Modeling and Simulation of Oil Transport for Studying Piston Deposit Formation in IC Engines

by

Sean McGrogan

B.S., Mechanical Engineering
Lehigh University, 2005

Submitted to the Department of Mechanical Engineering in Partial Fulfillment of the
Requirements of the Degree of

Master of Science in Mechanical Engineering

at the

Massachusetts Institute of Technology

June 2007

Signature of Author: _____
Department of Mechanical Engineering
May 21, 2007

Certified by: _____
Tian Tian
Lecturer of Mechanical Engineering
Thesis Supervisor

Certified by: _____
Victor Wong
Lecturer of Mechanical Engineering
Thesis Supervisor

Accepted by: _____
Professor Lallit Anand
Chairman, Department Committee on Graduate Studies
Department of Mechanical Engineering

This page was intentionally left blank.

Modeling and Simulation of Oil Transport for Studying Piston Deposit Formation in IC Engines

by

Sean McGrogan

Submitted to the Department of Mechanical Engineering on May 21, 2007 in Partial Fulfillment of the Requirements of the Degree of Master of Science in Mechanical Engineering

ABSTRACT

Carbonaceous deposits have long plagued the internal combustion engine, yet a fundamental comprehension of their underlying causes remains to be developed. In particular, *piston land* deposits can bring about an array of problems; for example, once a thickness threshold is crossed, the engine's reliability is threatened by an elevated possibility of seizure. As tightening emissions regulations continue to place more stringent constraints on power cylinder design, control of piston deposits, specifically in the top land and top ring groove, is becoming ever more difficult.

Tests run on a heavy duty diesel engine revealed the piston land carbon deposit distribution to be circumferentially nonuniform, and a theoretical inquiry was invoked to investigate the cause. Since these deposits are typically lubricant derived, a three-dimensional, unsteady model of the oil film attached to a piston land was formulated. Focus was placed on the top land, in order to explore the effects of both reciprocating inertia and combustion-driven gas flows on the film's motion and thickness distribution. The numerical simulation created uses results from a realistic CFD simulation of the combustion process as input data.

It was found that the gas velocities can have a profound effect. The gases create interesting wave structures on the free surface of the oil film, significantly altering the film thickness distribution. A new mechanism governing oil transport was discovered. Clever usage of this mechanism could substantially reduce the amount of oil, and hence the amount of deposit, on the top land. The simulation shows potential for application not only to the study of deposit formation, but also to that of oil consumption.

Thesis Supervisors:
Dr. Tian Tian (Lecturer of Mechanical Engineering)
Dr. Victor Wong (Lecturer of Mechanical Engineering)

This page was intentionally left blank.

ACKNOWLEDGMENTS

This page was intentionally left blank.

Contents

4 Numerical Approach   53

List of Figures

13

List of Tables

1 Introduction

The process by which one set of materials adheres to a surface composed of a different set of materials is an everyday phenomenon. Gum sticks to sidewalks. Soot from vehicles clings to windows. Dishes become dirty. Arteries become clogged. The internal combustion (IC) engine is no exception; degradation products of the various engineered fluids eventually form deposits, diminishing the engine's performance and reducing its life.

Deposits are found in many locations within an IC engine. For example, the metal surfaces of pistons, liners, fuel injectors, and valve seats of most engines commonly become accumulation sites for a variety of solidified compounds. As one would think, deposits are usually (if not always) *un*desirable.

1.1 Motivation

Deposits (or "carbon deposits") degrade an engine's overall performance. Their presence is detrimental for both spark ignition and compression ignition engines, but for this project, CI (diesel) engines were chosen as the focus application.

One key area of the diesel engine which has been receiving a considerable amount of attention in recent years is the piston's upper ringpack region, specifically the top land and top ring groove. An overall schematic of the power cylinder and piston ringpack, adapted from [22], is shown in Figure 1.
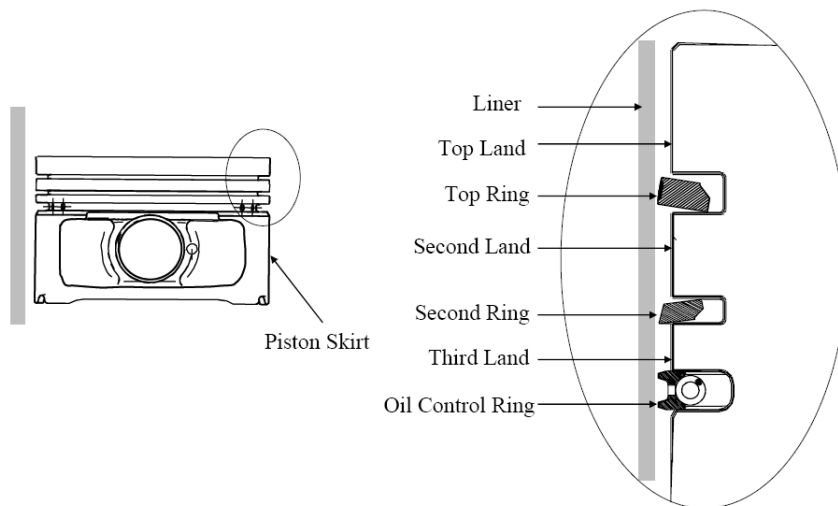
Figure 1: Schematic of power cylinder with piston ringpack detail, from [22].

Figure 2: Piston with a substantial carbon deposit problem.

Severe consequences can arise from deposits which become lodged on the piston top land and within the top ring groove. Once the deposits build up enough, loss of oil consumption control typically occurs; the ringpack can no longer minimize oil consumption and the amount of lubricant entering the combustion chamber increases substantially. Fully formulated lubricants contain quite an array of chemicals (in large part due to the additive package) and do not burn as cleanly as fuel, so it is no surprise that an increased oil consumption rate goes hand in hand with poor engine-out emissions characteristics. Perhaps worst of all, once the deposit becomes thick enough, friction between the piston and liner can become so large that the power cylinder seizes, causing premature, catasrophic engine failure. Dozens of pictures of unidentified pistons having quite a bit of carbon deposit are readily found on the internet; one such piston is displayed in Figure 2 as an example.

The underlying physical and chemical mechanisms governing carbon buildup are poorly understood. However, a large quantity of research has been performed to attempt to understand these mechanisms, with eliminating deposits completely (without creating other problems) being the ultimate goal.

Literature The vast majority of published work on carbon deposits takes an experimental approach. Studies range from fundamental (e.g. [65, 66, 67]) to applied (e.g. [63, 64]). Various experimental techniques are invoked in these investigations, including Differential Scanning Calorimetry [68], Thermo-Gravimetric Analysis, Fourier Transform Infrared Spectroscopy [69], and electron microscopy [66], as well as countless custom test apparatuses. The literature involving theoretical study applied to this problem is comparitively scarce[1]. Some of the notable work includes [71], [72], [74], and that of the group at Penn State University, e.g. [70].

The published analyses shed quite a bit of light on the nature of deposits, but to some extent their conclusions contradict each other. The problem is complex and has a large number of

---

[1]Probably because the kinetics governing the interaction of the thousands of chemical species in lubricating oil, in the presence of combustion gases, is not well understood.

dimensions; as of yet workers have not been able to collapse the space of the problem to some fundamental, reduced basis. Of those aspects of the problem which are understood, one is that for the *piston land* deposits studied in this work, the lubricant is consistently found to be the culprit [61, 62]. Section 2 discusses the current understanding of deposits in more detail.

This Project   As the fundamentals behind piston deposits are not well understood, and so much of the published work is experimental, a theoretical approach was taken in this work.

Observations of the top land of several pistons taken from an extensively tested modern diesel engine were made. As the observed carbon deposit distribution was nonuniform (in a consistent manner), these pistons seemed to indicate that oil was not present on certain parts of the top land during operation. This feature strongly suggested that *combustion gases* were entering the top land crevice. To investigate this possibility, CFD (Computational Fluid Dynamics) simulations of the combustion process were carried out. As was done in [71], the top land crevice was included in the computational domain. The results of this effort spurred questions regarding the effects of these gases on the oil film (since, again, oil is believed to be the source of piston deposits in diesel engines). As the primary focus area was the top land, a simulation tool, the "Top Land Oil Movement Model" (TLOMM), was developed to better quantify these effects[2].

The central question this work set out to answer was "are the crevice gas velocities sufficient to push oil off of the top land?" That is, the simulation developed seeks to explain the nonuniform distribution of carbon deposits (and hence lubricant) on the top land[3].

One may note that the work undertaken shares some similarity with that performed in [71], in which crevice gas flows' effects on lubricant vaporization and piston deposits were examined. A key difference is that in [71], the motion of the oil film was neglected and some film thickness was simply assumed to be present, whereas in this work the focus was to quantify the motion and distribution of the film in detail (while neglecting vaporization). A combination of the two studies, in which neither oil motion nor vaporization was neglected, would yield interesting predictions.

1.2   Objectives

The main objective of this work was:

---

[2]The model focuses on application to four-stroke engines, but could be applied to two-stroke engines as well, with some very minor modifications.

[3]Of course, other oil removal mechanisms, such as locally accelerated evaporation due to the very high temperature of combustion gases, would also require investigation if the answer to this central question turned out to be "No."

To create a simulation tool which makes accurate, detailed predictions of the lubricant distribution on the top land, capturing the effects (if any) of crevice gas flows.

Of course the crevice gas flow data used by the TLOMM would be provided by some external combustion CFD simulation, carried out either at MIT or by the project sponsors.

The end simulation also had to meet a few additional objectives. In particular,

- The finished program had to be fast enough that it could be deployed to run on standard desktop workstations without requiring an unreasonable amount of computing time. The project scale was not large enough to justify allowing for the usage of parallel processing (nor was it expected to be a necessity).

- The simulation had to be able to run for hundreds of engine cycles. The reason for this criterion has to do with the expected timescale of the problem. Engine tests indicated that significant carbon buildup takes place on the order of a few minutes (thousands of cycles), not on the order of milliseconds (one engine cycle). This second sub-objective carried with it an immediate requirement: any cumulative numerical errors needed to be small (negligible) by the end of the simulation. In other words, the steady state solution would be useless if it was completely dominated by errors inherent to the numerical algorithm which produced the results. Of course this objective competes with the first bullet, which seeks to minimize computing time.

- TLOMM needed to be developed with the expectation that future workers will extend it to include additional physics. Any algorithms it used had to be flexible enough that they did not prohibit future expansion.

