin-film structure does not cause significant air compression, but the same conclusion may not apply to thick structures. In this example we study a structure with a 3-finger static comb-drive and a 2-finger movable comb-drive (see Figure 9.10). The lengths of the fingers are 40um, with an overlap of 20 um; the air gap in the overlap area is 3 um; the thickness of the comb is 20 um. The movable comb is assumed to oscillate at 5MHz in both x and y directions, with an oscillation amplitude of 0.1 um.

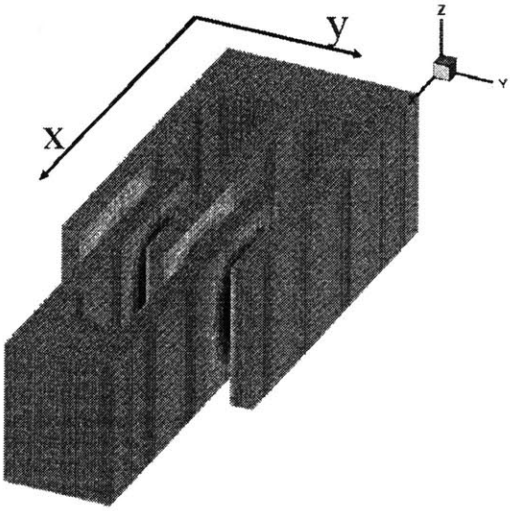

Figure 9.10: The thick comb-drive structure and y-direction force distribution when the movable comb oscillates in y-direction

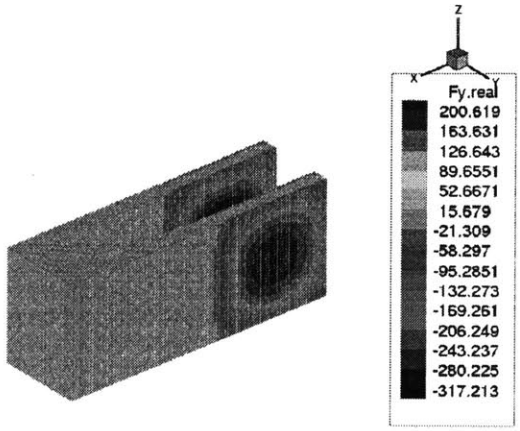

Figure 9.11: The damping force on the movable comb when it oscillates in y-direction

Both Figure 9.10 and Figure 9.11 show the y-direction damping distributions. Table 9.2 lists forces in the motion directions when the movable comb oscillates in x and y directions.

Table 9.2: Damping and spring forces on the movable comb-drive

Motion direction	Frequency	Squeeze Number	Forces in motion direction	
Y	5 MHz	2.7	Damping	1092.2 nN
			Spring	99.7 nN
X	5 MHz		Damping	78.2 nN
			Spring	0.64 nN

The y-direction motion is strongly affected by the squeeze film effect. However, because the aspect ratio of the overlapped area is only $20/3 \sim 6.6$, and also because the oscillation frequency is not very high, the squeeze film effect is not strong enough to generate large spring forces. If the resonance frequency of the y-direction motion in vacuum is assumed to be 5 MHz, the spring force shifts the resonance frequency of the y-direction motion by only

0.0028%; the quality factor in this case is 1621. Table 9.2 also shows the x-direction motion is very lightly damped, because there is little squeeze film effect.

Chapter 10

Conclusions

This thesis describes our most recent achievements in developing FastStokes 3.0, a fast Stokes flow simulation program based on the pre-corrected FFT accelerated boundary element method.

The FastStokes simulation program is designed to perform fast full 3-D fluid simulations on geometrically complicated MEMS devices packaged in gases at ambient air pressure or in liquids. Existing 3-D Navier-Stokes equation solvers using FEM or FDM have been found to be inadequate; they may take days to run to completion or fail when geometries get complicated. Both an incompressible FastStokes solver that solves the steady incompressible Stokes equation and a compressible FastStokes solver that solves the simplified linearized compressible Stokes equation are developed.

We have chosen to use the boundary element method because it needs only the boundary discretization, which is a significant reduction of discretization complexity compared with the volume discretization used in FEM or FDM. A standard piece-wise centroid collocation approach is applied in FastStokes (see Chapter 2). The implementation of the PFFT algorithm in combination with the GMRES algorithm dramatically reduces the computational cost to $O(n\log(n))$. As a result, FastStokes is fast and memory-efficient.