1.3   Scope

Study of the fluid mechanics of the film is within the scope of this project. To determine the effect of the gas flows, the simulation would have to be three dimensional and unsteady. Calculation of the gas flows themselves would be very limited in this work, as CFD data of the top land crevice was readily available. Inclusion of thermal and chemical degradation mechanisms were outside the scope of this work (though their importance is certainly acknowledged). Allowing the top land oil film to come into direct contact with the liner was also outside the scope, as was performing involved simulations of the mechanisms supplying oil to the top land. Inclusion of detailed calculations of the thermal environment was outside the scope, but thermal effects were accounted for in an average sense. Essentially, the project scope was mostly confined to Newtonian fluid mechanics applied to only the oil on the top land.

1.4 About this document

This thesis was prepared using L$_Y$X, an open source word processor which typesets documents using L^{A}T$_E$X.

Accessibility According to MIT's current policies, it sounds like this document will be available online in perpetuity. We suppose this work will not be very relevant once IC engines have been replaced by tabletop fusion reactors, but at any rate, the URL is `http://dspace.mit.edu/`. Some of the figures in this document would be very difficult to comprehend in black and white; the file at this web address is in color.

Audience This document was attempted to be written on a level that should be understandable to most engineers and scientists who are familiar with fluid mechanics and IC engines but have little to no experience in numerical methods. To this end, Section 4 tries to augment its mathematical presentations with qualitative and graphical demonstrations of the various issues. This practice may be too basic for those fluent in numerical analysis; these readers are invited to skip that section.

Modularity An effort was made to make this document at least somewhat modular. If one is only concerned with results, Sections 2, 4, and 5[4] may be skipped, but Section 3 should still be read[5].

---

[4]Except 5.3.1.

[5]The weaknesses of the underlying model should always be understood before attempting to interpret simulation results.

This page was intentionally left blank.

2 Background

This section presents some basic background information on carbon deposits. As the modeling efforts pertaining to the simulation developed in this work did not include chemical phenomena, this material is far from comprehensive.

Conventional Lubricants   Lubricants used in diesel engines are typically composed of 75-83% base oil, 5-8% viscosity modifier, and 12-18% additives [76]. The additive package is a mixture of highly specialized components, each of which serving some specific purpose. For example, ZDDP (Zinc Dialkyl Dithio Phosphate) is mainly an antiwear component. Detergents and dispersants attempt to keep oil-insoluble combustion products suspended in the lubricant, rather than allowing these products to migrate to solid surfaces. Antioxidants prevent the base oil from oxidizing. Entire books about lubricating oil additives are available and need not be repeated here.

2.1 Piston Thermal Environment

The top land environment is quite hot. [62] shows some representative temperatures at different locations on a conventional diesel engine piston; the top land temperature depicted is 349° C. Of course, the top land *oil* temperature cannot be summed up in just one number; one can envision that when hot combustion gases enter the top land crevice, they would force the oil temperature at the oil/gas interface to be higher than at the piston surface.

To address the issue, some heat transfer calculations were carried out in this project, since the oil temperature is required in order to evaluate material properties and speculate about possible reactions[1]. Various film thicknesses were assumed, as well as hypothetical deposit thicknesses. Studies of deposits' thermal characteristics (e.g. conductivity) report somewhat inconsistent results, but the values from [73] were taken to be representative. Calculations performed indicated that the oil temperature fluctuates a substantial amount ($\sim 40°$ C) within a cycle, and certainly has a large variation in the radial direction. To complicate matters, the results depend strongly on the assumed oil film thickness and deposit thickness, as well as engine load. There is no single number representing the oil temperature, as it varies spatially and temporally. However, for the purposes of this project, a constant, uniform oil temperature of 325° C was used[2]. It is believed that, for the engine on which this project focuses, this value represents the top land oil temperature in an average sense.

---

[1]Credit is due to another student, Raul Coral, who performed most of these calculations.

[2]The TLOMM simulation can readily use some given temperature distribution when calculating the material properties, if such data is available.

2.2 Lubricant Degradation

Deposition occurs after the lubricant undergoes a degradation process. On the top land of an operating engine, the thermal and chemical environments are harsh. Temperatures are high and combustion gases containing acids and soot (for example) may be present.

In the upper ringpack region of a diesel engine piston, where oil residence times are high, the additive package eventually becomes depleted and can no longer prevent deposition. There is a co-existence of many physical mechanisms acting on the oil, such as vaporization, oxidation, pyrolysis, polarization, polymerization, and the forces considered in this project (gas shear and inertia), to name a few. Somewhat separate mechanisms act on the deposit as well; it too may be oxidized (usually at higher temperatures than oil oxidation), thermally degraded, and mechanically scraped away, for example. These processes take place on disparate length scales. Some of them are catalyzed by the metallic piston surface. All of them combine to create a complex competition between deposit formation and breakdown.

Deposit formation pathway[3] The general consensus is that oil oxidation is a key degradation mechanism which strongly correlates with deposits [63, 75]. It is believed [77, 78, 79, 80] that at the high temperatures found in the upper ringpack, metal catalyzed oxidation of the oil's hydrocarbons occurs. Some of the products are peroxides, which decompose into highly reactive radicals, which then attack the base oil's unreacted hydrocarbons[4]. Some products polymerize into high molecular weight compounds. Dispersants and detergents attempt to solubilize these polymers; however, it has been proposed [70] that once these heavy compounds reach the lubricant's solubility limit, they drop out as deposit.

Oil oxidation rate The rate at which oil oxidation proceeds has been reported to be first order. Some studies have found the rate constant to be reasonably well approximated by the Arrhenius relation, $k = Ae^{\frac{-E_a}{RT}}$. Numerical values for the variables in this equation, for a limited range of conditions, are available in [70, 75]. Of course, much of the difficulty in modeling degradation processes, such as oxidation, lies in predicting the rate constant k.

The oil oxidation rate constant depends on many factors; quantitative understanding of these dependencies is not well developed, making theoretical modeling of the deposit process difficult. Some of the key factors governing the oil oxidation rate are summarized as follows.

---

[3]The rest of this section borrows heavily from the final report submitted by Amanda Shing, an undergraduate researcher in the Sloan Automotive Lab at MIT.

[4]Yes, this implies that the oxidation process accelerates itself.

- Temperature

  The oxidation rate observed experimentally increases as temperature increases [75, 82], consistent with the Arrhenius relation.

- Solid surface

  The type of metal governs the degree to which the oxidation reaction is catalyzed. [78] reported that an iron surface gives rise to more deposit than an aluminum surface[5]. [81] found that deposit does not form on glass. It has been suggested [78] that transition metals catalyze the deposit process (and hence oil oxidation, one would think) while aluminum inhibits its formation.

- Concentration of dissolved oxygen

  The oxidation rate is limited by the availability of oxygen [70].

- Oil type

  It was reported in [77] that ester derived base stocks, such as some synthetics, may have inherent antioxidative properties.

- Antioxidant additive

  The antioxidant can inhibit the process of oil oxidation by either preventing the initial formation of radicals, or by reacting with the radicals to stop their propagation [79, 80]. Antioxidants are reportedly consumed following a first order rate [79, 82]. It is believed that the base oil will not oxidize until all antioxidants have been depleted [79, 80].

- Other non-hydrocarbons

  It has been reported [83] that water inhibits the oxidation rate. Additionally, copper is a natural antioxidant [84].

Obviously lubricant degradation is a complex phenomenon. The present discussion is far from being a complete survey of the pertinent knowledge base. Its main purpose was not to break any new ground; rather, it meant to highlight that the work performed in this project, though illustrative, does not tell the whole story when it comes to predicting carbon deposition.

---

[5]One should take this statement with a full set of disclaimers; the nonlinearity of the problem means that blanket statements are usually inaccurate, and observations are only valid for the set of conditions tested.

This page was intentionally left blank.

3 Modeling Approach

Development of the Top Land Oil Movement Model (TLOMM) involves several steps. A classical fluid mechanics analysis is carried out. The lubricant is assumed to be incompressible, and the flow locally parallel. Velocity profiles are obtained[1]. A mass conservation constraint is applied to obtain the primary governing equation for this project. As discussed in Section 3.6.1, the lubricant is treated as being Newtonian (i.e., the viscosity is assumed not to depend on shear strain or shear rate), and the material properties are not allowed to vary spatially or temporally. The model developed is a direct extension of the work found in [3], in that the domain is expanded to include the circumferential direction, and the capability to model the effects of driving forces arising from gas flows within the top land crevice is incorporated.

3.1 Problem Description

An exhaustive amount of experimental and theoretical work studying the transport of oil on various areas of the piston was performed by Thirouard, using a Laser Induced Flourescence (LIF) setup, as detailed in [3]. It was observed in these tests that the lubricant is flung back and forth, along the axial direction, in phase with the piston acceleration. From these measurements, a characteristic oil film thickness (roughly speaking) typically found on piston lands is on the order of 20 microns or so. The exact values depend on many parameters, including engine speed, oil viscosity, axial height of the land, etc. For comparison, the width of a human hair is typically around 80 μm [12].

Effects of gas flows on the oil film were also observed experimentally in [3]. As such, one of the key inputs used by the TLOMM is a full set of spatially and temporally resolved gas velocity data within the top land crevice. This data will typically come from a CFD simulation of the combustion process. The data is not required, but there would be little reason for running the TLOMM simulation without any gas flows, as it would defeat the purpose extending of the work in [3]. A pictorial representation of the system being studied is displayed in Figure 3.

Geometry The fuel injectors found in modern diesel engines typically have several holes; each hole creates a spray of fuel during combustion. An engine with five or six fuel sprays within each combustion chamber is not uncommon. Most, if not all, engine manufacturers perform CFD analyses of their combustion processes. These combustion simulations typically involve complex chemical kinetics and some form of a turbulence model, in addition to the standard set of (spatially and temporally resolved) variables such as density, temperature, pressure, velocity, etc.

---

[1] The effect of gas flows is factored into the oil velocity profiles.
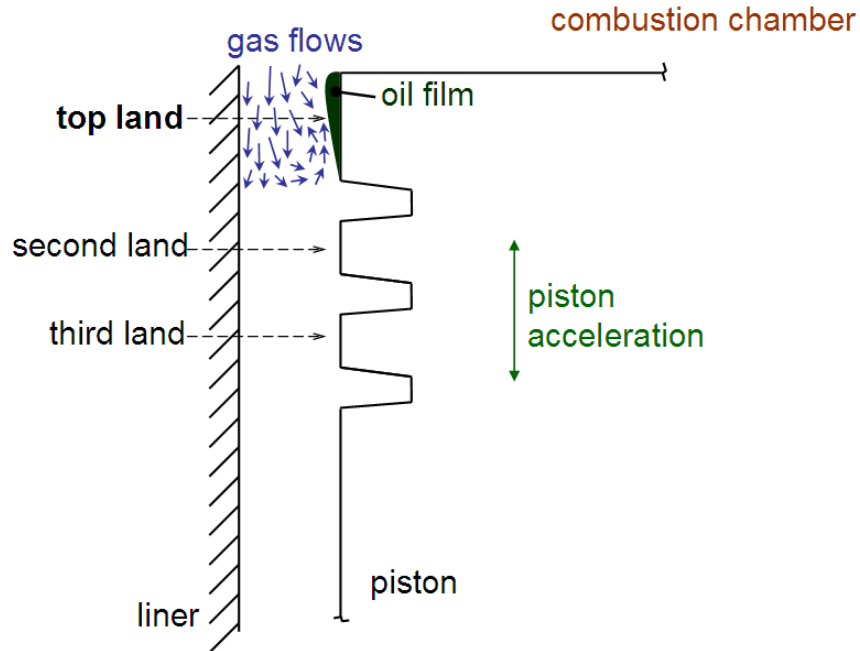
Figure 3: Schematic side view of a piston, showing the top land oil film and its driving forces. Piston rings not pictured, clearance is exaggerated.

As these simulations can be computationally expensive, manufacturers often assume that each fuel spray is the same, and just simulate one spray area (e.g. one fifth of the combustion chamber). This assumption is especially reasonable in engines which have combustion chambers that do not set up a swirling flow[2]. The TLOMM was developed knowing that its initial applications would be to diesel engines, so it too simulates just a portion of the piston. What fraction of the piston it actually simulates depends entirely on the CFD input data; hence, if the input data does span the full circumference of the top land crevice, the model automatically does so as well. The curvature of the piston is neglected in the model, because the radius of curvature of the piston is very large compared to the film thickness. This approximation is identical to one's everyday perception that the earth is flat; a human's height is very small compared to the radius of the earth.

Figure 4 shows a schematic of the coordinate system used by the model. The nomenclature of the coordinate directions and velocities follows the standard naming convention and is summarized in Table 1. The characteristic length scales in the x, y, and z directions represent the top land axial height, one fifth of the circumference of the top land, and an estimate of the oil film thickness, respectively. These top land dimensions represent a typical piston in a diesel engine having 2 liters of displacement per cylinder.

---

[2]At the same time, this assumption still oversimplifies the situation to some extent; the location of the ring gap plays a major role in the gas flows [3].
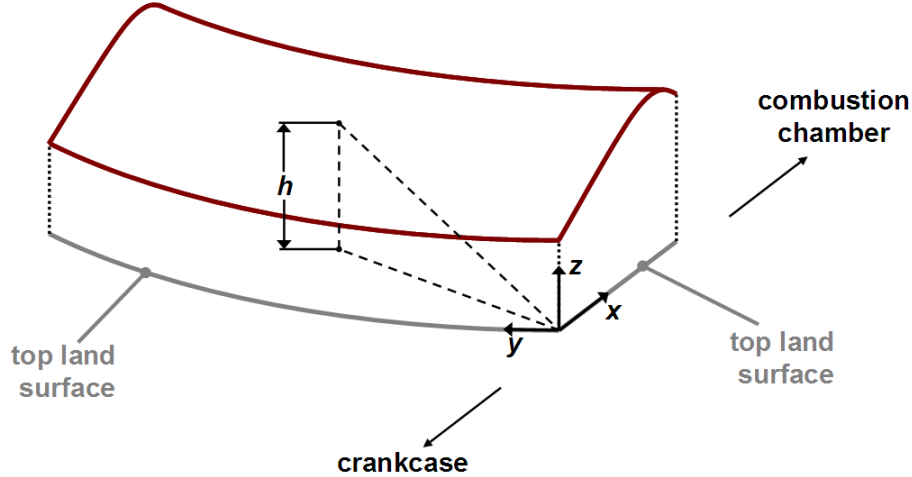
Figure 4: Isometric view of the TLOMM coordinate system, with some arbitrary film thickness distribution.

	axial	circumferential	radial
direction	x	y	z
velocity	u	v	w
characteristic length scale	10mm	80mm	$20\mu m$

Table 1: Direction and velocity nomenclature.

3.2   Model Derivation

The governing equation is the well known Navier-Stokes equation of incompressible, continuum fluid dynamics, which represents the physical requirement that momentum is conserved:

$$\frac{D\vec{v}}{Dt} = -\frac{1}{\rho}\vec{\nabla}p + \nu\nabla^2\vec{v} \tag{1}$$

where $\vec{v}$ represents the local velocity vector. Recall that velocity is defined here with respect to an inertial (i.e. non-accelerating) reference frame, taken in this work to be the cylinder liner.

In its unsimplified form, equation (1) is notoriously difficult to solve. In fact, as of the publication date of this thesis, a million dollar prize is offered to anyone who can prove or disprove existence of solutions to this equation (see [11]). Hence most analysis of (1) is performed on one of many simplified versions.

It is desirable to have all calculations in the reference frame of the piston. As the piston (and hence the reference frame itself) accelerates throughout an engine cycle, the reference frame is non-inertial. Since

$$\vec{v}_{oil,\,w.r.t.\,liner} = \vec{v}_{piston,\,w.r.t.\,liner} + \vec{v}_{oil,\,w.r.t.\,piston} \tag{2}$$

27
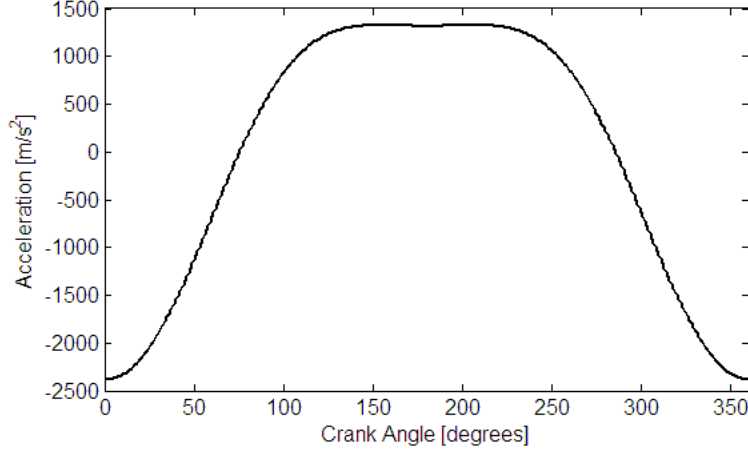
Figure 5: Piston acceleration for a typical engine at 1500 rpm.

and $\frac{d\vec{v}}{dt} = \vec{a}_p$, the piston acceleration $\vec{a}_p$ is manifested as a body force. Substituting (2) into (1) yields

$$\frac{D\vec{v}_{o/p}}{Dt} = -\vec{a}_p - \frac{1}{\rho}\vec{\nabla}p + \nu\nabla^2\vec{v}_{o/p} \tag{3}$$

where now the oil velocities are in the frame of the piston[3]. The piston acceleration for a representative engine operating at 1500 rpm is presented in Figure 5.

The classical "lubrication assumption" that the flow velocity in the z direction may be neglected is invoked. This approximation is justified because of the scales involved; namely, $\frac{h}{L} \ll 1$. Characteristic values of h and L are $20\,\mu$m and $10\,$mm, respectively. Applying this simplification and expanding the left hand side of (3), one obtains

$$\overbrace{\frac{\partial u}{\partial t}}^{a} + \overbrace{u\frac{\partial u}{\partial x}}^{b} + \overbrace{v\frac{\partial u}{\partial y}}^{c} \;=\; -\overbrace{a_p}^{d} - \overbrace{\frac{1}{\rho}\frac{\partial p}{\partial x}}^{e} + \nu\left(\overbrace{\frac{\partial^2 u}{\partial x^2}}^{f} + \overbrace{\frac{\partial^2 u}{\partial y^2}}^{g} + \overbrace{\frac{\partial^2 u}{\partial z^2}}^{h}\right) \tag{4}$$

$$\frac{\partial v}{\partial t} + u\frac{\partial v}{\partial x} + v\frac{\partial v}{\partial y} \;=\; -\frac{1}{\rho}\frac{\partial p}{\partial y} + \nu\left(\frac{\partial^2 v}{\partial x^2} + \frac{\partial^2 v}{\partial y^2} + \frac{\partial^2 v}{\partial z^2}\right) \tag{5}$$

A scaling analysis is required in order to determine which of the remaining terms may be safely neglected. Each term of the first equation has been labeled for easy reference in the following section.

---

[3]From now on, the o/p subscripts are dropped

28

3.2.1 Scaling Analysis

First, the pressure gradient $\vec{\nabla}p$ can be estimated as follows. From experiments, it is known that the film thickness is very small compared to the dimensions of any of the piston lands. Hence, the oil can be classified as a "thin film," and treated in a manner identical to a boundary layer. The film can only support a negligible pressure gradient in the z direction, so the pressure within the film can be regarded as being constant along this direction. Moreover, since the oil and gas pressures must be identical at the film's interface (and since surface tension ends up being neglected, to be discussed), the pressure at any point in the film must be equal to the local gas pressure at the interface. The pressure distribution within the film is imposed directly by the pressure distribution in the gas adjacent to the film. Hence, any pressure gradient within the film in the x or y directions can only be due to a gas pressure gradient in these directions. A rough estimation of this pressure gradient can be made. Using the ringpack gas dynamics simulation developed in [2], a ballpark figure for the maximum value (within a cycle) of the difference in gas pressures between the combustion chamber and the top ring groove is about 0.01 bar. Hence the axial pressure gradient within the gas, and the oil as well, is on the order of $\frac{1000\,\mathrm{Pa}}{.01\,\mathrm{m}}$, or 100,000 $\frac{\mathrm{N}}{\mathrm{m}^3}$. The circumferential pressure gradient depends on the location of the ring gap; since this simulation does not account for this feature, $\frac{\partial p}{\partial y}$ is assumed to be zero.