The singular BEM operator problem of the incompressible Stokes integral equations is difficult and has not been fully addressed in the literature. Since the governing equations do not have a unique solution without a pressure boundary condition, the incompressible Stokes integral operators are singular with null space vectors defined by the surface normals of the

objects. If the null space of the singular BEM operators is ignored or not treated carefully, numerical errors and solution discontinuities are unavoidable in some cases. The two-step approach presented in Chapter 6 successfully solves this problem. In the first step, a null-space-removing procedure is added to the GMRES algorithm to solve a null-space-free solution; this modification reduces numerical error and it also leads to a fast convergence. The second step is a pressure-pinning method that modifies the null-space-free solution to make the final solution unique and correct. This step removes the solution discontinuities across closed surfaces. The two-step method is easy to implement and makes the FastStokes program numerically robust.

An analytical flat panel kernel integration algorithm is implemented in the FastStokes solver to improve the accuracy of the nearby interaction calculations. This analytical method was originally developed by Newman [19]; we added a second recurrence scheme to calculate contour integrations (Section 3.3 and Appendix). This algorithm is also fast because it integrates the six independent Stokes kernels simultaneously using recursive schemes. Based on this analytical algorithm, we further developed a curved panel integration algorithm using a mapping method. A curved panel is first mapped to a flat panel, and the non-singular part of the integration is approximated using polynomial. The approximated expression is then calculated using the analytical flat panel integration algorithm. An accurate mapping is the critical point of this method; we use the ideal reference flat panel defined in Chapter 5. This method has been tested and it is found to be very accurate. This curved panel integration algorithm may provide great accuracy benefits to BEM solvers using curved panels.

The accuracy and the fast speed of the steady incompressible FastStokes solver have been proved by several numerical experiments. A sphere, a resonator, the ADXL76 accelerometer, and a micro-mirror have been simulated and the results are presented in Chapter 7. The simulations are usually finished in minutes on a PC computer, although very fine discretizations using more than 40K panels may take more than an hour's time. We found close matches between numerical simulation results and the testing results; the differences are within engineering accuracy (5-10% differences). This simulation program is, therefore,

qualified to be used to verify and optimize designs. The Couette flow model solution and the 1-D Stokes flow model solution are also compared with the FastStokes simulation result and the experimental result in the comb drive resonator case to show the inaccuracy of these simple fluid models (30~50% errors). Simple 1-D or 2-D fluid models can certainly give good estimates in special cases, but in general, they are not accurate enough for complicated 3-D geometries, particularly when the 3-D effect is significant.

The linearized compressible Stokes model and the simplified linearized compressible Stokes model are derived to model the compression effect in air flows perturbed by the oscillating motions of MEMS devices. The linearized compressible Stokes model is a general fluid model that models both compression effect and transient effect. However, the simplified linearized compressible Stokes model, which neglects only a trivial term in the linearized compressible Stokes equation, is found easy to solve because its BEM kernels are significantly simpler (Section 8.4). A compressible FastStokes solver that solves the simplified linearized compressible Stokes equation using the PFFT-accelerated BEM is developed and tested. This solver is also fast, accurate, and numerically robust.

The compressible FastStokes solver is a full 3-D solver that captures 3-D effects, which is important in the compressible flow case. Simulations of simple geometries using a 3-D finite-element compressible Navier-Stokes equation solver has been reported but shown to be too slow for general 3-D geometries [30]. The frequently-used linearized compressible Reynolds equation has been found to be inaccurate because it neglects important 3-D effects such as the entrance effect. A comparison between the solutions of the compressible FastStokes solver and the analytical solutions of the linearized compressible Reynolds equation clearly shows the inaccuracy of the linearized compressible Reynolds equation (Section 9.2); other researchers have also reported such results [40].

We compare the solutions of the steady incompressible FastStokes solver with the solutions of the compressible FastStokes solver in Section 9.2. When the air compression effect is not strong, the damping force solution of the incompressible FastStokes solver is close to the

solution of the compressible FastStokes solver, although the spring force solution is zero. However, when the compression effect is strong, only the compressible FastStokes solver is capable of predicting both accurate damping forces and accurate spring forces.