To carry out the scaling analysis, an estimate of the oil film's axial velocity is needed. The velocity profile for the fully viscous case is used to calculate this estimate. If one was to drop all terms in equation (4) except $-a_p$ and $\nu\frac{\partial^2 u}{\partial x^2}$, and integrate the resulting ODE, the axial velocity profile obtained would be

$$u = \frac{a_p}{\nu}\left(\frac{1}{2}z^2 - hz\right) \tag{6}$$

The average value, $\bar{u}$, where $\bar{u} = \frac{1}{h}\int_0^h u\,dz$, is

$$\bar{u} = -\frac{a_p h^2}{3\nu} \tag{7}$$

According to Figure 5, the instantaneous value of $\bar{u}$ is obviously dependent on the piston position within a cycle.

The terms in equation (4) may now be compared against each other by forming ratios. Overbars are used to denote characteristic scales (e.g. $\bar{x}$ for axial length scale)[4].

$$\frac{a}{h} \approx \frac{\bar{u}/\bar{t}}{\nu\bar{u}/\bar{z}^2} = \frac{\bar{z}^2}{\nu\bar{t}} \tag{8}$$

---

[4]The term labelled "h" is chosen as the denominator out of convenience, due to the expectation that it will probably be the dominant term. However, it makes no formal difference which of the terms chosen.

$$\frac{b}{h} \approx \frac{\bar{u}^2/\bar{x}}{\nu\bar{u}/\bar{z}^2} = \frac{\bar{u}\bar{z}}{\nu}\left(\frac{\bar{z}}{\bar{x}}\right) \tag{9}$$

$$\frac{c}{h} \approx \frac{\bar{v}\bar{u}/\bar{y}}{\nu\bar{u}/\bar{z}^2} = \frac{\bar{v}\bar{z}}{\nu}\left(\frac{\bar{z}}{\bar{y}}\right) = \frac{\bar{u}\bar{z}}{\nu}\left(\frac{\bar{z}}{\bar{x}}\right) \tag{10}$$

The substitution $\bar{v} \approx \bar{u}\frac{\bar{y}}{\bar{x}}$ was made in equation (10) because this relation must be true in order to make the terms of a nondimensionalized version of equation (4) of order 1. After all, the appropriate scales, such as $\bar{u}$ and $\bar{t}$, are chosen with the objective of forming dimensionless terms of order 1. Note that both equations (9) and (10) are essentially a Reynolds number multiplied by an aspect ratio, $\frac{\bar{z}}{\bar{x}}$, which is the classical $\frac{h}{L}$ commonly seen in lubrication theory. Continuing,

$$\frac{d}{h} \approx \frac{a_p}{\nu\bar{u}/\bar{z}^2} = \frac{a_p\bar{z}^2}{\nu\bar{u}} = 3 \tag{11}$$

(where equation (7) has been substituted for $\bar{u}$)

$$\frac{e}{h} \approx \frac{\frac{1}{\rho}\partial p/\partial x}{\nu\bar{u}/\bar{z}^2} = \frac{\partial p/\partial x\,\bar{z}^2}{\mu\bar{u}} \tag{12}$$

$$\frac{f}{h} \approx \frac{\nu\bar{u}/\bar{x}^2}{\nu\bar{u}/\bar{z}^2} = \frac{\bar{z}^2}{\bar{x}^2} \tag{13}$$

$$\frac{g}{h} \approx \frac{\nu\bar{u}/\bar{y}^2}{\nu\bar{u}/\bar{z}^2} = \frac{\bar{z}^2}{\bar{y}^2} \tag{14}$$

The time scale, $\bar{t}$, may be approximated as one period of an engine revolution, since it is anticipated (from experimental evidence) that the piston acceleration is the main forcing function for the flow. For a characteristic engine speed of 1500 rpm, the period is 40 msec. A value of $9.2 \cdot 10^{-4}$ Pa-sec is assumed for the dynamic viscosity, which is what one could reasonably expect using a 15w40 lubricant at a representative oil temperature of 325°C (see Section 3.6.1). Using a density of $850\,\text{kg/m}^3$, the momentum diffusivity, ν, is $1.08 \cdot 10^{-6}\,\text{m}^2/\text{s}$. $\bar{x}$, $\bar{y}$, and $\bar{z}$ are taken to be 10mm, 80mm, and 10μm respectively, as listed in Table 1.

Using these values, the expression for $\frac{a}{h}$ in equation (8), which is a ratio of viscous diffusion time $\frac{h^2}{\nu}$ to the timescale of the problem, is found to equal roughly 0.009. To evaluate $\frac{b}{h}$ and $\frac{c}{h}$ (equations (9) and (10)), the maximum value of $\bar{u}$ within a cycle is used, which comes out to be about $0.3\,\text{m/s}$ Accordingly, $\frac{b}{h}$ and $\frac{c}{h}$ evaluate to a value of 0.011. $\frac{d}{h}$ is 3, as shown in equation (11). Using the approximation for $\frac{\partial p}{\partial x}$ derived above, and $\bar{u} = \bar{u}_{max} \approx 0.3\,\text{m/s}$, equation (12) (term $\frac{e}{h}$) is approximately 0.14. Obviously this number is higher if a value of $\bar{u}$ other than $\bar{u}_{max}$

is used. Finally, $\frac{f}{h}$ and $\frac{g}{h}$ (equations (13) and (14)) are found to be $4 \cdot 10^{-6}$ and $6.3 \cdot 10^{-8}$, respectively.

Since $\frac{a}{h}$, $\frac{b}{h}$, $\frac{c}{h}$, $\frac{f}{h}$, and $\frac{g}{h}$ are all $\ll 1$, terms a, b, c, f, and g may be neglected. In addition, the estimates indicate that term e is less than term h, but by less than a factor of 10. However, the decision to neglect the pressure gradient was made, since during most of the cycle the pressure difference between the combustion chamber and the bottom of the top land is barely detectable. Note that dropping this term implicitly neglects the role of surface tension; see Section 3.5 for discussion. With the pressure gradient term neglected, only terms d and h remain in equation (4).

A similar analysis was conducted for equation (5). After simplifying equations (4) and (5) according to the scaling breakdown, simple ordinary differential equations for the oil velocities were derived:

$$\nu \frac{d^2 u}{dz^2} = a_p \tag{15}$$

$$\nu \frac{d^2 v}{dz^2} = 0 \ . \tag{16}$$

3.2.2 Robustness of Assumptions

It can be seen that, although a characteristic radial length scale of $20\,\mu\mathrm{m}$ was assumed, the results of the scaling analysis remain the same for film thicknesses well above this value. The pressure difference between the bottom of the top land and the combustion chamber is probably significantly less than the value used in the above calculations, because that value represented the pressure drop between the combustion chamber and the top ring groove, which is downstream of the bottom of the top land. Keeping all other parameters constant, equations (9) and (10) don't reach a value of 1 until the film thickness is slightly above $60\,\mu\mathrm{m}$. Still, in neglecting the "inertia" terms (or "advective" terms, i.e. the full left hand side of the Navier-Stokes equations), the scope of applicable usage of the model is reduced. For example, a significant enough reduction in the oil viscosity would cause the inertia terms to be on the order of the viscous terms. Likewise, the inertia terms would most likely dominate for racing engines, which have very high crankshaft speeds (typically in excess of 10,000 rpm).

3.2.3 Velocity Profiles

Two boundary conditions are needed to recover the velocity profiles from equations (15) and (16). First, the no-slip condition is used to set the velocity at $z = 0$ to 0. The second boundary

31
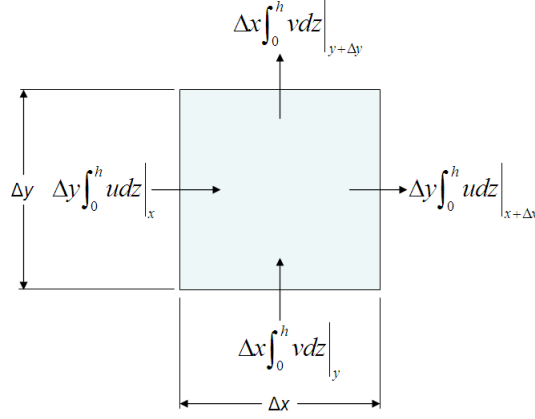
Figure 6: Control volume mass balance schematic.

condition comes from requiring the shear stress within the oil to match the shear stress within the gas, at the oil/gas interface. In other words,

$$\mu_{oil}\frac{\partial u_{oil}}{\partial z}\bigg|_{z=h} = \mu_{gas}\frac{\partial u_{gas}}{\partial z}\bigg|_{z=h} \tag{17}$$

$$\mu_{oil}\frac{\partial v_{oil}}{\partial z}\bigg|_{z=h} = \mu_{gas}\frac{\partial v_{gas}}{\partial z}\bigg|_{z=h} , \tag{18}$$

where u_{gas} and v_{gas} come from the input data set of gas velocities within the top land crevice. Integration of equations (15) and (16) subject to the boundary conditions discussed yields the axial and circumferential velocity profiles of the oil film:

$$u = \frac{a_p}{\nu_{oil}}\left(\frac{1}{2}z^2 - hz\right) + z\frac{\mu_{gas}}{\mu_{oil}}\frac{\partial u_{gas}}{\partial z}\bigg|_{z=h} \tag{19}$$

$$v = z\frac{\mu_{gas}}{\mu_{oil}}\frac{\partial v_{gas}}{\partial z}\bigg|_{z=h} \tag{20}$$

3.2.4   Governing Equation

Consider a control volume with arbitrary dimensions Δx x Δy x h, portrayed in Figure 6. The fluxes through the boundaries of this cell are depicted. Setting the sum of the fluxes equal to the net rate at which mass accumulates, and dividing by Δx and Δy, one obtains

$$\frac{\int_0^h udz\big|_{x+\Delta x} - \int_0^h udz\big|_x}{\Delta x} + \frac{\int_0^h vdz\big|_{y+\Delta y} - \int_0^h vdz\big|_y}{\Delta y} + \frac{\partial h}{\partial t} = 0 . \tag{21}$$

Taking the limit as Δx and Δy approach zero, and carrying out the integrals according to

equations (19) and (20), we arrive at the governing equation,

$$\frac{\partial}{\partial x}\left(-\frac{a_p}{3\nu}h^3 + \frac{1}{2}\frac{\mu_{gas}}{\mu_{oil}}\left.\frac{\partial u_{gas}}{\partial z}\right|_{z=h}h^2\right) + \frac{\partial}{\partial y}\left(\frac{1}{2}\frac{\mu_{gas}}{\mu_{oil}}\left.\frac{\partial v_{gas}}{\partial z}\right|_{z=h}h^2\right) + \frac{\partial h}{\partial t} = 0 \ , \qquad (22)$$

which is characterized in the next section.