The fluid compression effect has attracted a significant amount of attention since a large spring force may shift the resonance frequency of the devices. The strength of the compression effect can be estimated using a non-dimensional number given in Section 8.10; this non-dimensional number is equivalent to the squeeze number in the squeeze film cases. When the strength of air compression is uncertain, we recommend using the compressible FastStokes solver that solves both the damping forces and spring forces although it is a little bit slower than the incompressible FastStokes solver. Several numerical examples are given in Chapter 9. Simulation results show large spring forces when the compression effect is strong. An etch hole example is given in Section 9.4.1; when the squeeze number is large, etching uniformly distributed holes on the structure may dramatically reduce damping and spring forces. A comb drive example is given in Section 9.4.2; as expected, the spring force is negligibly small since the squeeze number is small.

The development of the FastStokes simulation program has made rapid full 3-D fluid analysis of geometrically complicated micro-machined devices possible, as proven by numerical simulation examples, in both incompressible and compressible cases.

Appendix

An analytical flat-panel integration algorithm

1. Set up a local coordinate system

To simplify the calculations, a local Cartesian coordinate system (ξ,η,ζ) is set up and the panel is put in the $\xi-\eta$ coordinate plane with the centroid at the origin. The transformation between the local coordinates (ξ,η,ζ) and the global coordinates (x,y,z) can be easily expressed as

$$\begin{aligned}\begin{bmatrix}\xi\\ \eta\\ \zeta\end{bmatrix}_{3\times1} &= \begin{bmatrix}Coordinate\\ Transformation\\ Matrix\end{bmatrix}_{3\times3}\left\{\begin{bmatrix}x\\ y\\ z\end{bmatrix}_{3\times1} - Centroid\right\};\\ \begin{bmatrix}X\\ Y\\ Z\end{bmatrix}_{3\times1} &= \begin{bmatrix}Coordinate\\ Transformation\\ Matrix\end{bmatrix}_{3\times3}\left\{\begin{bmatrix}X'\\ Y'\\ Z'\end{bmatrix}_{3\times1} - Centroid\right\}\end{aligned} \tag{A.1}$$

where (X,Y,Z) are the local coordinates of the evaluation point and (X',Y',Z') are the corresponding global coordinates.

2. Calculate the solutions of lower-order source and dipole distributions

According to Newman [19], the Gauss-Bonnet theorem can be used to solve the potential at the evaluation point due to constant -4π normal dipole distribution over the flat panel:

$$\Phi = Z\iint \frac{1}{r^3} d\xi d\eta = \sum_{i=1}^{ns} \left\{ \tan^{-1}\left(\frac{\delta\eta_i\left[(X-\xi_i)^2 + Z^2\right] - \delta\xi_i (X-\xi_i)(Y-\eta_i)}{R_i Z \delta\xi_i} \right) \right.$$
$$\left. - \tan^{-1}\left(\frac{\delta\eta_i\left[(X-\xi_{i+1})^2 + Z^2\right] - \delta\xi_i (X-\xi_{i+1})(Y-\eta_{i+1})}{R_{i+1} Z \delta\xi_i} \right) \right\} \tag{A.2}$$

In the above equation, r is the distance between the evaluation point and the point on the panel; R_i is the distance between the evaluation point and the i^{th} panel corner; ξ_i and η_i are the local coordinates of the i^{th} panel corner; $\delta\xi_i = \xi_{i+1} - \xi_i$, $\delta\eta_i = \eta_{i+1} - \eta_i$; and ns is the number of corners. Integrating Φ in the direction of panel normal direction yields Ψ, which is the potential due to -4π source distribution. Furthermore, the potentials due to linear and bilinear source and dipole distributions can be obtained in a similar way.

$$\Psi = \iint \frac{1}{r} d\xi d\eta = \sum_{i=1}^{ns} \left[(X-\xi_i)\sin\theta_i - (Y-\eta_i)\cos\theta_i \right] \log \frac{R_i + R_{i+1} + s_i}{R_i + R_{i+1} - s_i} - Z\Phi \tag{A.3}$$

$$\begin{pmatrix} \Psi_x \\ \Psi_y \end{pmatrix} = \iint \begin{pmatrix} \xi \\ \eta \end{pmatrix} \frac{d\xi d\eta}{r} = \begin{pmatrix} X \\ Y \end{pmatrix} \Psi \mp \sum_{i=1}^{ns} P_i \begin{pmatrix} \sin\theta_i \\ \cos\theta_i \end{pmatrix} \tag{A.4}$$