Equation (22) is presented in *conservative* form; i.e. its spatial derivatives can be interpreted as being performed on flux functions, f and g:

$$\frac{\partial}{\partial x}\left(f\left(x,y,t,h\right)\right) + \frac{\partial}{\partial y}\left(g\left(x,y,t,h\right)\right) + \frac{\partial h}{\partial t} = 0 \ . \qquad (23)$$

3.3 Classification of the Governing Equation

Equation (22) is a partial differential equation (PDE). It has the following properties:

- MULTIDIMENSIONAL
  There are three independent variables: x, y, and t.

- SCALAR
  There is one dependent variable: h.

- NONLINEAR
  The fluxes f and g in equation (23) are proportional to higher powers of h, e.g. h^3.

- FIRST ORDER
  The derivatives are first order.

- HYPERBOLIC
  All first order PDE's are hyperbolic. The solution will exhibit wave behavior.

- VARIABLE COEFFICIENT
  The gas flows are allowed to vary arbitrarily throughout the computational domain.

In addition, since the equation can be written in conservative form, (23), it is often referred to as a *conservation law*.

Quite a lot of computational difficulties arise when seeking numerical solutions to this type of equation. To better understand these numerical issues, it helps to understand the qualitative behavior of solutions from an analytical point of view, as discussed in the following section.

3.4 Properties of Solutions to the Governing Equation

Equation (22) is a wave-type conservation law. It states that the rate at which the volume within some infinitesimally small control volume increases is proportional to the net volume flow rate through the control volume's boundaries. Before delving into numerical methods, properties of solutions to equation (22) are discussed. Several authors have presented excellent accounts of these properties. Due to the abundance of material available on this topic, the details here are sketched rather than developed in full. See [6], [7], [8], [9], [10], [13], [14], and [16], for example.

Due to its nonlinearity, one of the important aspects of the type of PDE considered here, which sets it apart from the other (elliptic, parabolic) types of PDEs, is that the solution can naturally develop discontinuities even if the initial conditions and boundary conditions are arbitrarily smooth. To shed light on how this process takes place, the method characteristics is applied to the governing equation.

3.4.1 Characteristic Curves

The x and y derivatives in equation (22) have not been carried through on purpose, for reasons that will be understood soon. To develop a solution using characteristics, equation (22) is first rewritten with these derivatives carried out.

$$-\frac{a_p}{\nu}h^2\frac{\partial h}{\partial x} + \frac{\mu_{gas}}{\mu_{oil}}\left.\frac{\partial u_{gas}}{\partial z}\right|_{z=h}h\frac{\partial h}{\partial x} + \frac{\mu_{gas}}{\mu_{oil}}\left.\frac{\partial v_{gas}}{\partial z}\right|_{z=h}h\frac{\partial h}{\partial y} + \frac{\partial h}{\partial t} =$$
$$-\frac{1}{2}h^2\left[\frac{\partial}{\partial x}\left(\frac{\mu_{gas}}{\mu_{oil}}\left.\frac{\partial u_{gas}}{\partial z}\right|_{z=h}\right) + \frac{\partial}{\partial y}\left(\frac{\mu_{gas}}{\mu_{oil}}\left.\frac{\partial v_{gas}}{\partial z}\right|_{z=h}\right)\right] \tag{24}$$

An important point to note, which will come up later in the analysis and results, is that the terms which represent the gradient of the axial gas flows (in the axial direction) and the gradient of the circumferential gas flows (in the circumferential direction) appear as source terms in this equation.

In equation (24), the spatial derivatives of the dependent variable, $\frac{\partial h}{\partial x}$ and $\frac{\partial h}{\partial y}$, appear explicitly. This equation is called the *strong* form (or sometimes, "quasilinear" form) of the conservation law. Likewise, equation (22) is a weaker form of the conservation statement[5]. In the strong form, the partial differential equation is forced to be satisfied in a strict (i.e. at every point) sense. In the weak form, the partial differential equation is only required to be satisfied in an integral (i.e. average) sense. When there are no discontinuities, the two forms are equivalent. The terminology stems from the fact that the strong form is slightly more restrictive than the

---

[5]The formal weak form of the equation will be presented.

weak form (all solutions of the strong form are solutions of the weak form, but not vice versa). However, the weak form is more fundamental. Nature requires that in the absence of nuclear reactions, the total mass stays constant. It does not care about the manner in which the mass is conserved (and hence, it does not rule out the possibility of discontinuities), as long as it is conserved.

Many of the features of solutions to equation (24) may be conveniently demonstrated using a simplified version. Consider a model equation

$$\frac{\partial h}{\partial t} + \frac{\partial}{\partial x}\left(f\left(h\right)\right) = 0 \ , \tag{25}$$

with $f\left(h\right) = \frac{1}{3}h^3$. Or, with the derivatives carried through, the strong form is

$$\frac{\partial h}{\partial t} + h^2 \frac{\partial h}{\partial x} = 0 \ . \tag{26}$$

In this PDE, $h = h(x,t)$. The total time derivative of h may be written out:

$$\frac{dh}{dt} = \frac{\partial h}{\partial x}\frac{\partial x}{\partial t} + \frac{\partial h}{\partial t} \ . \tag{27}$$

Define a curve C_x in the $x - t$ plane as

$$C_x\left(h\right) = h^2 \ . \tag{28}$$

Substituting $\frac{dx(t)}{dt} = C_x\left(h\right)$ and (28) into (27) yields

$$\frac{dh}{dt} = \frac{\partial h}{\partial x}h^2 + \frac{\partial h}{\partial t} = 0 \ , \tag{29}$$

which means that along the curve C_x, the film thickness is constant. C_x is termed a "characteristic curve." Interpreted temporally, a point along a characteristic curve "moves" in the x direction with velocity h^2, and the film thickness is constant. The interpretation is clearer upon expressing these findings as relationships (ODEs) which must hold *along the characteristic curve* defined by $\frac{dx}{dt} = C_x$:

$$\frac{dx}{dt} = C_x = h^2 \tag{30}$$

$$\frac{dh}{dt} = 0 \ . \tag{31}$$

Equations (30) and (31) completely specify the solution to equation (26), given initial and boundary conditions. Each characteristic curve has two traits (state variables): position and film thickness. Interestingly, each characteristic propagates at the wave speed, h^2. Hence, a characteristic
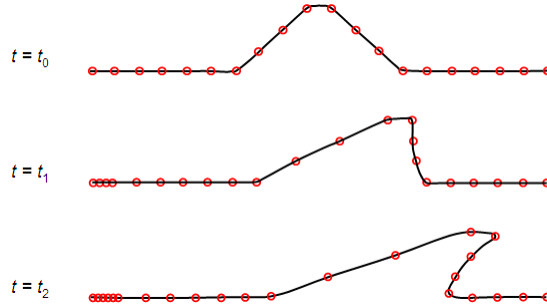
Figure 7: Creation of a multivalued solution due to nonlinearity.

whose h trait is large compared to other characteristics will move faster than than the other characteristics.

3.4.2 Shocks

The validity of the characteristics approach should seem questionable when one realizes that there is no provision within equations (30) and (31) for the characteristics to interact with one another. These equations seem more like expressions governing the ballistic motion of discrete particles within an ideal gas (a hypothetical type of matter in which the particles do not interact with one another) than a set of equations that would be appropriate for the mechanics of condensed matter. According to equation (31), the height h of the oil film along a characteristic should never change. Figure 7 depicts what the solution looks like when some characteristics overcome each other. It is a series of snapshots showing the evolution of the solution to equations (30) and (31), for which the initial conditions are single valued and (relatively) smooth. Each data point pictured represents the instantaneous x and h traits of one of the characteristic curves; bestfit curves were drawn between them for easy visualization of the wave structure they represent. The characteristics whose h traits are large move faster than the characteristics whose h traits are small, since the wave speed scales with h^2. As a result, after some finite time has passed, the wave steepens and even becomes multivalued[6].

The simplified equation has demonstrated one feature of the TLOMM governing equation, (22). Unfortunately, this lack of coupling between characteristics is mathematically incorrect, for two main reasons. First, looking back at Section 3.2, an implicit assumption was made just before arriving at equation (22). When the velocity profiles were integrated from 0 to h, it was assumed that there were no voids between $z = 0$ and $z = h$. Second, equation (22) is one to one; it does not admit multivalued solutions for which there could be more than one value of h at any

---

[6]Multiple values of h for one value of x.

given point x, y. Hence, the approach presented so far cannot be used (without modification) to generate correct solutions.

Why does the characteristics approach create a solution which, after a certain amount of time passes, eventually violates the initial governing equation? The reason is because the characteristics method solves the *strong* form of the PDE; as soon as one characteristic overtakes another, $\frac{\partial h}{\partial x}$ (which appears explicitly in the strong form) goes to infinity and the strong form <u>itself</u> becomes invalid.

After characteristics overtake one another, the correct solution to (22) is a nonlinear jump discontinuity. This claim can be substantiated by considering a parabolic version of equation (26), in which a small viscous (i.e. diffusive) term has been added. In the limit that this term's diffusivity coefficient approaches zero, the solution approaches a discontinuity. Not surprisingly, this argument is called the "vanishing viscosity approach."

This observed behavior, i.e. the creation of a discontinuity due to nonlinearity of the flux, despite smooth initial conditions, is the underlying principle behind what is referred to as a *shock*. Shock is a type of discontinuity and is a defining trait of PDE's of the type (22). The phenomenon inherits its name from the shock waves found in compressible gas flows at high Mach number; because the governing equations are of the same type (hyperbolic conservation laws) as the one studied in this project, solutions exhibit the same behavior. In fact, shock is manifested in many other areas of science as well - in studies concerning the motion of galaxies, multiphase fluid flow, blast waves, traffic flow, magnetohydrodynamics (e.g. in fusion reactors), and weather prediction, for example [7].

3.4.3 Weak Form

The strong form of the PDE may not admit solutions which are discontinuous, but the weak form does. The formal weak form of equation (25) would be [6]

$$\int_0^\infty \int_{-\infty}^\infty \left[h\varphi_t + f\left(q\right)\varphi_x \right] dx dt + \int_0^\infty h\left(x, 0\right)\phi\left(x, 0\right) dx$$

but in this form it has little use on its own.

A modification to the method of characteristics, called the "Equal Area Rule" (e.g. [14]), can be successfully applied to problems in two independent variables, as was done in [3]. A demonstration of this modification is found in Figure 8. However, in three independent variables, no generalization of this modification exists. In adding one independent variable, one goes from needing to find the shock *position* to needing to find the shock *front*: some arbitrary, perhaps

37
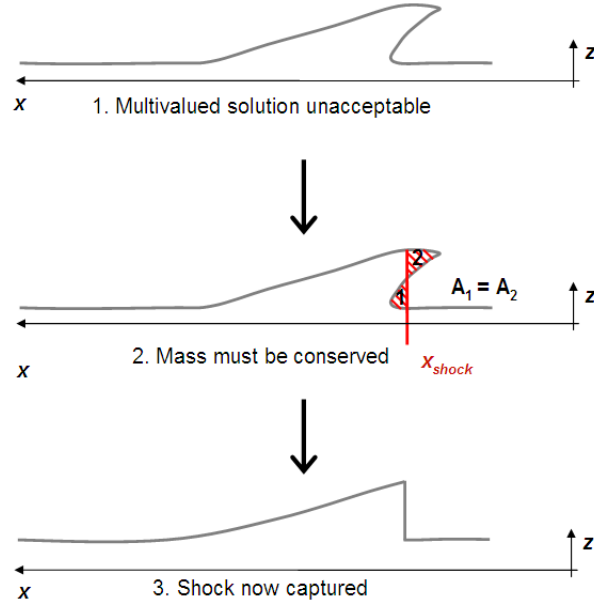
Figure 8: Demonstration of procedure for applying equal area rule.

discontinuous curve in the $x - y$ plane, given by $f(x, y) = 0$. After a good deal of effort in trying to derive a three dimensional generalization of the equal area modification, it was decided that characteristics alone cannot solve the problem. As a result, the weak form must be discussed.

The weak form is more fundamental and general than the strong form, but it comes with a price: when the solution becomes discontinuous, solutions to the weak form are no longer unique. For a demonstration, consider a classical example (from [16]): Burgers' equation is

$$\frac{\partial u}{\partial t} + u \frac{\partial u}{\partial x} = 0 \ , \tag{32}$$

where u is now the dependent variable. Take the initial condition to be

$$u(x, 0) = \begin{cases} -1 & x < 0 \\ 1 & x > 0 \end{cases} . \tag{33}$$

Several solutions can be found which satisfy a weak statement of (32) and (33). For one, the initial condition itself (a shock wave propagating at zero speed) is a solution for all times. A second possible solution is a rarefaction wave,

$$u(x, t) = \begin{cases} \text{-1} & x < -t \\ x/t & -t \leq x \leq t \\ 1 & x < t \end{cases} , \tag{34}$$

38

which can also be verified to satisfy a weak form of equation (32)[7]. It turns out that infinitely many weak solutions may be constructed. To single out the correct solution, the concept of entropy must be introduced.

3.4.4 Entropy

Solutions to the weak statement of nonlinear conservation laws are not unique once a solution develops a discontinuity. Some sort of additional constraint or condition is required in order to pick the physically correct solution among the set of solutions to the weak form.

From the vanishing viscosity argument, because all physical systems have some amount of viscosity, the solution "h" to (25) should be the same as the solution "u" one would obtain by solving

$$\lim_{\epsilon \to 0} \left(\frac{\partial u}{\partial t} + \frac{\partial}{\partial x} \left(f\left(u \right) \right) = \epsilon \frac{\partial^2 u}{\partial u^2} \right) \ . \tag{35}$$

To actually find a solution which agrees with the solution to (35), an "entropy condition" must be enforced. There are several variations on this condition, and [7] discusses them in detail. One version, presented in [15], requires that for all discontinuities,

$$\frac{f\left(u \right) - f\left(u_l \right)}{u - u_l} \geq s \geq \frac{f\left(u \right) - f\left(u_r \right)}{u - u_r} \ , \tag{36}$$

for all values of u between u_l and u_r. Here s denotes the shock speed. u_l and u_r are the values of u on the left and right hand sides of the discontinuity. $\frac{f(u)-f(u_l)}{u-u_l}$ represents the characteristic speed, $f'\left(u \right)$, at the left of a shock, and likewise for $\frac{f(u)-f(u_r)}{u-u_r}$. This condition can be interpreted as requiring that characteristics must run *into* shocks, not emanate from them. An alternative interpretion is simply that shocks must act as information sinks. Enforcing (36) guarantees that the solution obtained is the same as the vanishing viscosity solution. Hence, correct and unique solutions to (25) may be found by choosing the solution to the weak form which satisfies equation (36).

In the case of the TLOMM, the "entropy" discussed here is a notion even more abstract than usual; it is not related to the physical entropy of the oil, for example. The name "entropy condition" comes again from workers in the field of compressible gas dynamics. In their case, entropy does correspond to the standard thermodynamic property of a fluid referred to as "entropy," which is a measure of the disorder of the particles within that fluid, and physics dictates that the thermodynamic entropy must increase across a shock.

---

[7]Even without having to resort to the weak form, one can easily see that except at the points where the slope in u is discontinuous, equation (34) satisfies the *strong* form of Burgers' equation, (32).

To some extent, an analog can be observed between the fact that entropy may only increase across a shock and Boltzmann's fundamental definition of entropy (from statistical mechanics),

$$S = k \log W \ ,$$

where k is Boltmann's constant, and W is the multiplicity of states. For an ensemble of particles, W can be thought of as being the number of microstates that are available to the system at a given energy level. Of course this project is concerned with the continuum approximation, so we need not be concerned with microstates or individual particles, but the fact that the weak form goes from producing one unique solution (pre-shock) to producing many solutions (post-shock) is, in a way, similar to an increase in the multiplicity of an ensemble of particles. Hence even though the TLOMM's modeling steps make no use of the conservation of energy equation, nor require any mention of the thermodynamic entropy, the fact that entropy must increase (or that information can only be lost) still turns up.

3.4.5 Rankine-Hugoniot relation

The speed at which a shock should propagate, s, is well defined. Considering a mass balance across a shock, one obtains the Rankine-Hugoniot jump condition,

$$s = \frac{f(u_l) - f(u_r)}{u_l - u_r} \ , \tag{37}$$

which defines the shock speed as the difference in fluxes on the left and right sides of the shock divided by the size of the jump in u across the shock.

3.4.6 Total Variation

The total variation of a smooth function $u(x)$, having a domain $-\infty < x < \infty$, may be defined as

$$TV(u) = \int_{-\infty}^{\infty} \left| \frac{\partial u}{\partial x} \right| dx \ .$$

The total variation is essentially the sum of the absolute values of all variations of u over the whole domain. For a discontinuous function having jump discontinuities, the interpretation of total variation is the same, but to avoid the issue of $\frac{\partial u}{\partial x}$ taking on infinite values, the definition is slightly more complicated (see [8]).

It can be proven that solutions to certain classes of PDE's have total variations which do not increase with time. For example, according to the characteristics analysis performed earlier,

equation (29) shows that the total variation of solutions to equation (25), with $f(h) = \frac{1}{3}h^3$, does not increase with time. In fact, since shocks cause the film thickness to spread out (causing values of h to decrease), the total variation of solutions to equation (25) can only decrease in time. Equations having this property are referred to as Total Variation Diminishing, or TVD.

The full governing equation, (22), does not formally have the TVD property itself. The gas flows can potentially have spatial gradients, which act as source terms, as can be seen in equation (24). Additionally, even without any gradients in the gas velocities, since the flux function can be nonconvex (to be discussed in Section 4.4.3), decreases in h can occasionally yield an *increase* in the magnitude of the flux, making equation (22) not strictly TVD.

This odd property may at first seem like an abstract peculiarity of interest only to mathematicians. However, in Section 4, it will become obvious that an algorithm which creates TVD solutions when applied to continuous PDEs which are TVD themself is one of the key components of a successful numerical scheme.

3.5 Importance of Surface Tension

The informed reader has undoubtedly noticed that in formulating this model, surface tension was neglected. As the TLOMM is a model of a free surface, some justification is in order.

On a piston land, surface tension constrains the shape of the film's free surface. If the fluid has high enough surface tension, steep fronts having a small radius of curvature do not form (instead, the puddle spreads out). However, since nothing is in the model to prohibit the film from creating a steep slope, shocks form. The occurrence of shock causes the oil film to spread out, as mentioned in Section 3.4.2. Hence, in the TLOMM, *shocks play the role of surface tension.*

For this problem, the dimensionless parameter indicating whether or not surface tension is important is the Capillary number. Ca is an approximate ratio of viscous forces $\left(\mu\frac{\bar{u}}{h}\right)$ to surface tension forces $\left(\frac{\sigma}{R}\right)$,

$$Ca = \frac{\mu\bar{u}R}{\sigma h} \ , \tag{38}$$

where σ is the surface tension of the lubricant and R is the local radius of curvature. Substituting expression (7) into (38), one obtains

$$Ca = \frac{\rho a_p h R}{3\sigma} \ . \tag{39}$$

Obviously the Capillary number can vary quite a lot within an engine cycle, due to the dependence on the piston acceleration. Before Ca may be calculated, an appropriate value of σ is required.
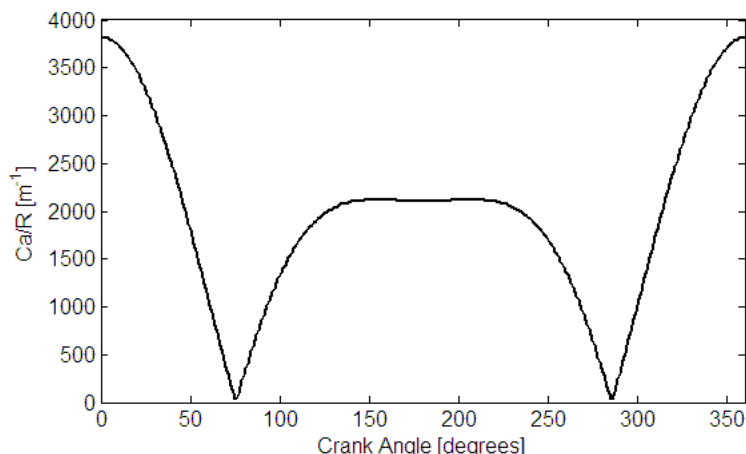
Figure 9: Plot of Capillary number divided by radius of curvature, throughout an engine cycle.