$$\begin{pmatrix} \Phi_x \\ \Phi_y \end{pmatrix} = Z\iint \begin{pmatrix} \xi \\ \eta \end{pmatrix} \frac{d\xi d\eta}{r^3} = \begin{pmatrix} X \\ Y \end{pmatrix} \Phi \pm Z \sum_{i=1}^{ns} Q_i \begin{pmatrix} \sin\theta_i \\ \cos\theta_i \end{pmatrix} \tag{A.5}$$

$$\Psi_{xy} = X\Psi_y + Y\Psi_x - XY\Psi + \sum_{i=1}^{ns} \cos\theta_i \left[-v_i P_i \sin\theta_i + \frac{1}{3}\left(R_{i+1}^3 - R_i^3\right)\cos\theta_i \right] \tag{A.6}$$

$$\Phi_{xy} = X\Phi_y + Y\Phi_x - XY\Phi + Z\sum_{i=1}^{ns} \cos\theta_i \left[v_i Q_i \sin\theta_i - \left(R_{i+1} - R_i\right)\cos\theta_i \right] \tag{A.7}$$

$$\Phi_{xx} = \Psi + \sum_{i=1}^{ns}\left\{(R_{i+1} - R_i)\cos\theta_i \sin\theta_i + (\xi_i + u_i\cos\theta_i - X)\sin\theta_i \ln\left(\frac{R_{i+1} - U_i}{R_i - u_i}\right)\right\} \quad \text{(A. 8)}$$

$$\Phi_{YY} = \Psi - \sum_{i=1}^{ns}\left\{(R_{i+1} - R_i)\cos\theta_i \sin\theta_i - (\eta_i + u_i\sin\theta_i - Y)\cos\theta_i \ln\left(\frac{R_{i+1} - U_i}{R_i - u_i}\right)\right\} \quad \text{(A. 9)}$$

where $Q_i = \log\dfrac{R_i + R_{i+1} + s_i}{R_i + R_{i+1} - s_i}$, $P_i = \dfrac{1}{2}\left[u_i R_i - U_i R_{i+1} + (R_i^2 - u_i^2)Q_i\right]$, θ_i is the polar angle of the ith edge and s_i is the length of the ith edge. $(u_i,-v_i)$ and $(U_i,-V_i)$ are real and imaginary parts of two 2-D vectors starting from the ith corner and the (i+1)th corner individually; both vectors end at the projection of the evaluation point on the ith edge. The real axis is along the line from the ith corner to the (i+1)th corner, and the imaginary axis is perpendicular to it.

3. Calculate the solutions of higher-order potential distributions

For higher-order potential distribution in the form of

$$\begin{bmatrix}\Psi_{m,n}\\ \Phi_{m,n}\end{bmatrix} = \iint(\xi - X)^m(\eta - Y)^n\begin{bmatrix}r^{-1}\\ r^{-3}\end{bmatrix}d\xi d\eta, \quad \text{(A.10)}$$

three recurrence formulas were given by Newman [19]:

$$\begin{aligned}
\Phi_{m+2,n} &= (m+1)\Psi_{m,n} - \iint\frac{\partial}{\partial\xi}\left[r^{-1}(\xi - X)^{m+1}(\eta - Y)^n\right]d\xi d\eta \\
\Phi_{m,n+2} &= (n+1)\Psi_{m,n} - \iint\frac{\partial}{\partial\eta}\left[r^{-1}(\xi - X)^m(\eta - Y)^{n+1}\right]d\xi d\eta \\
(m+n+1)\Psi_{m,n} + Z^2\Phi_{m,n} &= \iint\left\{\frac{\partial}{\partial\xi}\left[r^{-1}(\xi - X)^{m+1}(\eta - Y)^n\right]\right. \\
&\qquad \left. + \frac{\partial}{\partial\eta}\left[r^{-1}(\xi - X)^m(\eta - Y)^{n+1}\right]\right\}d\xi d\eta
\end{aligned} \quad \text{(A.11)}$$

However, it is quite obvious that the integrals in the recurrence equations are not easy to solve. In this thesis, a second recurrence scheme is introduced to solve these integrals.