Temperature dependence of σ Surface tension is strongly temperature dependent. Estimating the temperature dependence of the surface tension of *pure* substances may be accomplished using the Eötvös rule,

$$\sigma V^{2/3} = k \left(T_c - T \right) \tag{40}$$

where V is the molar volume (molar mass divided by mass density), T_c is the critical temperature, and k is the Eötvös constant, $k = 2.1 \cdot 10^{-7} \, \mathrm{mol}^{-2/3} \, \mathrm{J/K}$. Unfortunately, the Eötvös rule is intended for pure substances, while lubricating oil is a mixture of many compounds. Determination of the critical temperature of oil, T_c, is also complicated by the fact that oil is a mixture. In [5], the material properties of several lubricants were studied and various equations of state were extracted from the experimental data. Oil was modeled as being composed of a finite number of species, each with its own properties, and the resulting aggregate properties were reported. For the sake of the scaling calculation, the properties of POE 68, a polyol ester oil, were chosen to be representative. From this paper, the reported critical temperature for POE 68 is 746 K, and the molar mass 700 kg/kmol. Assuming a density of 850 kg/m³ and a characteristic oil temperature of 325°C, as before, the surface tension predicted by equation (40) is $\sigma = 3.5 \, \mathrm{mN/m}$.

Estimated values of Ca Figure 9 shows a plot of the absolute value of Ca/R, using the characteristic scales estimated above and $\sigma = 3.5 \, \mathrm{mN/m}$, throughout one engine cycle. The units are m^{-1}. One can see that for a small radius of curvature (~250 μm or less), the Capillary number is less than one, for the entire engine cycle. In regions where the radius of curvature is large, say 1 mm or more, the Capillary number is greater than one for almost all of an engine cycle. Of course, Ca goes to 0 during the parts of the cycle when the piston acceleration goes to zero.

According to the these rough calculations, it is not surprising that surface tension is only impor-

$$R \sim \frac{L^2}{h} \qquad\qquad R \sim h$$

$Ca \gg 1$, surface tension unimportant $\qquad Ca \approx O(1)$
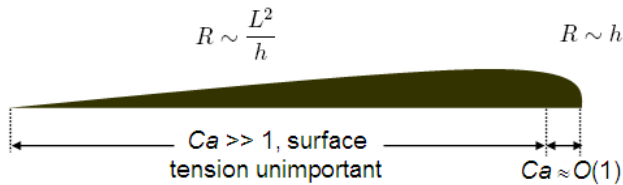
Figure 10: Schematic of the oil film.

tant in the parts of the puddle that have a small radius of curvature. Figure 10 shows a schematic of a typical oil puddle found on the piston lands in the experiments reported in [3]. For most of the puddle, the radius of curvature is large, and hence surface tension is not important (except when $a_p \to 0$). However, a very small portion near the "front" (location of max film thickness) of the puddle was typically observed in [3] to have a radius of curvature on the order of the film thickness itself. In this region, surface tension is dominant, as is evident from the calculations above. Still, since this region is very small compared to the puddle length, surface tension is negligible compared to the viscous forces within most of the puddle, during most of a cycle. As a result, it was decided to disregard surface tension in this model.

Consequences of neglecting surface tension Physically, the amount of curvature that may be attained by the surface of the oil film is governed by surface tension. Without surface tension in the model, no physics are present in the governing equation that would drive the puddle to spread out, except while shock is occurring. In addition to not addressing the issue of the stability of the surface, the model also does not account for the wetting contact angle made at the liquid-solid interface. These two effects interact with one another in complicated ways which simply cannot be quantified by the model presented here.

Wetting contact angle aside, the act of neglecting surface tension introduces two potential vulnerabilities concerning the issue of film stability. Compared to the actual behavior of the film taking place in an operating engine, the model's predictions *could* be either underly or overly dissipative. Consider both possibilities:

- At one extreme, the actual oil surface tension is much larger than the value predicted using equation (40). The front of the puddle spreads out before it is allowed to become steep, and the free surface is stable. The radius of curvature of the whole puddle remains relatively large, because external forces which attempt to bunch up the film into one location only slightly decrease the local radius of curvature. The strong internal cohesion between oil molecules means that the likelihood of the oil film breaking up, and droplets detaching from the surface, is low. If this case represented the reality taking place within an engine, the model's predictions would be underly dissipative, because in the simulation, the film does not dissipate at all until a very steep shape forms and shock occurs.

43

- At the other extreme, the oil surface tension is much lower than the value calculated using (40). The puddle radius of curvature can decrease down to an arbitrarily low amount, and the slope of the film at the front of the puddle may be arbitrarily steep. The free surface is not necessarily stable. The film thickness can be multivalued and take on shapes such as the bottom image in Figure 7[8]. The internal cohesion between oil molecules is low; droplets often break off and are thrown away from the bulk film when the puddle takes on the multivalued shape shown in Figure 7. If this case represented the reality taking place within an engine, the model's predictions would be overly dissipative; the model would dissipate the film whenever a steep front (shock) occurs, to avoid multivalued solutions, while in reality there would be little or no spreading out.

Luckily for the TLOMM, neither of these cases was actually observed in the engine tests performed in [3]. The oil film was consistently found to have a steep front edge at the ends of the downward and upward engine inertia periods. Surface tension was strong enough to prevent the film from breaking up, but was not so strong that the film was just completely spread out over the entire surface. Hence, on a piston land, surface tension appears to be important only in the vicinity of a steep front. Its role is analogous to the role viscosity plays in shock wave within a compressible gas; on a piston land, surface tension prevents the shape of the oil film from becoming completely discontinuous, while in a shock wave in air, viscosity prevents the flow variables from being completely discontinuous[9].

The TLOMM was intended to run hundreds of engine cycles on a desktop PC within a reasonable amount of time (a few hours), as described in Section 1.2. Inclusion of surface tension would have required discretization of the radial (z) direction, so the number of independent variables in the problem would have been four instead of three. The governing equation would not have been scalar; rather, it would have been a system of several PDE's. Though it would more accurately capture the details of the steep structure near the front of the film, and would predict fluid free surface phenomena such as the Rayleigh-Taylor instabilities observed in [3], this approach would have resulted in a much slower simulation than the one developed in this project. Simulations of film formation and breakup are notorious for being computationally expensive.

Without a doubt, accuracy is lost by neglecting surface tension. However, given the fact that the model agrees well with the behavior observed in an engine (according to [3]), the overall fluid transport behavior and top land oil film distribution should still be accurate enough to accomplish the objectives of the project.

---

[8]Though equation (22) is still inappropriate since it does not account for voids.
[9]The width of a shock wave is not zero, it is just small - several mean free paths of the gas atoms.

3.6 Auxiliary Models

In addition to the primary PDE model of the oil film, two simple submodels are used in this project which enhance the accuracy of the main model's predictions. Namely, the lubricant viscosity's temperature dependence is taken into account, and a simplified model of the gas flows in the top land crevice is derived (for use when externally provided input data is unavailable).

3.6.1 Lubricant Properties

The viscosity of typical multigrade lubricating oils used in IC engines depends on both temperature and shear rate in important ways. It is well known that engine oil can exhibit non-Newtonian fluid behavior, in which the local viscosity is dependent on local shear rate due to a phenomenon referred to as shear thinning. For a Couette flow between two parallel plates in relative motion, the shear rate γ is constant along the direction perpendicular to the flow, and can be calculated as

$$\gamma = \frac{U}{h} \ , \tag{41}$$

where U is the relative difference in speed of the plates and h represents the separation between the plates.

To determine whether or not shear thinning may be neglected, an estimate of the typical shear rate on the top land must be compared with the critical shear rate, β. Shear thinning is important if the shear rate is on the order of, or is greater than, the critical shear rate. From [17], β may be estimated using

$$\beta = 10^{a+bT} \ , \tag{42}$$

where a and b are correlation parameters usually read from a table such as the one found in Appendix A.

For a 15w40 lubricant, take $a = 2.3$ and $b = .0225$ $(°C)^{-1}$. For consistency, assume $T = 325°C$. Using these values, $\beta = 4.1 \cdot 10^9 \, \mathrm{s}^{-1}$. To compare the critical shear rate to the estimated shear rate, expected characteristic values of U and h, for equation (41), are required. Assuming an engine operating at 1500 rpm, with oil viscosity equal to the low-shear viscosity at $T = 325°C$, and oil film thickness of $20 \, \mu m$, the maximum oil velocity within a cycle (U_{max}) found to be about $0.57 \, \mathrm{m/s}$. Hence, the order of magnitude estimate of γ is found to be $2.9 \cdot 10^4 \, \mathrm{s}^{-1}$. As a result,

$$\frac{\gamma}{\beta} \approx \frac{2.9 \cdot 10^4}{4.1 \cdot 10^9} \ll 1 \ ,$$

and it is safe to assume that shear thinning is unimportant for the TLOMM.

It is interesting to note that in some locations within an engine, the shear rate is very high, and shear thinning plays an important role. For example, the lubricating oil trapped between a piston ring and the liner typically exhibits shear thinning [21]. For the sake of demonstration, a characteristic film thickness during midstroke is on the order of a few, say 3, microns [2]. A typical mean piston speed of $7.5 \, \mathrm{m/s}$ would give rise to a shear rate $2.5 \cdot 10^6 \, \mathrm{s}^{-1}$. A characteristic temperature of the liner is much less than that of the top land the piston; at midstroke, the local temperature would probably be in the neighborhood of $150°\mathrm{C}$ for most engines [21]. At this temperature, the critical shear rate is much lower than it was above; repeating the calculation gives a critical shear rate of $4.7 \cdot 10^5 \, \mathrm{s}^{-1}$. Obviously shear thinning is important in this scenario, since the estimated shear rate of the oil between the ring and liner exceeds the critical shear rate.

At the (relatively) low shear rates expected in the TLOMM, oil behavior is well approximated by treating it as a Newtonian fluid and neglecting shear thinning. Regardless of shear rate, oil viscosity is a decreasing function of temperature. This dependence may be modeled using the Vogel equation [2],

$$\mu_0 \left(T \right) = k e^{\frac{\theta_1}{\theta_2 + T}} \ , \tag{43}$$

where k, θ_1, and θ_2 are also obtained from a table such as the one included in Appendix A. μ_0 represents the low-shear viscosity corresponding to temperature T. As shear thinning is neglected, the TLOMM uses the viscosity μ_0 in its calculations. Of course, viscosity depends strongly on the degree to which the lubricant has degraded, but this aspect is not modeled at present.