First, Green's formula is used to simplify the surface integration to line integration. We define:

$$\begin{aligned}\Omega_y(m,n) &= \iint \frac{\partial}{\partial \xi}\left[r^{-1}(\xi - X)^m(\eta - Y)^n\right] d\xi d\eta = \oint \left[r^{-1}(\xi - X)^m(\eta - Y)^n\right] d\eta \\ \Omega_x(m,n) &= -\iint \frac{\partial}{\partial \eta}\left[r^{-1}(\xi - X)^m(\eta - Y)^n\right] d\xi d\eta = \oint \left[r^{-1}(\xi - X)^m(\eta - Y)^n\right] d\xi\end{aligned} \tag{A.12}$$

and then integrate in a counterclockwise direction along each edge of the panel. The integral $\Omega_y(m,n)$ becomes:

$$\Omega_y(m,n) = -\sum_{i=1}^{ns} \int_{-u_i}^{U_i} \frac{(\xi_i - X + (s+u_i)\cos\theta_i)^m (\eta_i - Y + (s+u_i)\sin\theta_i)^n \sin\theta_i}{r} ds \tag{A.13}$$

where s is a dummy variable along the panel edges; the projection of the evaluation point on the edge is chosen as the origin. Note that the minus sign comes from the integration direction, and the corners of the panel are numbered counterclockwise. ξ_i, X, u_i, θ_i, and so on are all constants once the evaluation point is fixed. The problem remaining is to solve a simpler integral:

$$\Gamma(i,k) = \int_{-u_i}^{-U_i} \frac{s^k}{\sqrt{R_i^2 - u_i^2 + s^2}} ds \tag{A.14}$$

Another recurrence scheme is introduced to solve $\Gamma(i,k)$ at very low computational cost:

$$\Gamma(i,k)=\frac{(-U_i)^{k-1}}{k}R_{i+1}-\frac{(-u_i)^{k-1}}{k}R_i-\frac{(k-1)(R_i^2-U_i^2)}{k}\Gamma(i,k-2)$$
$$\Gamma(i,0)=\ln\left(\frac{R_{i+1}-U_i}{R_i-u_i}\right) \tag{A.15}$$
$$\Gamma(i,1)=R_{i+1}-R_i$$

Define:

$$\Theta(i,k)=\int_{-u_i}^{-U_i}\frac{(s+u_i)^k}{\sqrt{R_i^2-u_i^2+s^2}}ds=\sum_{j=0}^{k}P_k^j\Gamma(i,j)u_i^{k-j}$$
$$P_k^j=\frac{k!}{j!(k-j)!} \tag{A.16}$$

The solution of $\Omega_y(m,n)$ is then expressed as

$$\Omega_y(m,n)=-\sum_{i=0}^{m}\sum_{j=0}^{n}\sum_{k=1}^{ns}P_m^iP_n^j\Theta(k,i+j)\cos^i\theta_k\sin^{j+1}\theta_k(\xi_k-X)^{m-i}(\eta_k-Y)^{n-j} \tag{A.17}$$

and $\Omega_y(m,n)$ is solved in a similar way. Then Newman's recurrence scheme is applied to compute the higher-order Φ's and Ψ's.

4. Transfer local solutions back to the global coordinate system

These solutions given by the recurrence equations are the local solutions that must be transferred back to the global system. Here we offer a general idea of how to transfer solutions back. If the orders of Φ's and Ψ's are very high, the transformation might take more CPU time than computations in the local system. It is suggested that big loops be broken up into

smaller ones in order to save CPU time. Efficiently designed loops might save CPU time by a factor of 20.

The solutions in the local system and global system are:

$$\Phi_{m,n,k}^{\text{local}} = \iint \frac{1}{r^3}(X-\xi)^m (Y-\eta)^n Z^k ds = (-1)^{m+n}\Phi(m,n)Z^k$$
$$\Phi_{m,n,k}^{\text{global}} = \iint \frac{1}{r^3}(X'-x)^m (Y'-y)^n (Z'-z)^k ds \tag{A.18}$$

Using the coordinate transformation equation A1, we have

$$\begin{bmatrix} X'-x \\ Y'-y \\ Z'-z \end{bmatrix}_{3\times 1} = \begin{bmatrix} \textit{Coordinate} \\ \textit{Transformation} \\ \textit{Matrix} \end{bmatrix}^T_{3\times 3} \begin{bmatrix} X-\xi \\ Y-\eta \\ Z-\zeta \end{bmatrix}_{3\times 1} = \begin{bmatrix} c_{11} & c_{12} & c_{13} \\ c_{21} & c_{22} & c_{23} \\ c_{31} & c_{32} & c_{33} \end{bmatrix} \begin{bmatrix} X-\xi \\ Y-\eta \\ Z-\zeta \end{bmatrix}_{3\times 1} \tag{A.19}$$

Then, substituting the above equation into the global expression gives the final solution:

$$\Phi_{m,n,k}^{\text{global}} = \iint \frac{1}{r^3}(X'-x)^m (Y'-y)^n (Z'-z)^k ds$$
$$= \sum_{i1=0}^{m}\sum_{j1=0}^{m-i1}\sum_{i2=0}^{n}\sum_{j2=0}^{n-i2}\sum_{i3=0}^{k}\sum_{j3=0}^{k-i3} c_{11}^{i1}c_{12}^{j1}c_{13}^{m-i1-j1} P_m^{i1} P_{m-i1}^{j1} c_{21}^{i2}c_{22}^{j2}c_{23}^{n-i2-j2} P_n^{i2} P_{n-i2}^{j2} c_{31}^{i3}c_{32}^{j3}c_{33}^{k-i3-j3} \tag{A.20}$$
$$P_k^{i3} P_{k-i3}^{j3} \Phi_{i1+i2+i3,j1+j2+j3,m+n+k-i1-i2-i3-j1-j2-j3}^{\text{local}}$$

For the Stokes kernels, there is a simpler way to transfer the solution back to the global system. We first define:

$$
\begin{aligned}
[\Phi]_1^{\text{local}} &= \begin{bmatrix} \Phi_{1,0,0}^{\text{local}} \\ \Phi_{0,1,0}^{\text{local}} \\ \Phi_{0,0,1}^{\text{local}} \end{bmatrix} \\
[\Phi]_2^{\text{local}} &= \begin{bmatrix} \Phi_{2,0,0}^{\text{local}} & \Phi_{1,1,0}^{\text{local}} & \Phi_{1,0,1}^{\text{local}} \\ \Phi_{1,1,0}^{\text{local}} & \Phi_{0,2,0}^{\text{local}} & \Phi_{0,1,1}^{\text{local}} \\ \Phi_{1,0,1}^{\text{local}} & \Phi_{0,1,1}^{\text{local}} & \Phi_{0,0,2}^{\text{local}} \end{bmatrix} \\
[C] &= \begin{bmatrix} \textit{Coordinate} \\ \textit{Transformation} \\ \textit{Matrix} \end{bmatrix}_{3\times 3}
\end{aligned}
\tag{A. 21}
$$

The solutions in the global system, $[\Phi]_1^{\text{global}}$ and $[\Phi]_2^{\text{global}}$, are defined in a similar way. We then apply the following equation to yield the final solution:

$$
\begin{aligned}
\Phi_{0,0,0}^{\text{global}} &= \Phi_{0,0,0}^{\text{local}} \\
\Psi_{0,0,0}^{\text{global}} &= \Psi_{0,0,0}^{\text{local}} \\
[\Phi]_1^{\text{global}} &= [C]^T [\Phi]_1^{\text{local}} \\
[\Phi]_2^{\text{global}} &= [C]^T [\Phi]_2^{\text{local}} [C]
\end{aligned}
\tag{A. 22}
$$

Bibliography

[1] N. R. Aluru and J. White. A Fast Integral Equation Technique for Analysis of Microflow Sensors Based on Drag Force Calculations. *Proc. of MSM*, pp. 283-286, 1998.

[2] M. Andrews, I. Harris, and G. Turner. A Comparison of Squeeze-film Theory with Measurements on A Microstructure. *Sensors and Actuators A*, Vol. A36, 1993, pp. 79-87.

[3] P. K. Banerjee. *The Boundary Element Methods in Engineering*. McGraw-Hill Book Company, England, 1981.

[4] J. J. Blech. On Isothermal Squeeze Films. *Journal of Lubrication Technology*, Vol. 105, 1983, pp. 615-620.

[5] C.A. Brebbia and J. Dominguez. *Boundary Elements, An Introductory Course*. McGraw-Hill Book Company, New York, 1989.

[6] D. C. Champeney. *Fourier Transforms and Their Physical Applications*. Academic Press, New York, 1973.

[7] Y.-H. Cho, B.-M. Kwak, A. P. Pisano and R. T. Howe. Viscous Energy Dissipation in Laterally Oscillating Planar Microstructures: A Theoretical and Experimental Study. *Proceedings, IEEE. Micro Electro Mechanical Systems. An Investigation of Micro Structures, Sensors, Actuators, Machines and Systems (Cat. No. 93CH3265-6)*. IEEE. pp.93-8. New York, NY, USA, 1993.

[8] L. Greengard. *The Rapid Evaluation of Potential Fields in Particle Systems*. Cambridge, Massachusetts, MIT Press, 1988.

[9] G. S. Gipson. *Boundary Element Fundamentals*. Computational Mechanics Publications, Boston, 1987.

[10] B. J. Hamrock. *Fundamentals of Fluid Film Lubrication*. McGraw-Hill Inc., New York, 1994.

[11] J. L. Hess and A.M.O. Smith. Calculation of Non-Lifting Potential Flow about Arbitrary Three-Dimensional Bodies. *J. Ship Res.* Vol. 8, No. 2, pp. 22-44, 1962.

[12] E. Kim, Y. Cho, and M. Kim. Effect of Holes and Edges on the Squeeze Film Damping of Perforated Micromechanical Structures. *Proc. of MSM*, Puerto Rico, 1999, pp. 296-301.

[13] S. Kim and S.J. Karrila. *Microhydrodynamics: Principles and Selected Applications*. Butterworth-Heinemann, Boston, 1991.

[14] O. A. Ladyzhenskaya. *The Mathematical Theory of Viscous Incompressible Flow*. Gordon and Breach, New York, 1963.

[15] O. A. Ladyzhenskaya. *The Boundary Value Problems of Mathematical Physics*. Springer-Verlag, New York, 1985.

[16] N. Maluf. *An Introduction to Microelectromechanical Systems Engineering*. Artech House, Boston, 2000.

[17] A. F. Mills. *Heat Transfer, second edition*. Prentice-Hall Inc., Upper Saddle River, New Jersey, 1999.

[18] K. Nabors and J. White. Fastcap: A Multipole Accelerated 3-D Capacitance Extraction Program. *IEEE Transactions on CAD of Integrated Circuits and Systems*, Vol. 10, pp. 1447-1459, Nov. 1991.

[19] J. N. Newman. Distribution of Sources and Normal Dipoles over a Quadrilateral Panel. *Journal of Engineering Mathematics*, Vol. 20, pp. 113-126,1986.

[20] J. A. Owczarek. *Fundamentals of Gas Dynamics*. International Textbook Company, Scranton, Pennsylvania, 1964.

[21] J. R. Phillips and J. K. White. A Precorrected-FFT Method for Electrostatic Analysis of Complicated 3-D Structures. *IEEE Trans. on Computer-Aided Design*, October 1997, Vol. 16, No. 10, pp. 1059-1072.

[22] J. R. Phillips. *Rapid Solution of Potential Integral Equations in Complicated 3-D Geometries*. Ph.D. Thesis, MIT, June 1997.

[23] C. Pozrikidis. A Study of Linearized Oscillatory Flow past Particles by the Boundary-integral Method. *J. Fluid Mech.*, vol. 202, pp. 17-41, 1989

[24] C. Pozrikidis. *Boundary Integral and Singularity Methods for Linearized Viscous Flow*. Cambridge University Press, Cambridge, 1992.

[25] C. Pozrikidis. *Introduction to Theoretical and Computational Fluid Dynamics*. Oxford University Press, New York, 1997.

[26] D. Ramaswamy, W. Ye, X. Wang, and J. White. Fast Algorithms for 3-D Simulation. *Journal of Modeling and Simulation of Microsystems*, Vol. 1, pp. 77-82, 1999.

[27] D. Ramaswamy. *Simulation Tools for MicroElectroMechanical System*. Ph.D. Thesis, MIT, May 2001.

[28] V. Rokhlin. Rapid Solution of Integral Equations of Scattering Theory in Two Dimensions. *J. Comp. Phys.*, 86:414--439, 1990.

[29] Y. Saad and M. Schultz. GMRES: A Generalized Minimal Residual Algorithm for Solving Nonsymmetric Linear Systems. *SIAM J. Sci. Statist. Comput.*, 7 (1986), pp. 856-869.

[30] G. Schrag, P. Voigt and G. Wachutka. Physically-based Modeling of Squeeze Film Damping by Mixed Level System Simulation. *Proc. of Transducers*, Munich, Germany, June, 2001, pp. 670-673.

[31] S. D. Senturia, R. M. Harris, B. P. Johnson, S. Kim, K. Nabors, M. A. Shulman, and J. K. White. A Computer-Aided Design System for Microelectromechanical Systems. *J. MEMS*, Vol. 1, No. 1, March 1992.

[32] S. D. Senturia, N. Aluru, and J. White. Simulating the Behavior of MEMS Devices: Computational Methods and Needs. *IEEE Computational Science & Engineering*, vol. 4, no. 1, 1997.

[33] S. D. Senturia. *Microsystem Design.* Kluwer Academic Publishers, Boston, 2001.

[34] W. C. Tang, T.-C. H. Nguyen, M. W. Judy and R. T. Howe. Electrostatic-comb Drive of Lateral Polysilicon Resonators. *Sensor and Actuators*, Vol. A21-23, pp. 328-381, 1990.

[35] W. C. Tang, M. G. Lim and R. T. Howe. Electrostatic Comb Drive Levitation and Control Method. *Journal of MEMS*, vol. 1, No. 4, pp. 170 - 177, 1992.

[36] L.N. Trefethen and D. Bau. *Numerical Linear Algebra.* Society for Industrial and Applied Mathematics, Philadelphia, 1997.

[37] T. Veijola, H. Kuisma, J. Lahdenpera, and T. Ryhanen. Equivalent-circuit Model of the Squeezed Gas Film in a Silicon Accelerometer. *Sensors and Actuators*, A 48, 1995, pp 239-248.

[38] T. Veijola, H. Kuisma, and J. Lahdenpera. Model for Gas Film Damping in a Silicon Accelerometer. *Proceedings of Transducers'97*, Chicago, June 1997, pp. 1097-1100.

[39] T. Veijola and T. Mattila. Compact Squeezed-Film Damping Model for Perforated Surface. *Proceedings of Transducers 2001*, Munchen, June 10-14, 2001, pp. 1506-1509.

[40] T. Veijola, K. Ruokonen, and I. Tittonen. Compact Squeezed-Film Damping Model Including the Open Border Effects. *Proceedings of MSM 2001*, Hilton Head March 19-21, 2001, pp 76-79.

[41] T. Veijola. Compact Damping Models for Lateral Structures Including Gas Rarefaction Effects. *Proceedings of MSM 2000*, San Diego, March 27-29, pp. 162-165, 2000.

[42] T. Veijola and M. Turowski. Compact Damping Models for Laterally Moving Microstructures with the Gas-Rarefaction Effects. *Journal of Microelectromechanical Systems*, vol. 10, no. 2, June 2001, pp. 263-272.

[43] X. Wang, J.N. Newman, and J. White. Robust Algorithms for Boundary-Element Integrals on Curved Surfaces. *Proc. MSM*, San Diego, CA, 2000, pp. 473-476.

[44] X. Wang, P. Mucha and J. White. Fast Fluid Analysis for Multibody Micromachined Devices. *Proc. MSM*, Hilton Head Island, SC, 2001, pp. 19-22.

[45] X. Wang and J. White. Analyzing Fluid Compression Effect in Complicated Micro-machined Devices. *Proc. of Transducers*, pp. 666-669, 2001.

[46] X. Wang, M. Judy, and J. White. Validating Fast Simulations of Air Damping in Micromachined Devices. *Proc. MEMS 2002*, Las Vegas, NV, Jan. 2002. pp 210~213.

[47] Y.-J. Yang and S.D. Senturia. Numerical Simulation of Compressible Squeezed-Film Damping. *Tech. Digest, Solid State Sensor and Actuator Workshop*, Hilton Head Island, SC, June 1996, pp. 76-79.

[48] Y.-J. Yang. Squeeze-Film Damping for MEMS Structures. M.S. Thesis, MIT, January 1998.

[49] W. Ye, J. Kanapka, X. Wang, and J. White. Efficiency and Accuracy Improvements for FastStokes. A Precorrected-FFT Accelerated 3-D Stokes Solver, *Proc. of MSM*, Puerto Rico, 1999, pp. 502-505.

[50] W. Ye, X. Wang, Werner Hemmert, D. Freeman and J. White. Viscous Drag on A Lateral Micro-Resonator: Fast 3-D Fluid Simulation and Measured Data. *Tech. Digest, Solid-State Sensor and Actuator Workshop*, Hilton Head Island, SC, June 2000, pp 124-127.