3.6.2 Top Land Crevice Gas Velocities

Ideally the CFD gas flow input data would be available for the entire engine cycle, and the TLOMM simulation would use this data exclusively without any need for a sub-model. However, many engine companies only simulate a portion of a cycle, usually focusing on the part of the cycle in which combustion takes place. To approximately account for the effect, if any, of gas flows during the parts of the cycle for which gas velocity data may not be available, a simplified model of the axial gas flow in the top land crevice was developed. The shear stress expression derived here is the same as the one presented in [3].

Reynolds number An order of magnitude estimate of the ratio of inertia forces to viscous forces in the gas flow is given by

$$Re = \frac{\rho V D_h}{\mu}$$

where D_h is an approximate hydraulic diameter, $D_h = \frac{4 A_c}{\wp}$ [20]. For this problem $D_h = 2 h_{gas}$ is appropriate, where h_{gas} is the clearance between top land and liner. A characteristic velocity

which might take place during the exhaust and intake strokes (when the axial pressure gradient is small) can be estimated to be the mean piston speed, $\bar{S}_p = 2LN$ [18], where L is the engine stroke and N is the crankshaft speed, in revolutions (not radians!) per unit time. For a typical heavy duty diesel engine operating at 1500 rpm, take $V = \bar{S}_p = 7.5\,\text{m/s}$. The gas density ρ varies significantly throughout the cycle, due to the compression stroke, but when the valves are open (during exhaust & intake) the density should be about the same as that found in ambient conditions. For this calculation, the gas density was approximated by $\rho = 1.2\,\text{kg/m}^3$, which is the atmospheric density of air at sea level. The viscosity was taken to be $\mu = 184.6 \cdot 10^{-7}\,\text{Pa-sec}$, roughly the viscosity of air at room temperature [19]. h_{gas} was taken to be $900\,\mu\text{m}$, which is a typical clearance between the top land and the liner. Using these estimated parameters, Re is found to be 880. Hence, the flow is expected to be laminar.

The Reynolds number was also calculated for the part of the cycle typically simulated by engine manufacturers. Using a representative average velocity taken from supplied CFD input data, Re of the gases within the top land crevice during the combustion part of the cycle was estimated to be 4200, which indicates that the flow is not laminar but the turbulence is probably not fully developed [20]. This calculation used the following parameter values, which would be representative of the crevice environment during combustion: $\rho = 23\,\text{kg/m}^3$, $V = 8\,\text{m/s}$, $\mu = \mu\,(T = 2300\,\text{K}) = 7.7 \cdot 10^{-5}\,\text{Pa-sec}$, and $h_{gas} = 900\,\mu\text{m}$.

Mach number   The speed of sound in atmospheric conditions, V_s, is roughly $340\,\text{m/s}$. Using the characteristic speed V estimated above, the Mach number

$$Ma = \frac{V}{V_s}$$

is expected to be 0.02. Hence, compressibility effects may be neglected.

Model   The flow was assumed to be one dimensional, in the axial direction. The coordinate system was defined attached to the piston, with x being the axial direction (positive upward) and y being the radial direction (positive outward). As before, the piston acceleration term was included to account for the acceleration of the reference frame.

A scaling analysis was performed on the full Navier-Stokes equations, along the same lines as Section 3.2. The piston acceleration term was found to be significant and was included. The term $u\frac{\partial u}{\partial x}$ was found to be potentially significant, but the assumption of 1D flow implies, to the contrary, that $\frac{\partial u}{\partial x} = 0$, due to the continuity equation. This contradiction indicates that the assumption of 1D flow is a bit simplistic. However, in the middle of the land, far from the top ring and far from the combustion chamber, the flow is probably fully developed, and $\frac{\partial u}{\partial x}$ is

probably zero. As in [3], the purpose of this sub-model is simply to get an order of magnitude expression. Hence, the fluid inertia terms were neglected and the resulting simplified governing equation is

$$\frac{dp}{dx} = \mu_{gas}\frac{d^2u}{dy^2} - \rho a_p \; , \tag{44}$$

with boundary conditions

$$
\begin{aligned}
u\,(y = 0) &= 0 \qquad \text{and} \\
u\,(y = h_{gas}) &= -S_p \qquad ,
\end{aligned}
\tag{45}
$$

where partial derivatives have appropriately been replaced by total derivatives in (44). S_p is the instantaneous piston speed.

Solving (44) with (45) yields the velocity profile in the piston frame,

$$u = \frac{1}{2\mu}\frac{dp}{dx}\left(y^2 - yh_{gas}\right) - S_p\frac{y}{h_{gas}} + \frac{1}{2\nu}a_p\left(y^2 - yh_{gas}\right) \; . \tag{46}$$

Due to incompressibility and the assumption that the flow is one dimensional, in the fully developed region the net mass flow through any cross section must be zero:

$$\int_0^{h_{gas}} u\,dy = 0 \; . \tag{47}$$

Carrying out the integral gives an expression for the pressure gradient

$$\frac{dp}{dx} = -6\mu S_p\frac{1}{h_{gas}^2} - \rho a_p \; . \tag{48}$$

Hence, a pressure gradient is set up which modifies a Couette flow velocity profile just enough to satisfy the requirement that no net mass flow take place. Substituting equation (48) into (46) gives the velocity profile,

$$\frac{u}{S_p} = 2\frac{y}{h_{gas}} - 3\frac{y^2}{h_{gas}^2} \; , \tag{49}$$

which is plotted in Figure 11[10]. The shear stress at the oil surface is thus

$$\tau_{gas} = \frac{2\mu_{gas}S_p}{h_{gas}} \; . \tag{50}$$

According to this expression, the shear stress exerted on the oil by the gas flow actually drives the film in the direction of the piston velocity (in contrast with what would happen if there was no liner and the piston was just moving through free space). Note that the equations in this

---

[10]Interestingly, the a_p (piston acceleration) term is cancelled out and does not appear in the final expression for the velocity profile.
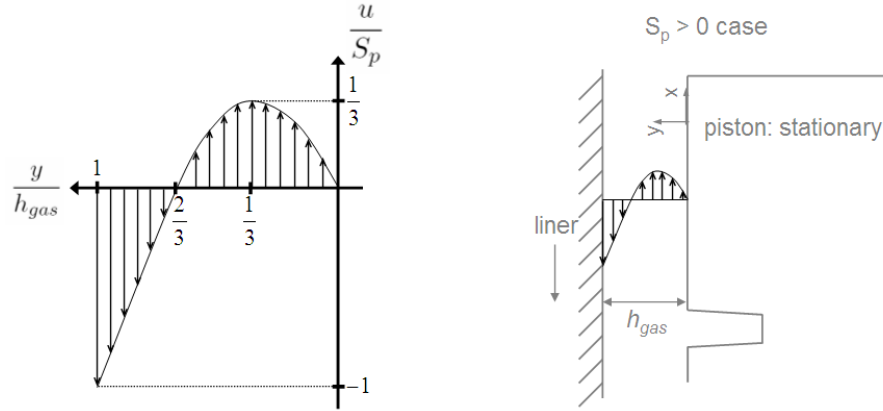
Figure 11: Simplified axial gas velocity profile, in the reference frame of the piston. Top land clearance exaggerated for clarity.

analysis do not agree with the equations in [3], partly due to the inclusion of the noninertial body force ρa_p, yet the end results ((50) and equation (3.55) of [3]) do agree completely. Of course, due to all of the assumptions, this expression is only close to being correct within a region that is sufficiently far from both the top ring and the combustion chamber, where the flow is expected to be fully developed.

When calculating the velocity gradient within the gas (as needed for equations (17) and (18)), this model assumes that the velocity of the oil film is small enough, relative to the gas velocity, to be neglected. So in practice, the TLOMM slightly exaggerates the effect of the gas flows; the gas dragging mechanism has more of an effect in the model than it would in reality, because the oil surface velocity would not simply be the piston velocity. As mentioned in Section 3.6.1, a characteristic maximum oil velocity is $0.57\,\mathrm{m/s}$, whereas a characteristic gas velocity (from actual CFD data or from the mean piston speed) is around $8\,\mathrm{m/s}$[11]. Therefore, since $\frac{0.57}{8} \ll 1$, this "static film" assumption is justified. As such, the TLOMM uses Equation (50) for the axial shear stress exerted by the gas on the oil surface whenever externally provided data is not available.

Expected effect of gas flow force   As discussed in [3], the inertia force should be expected to play the dominant effect when the oil thickness is large, but gas dragging should be dominant for very thin oil layers. The cause for two regimes of dominance is due to the fact that according to the model, equations (22) and (23) show that the inertia induced flux scales with h^3, while the gas flow induced flux scales with h^2. A rough idea of the relative effects of the two (sometimes competing) forces can be obtained by forming a ratio of the coefficients of the first two terms in

---

[11]Both are relative to the piston.
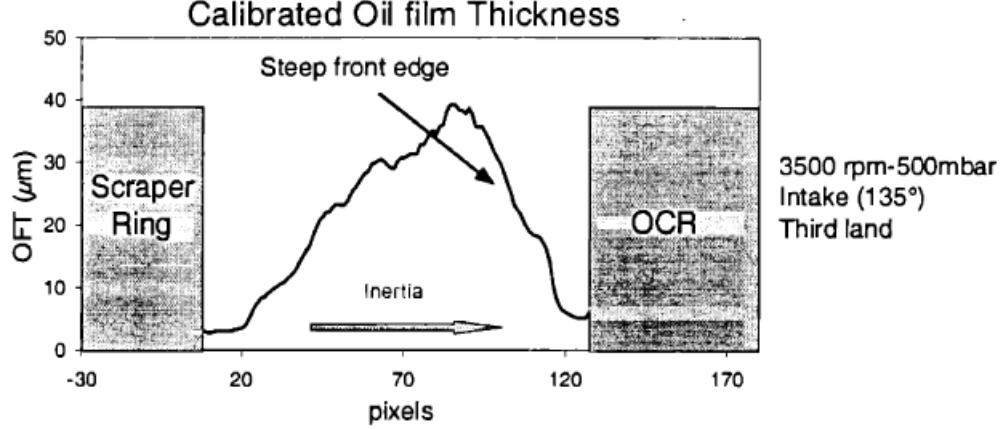
Figure 12: Experimental oil film thickness measurement, from [3].

equation (24):

$$\frac{\text{term}\,1}{\text{term}\,2} = \frac{\frac{a_p h^2}{\nu_{oil}}}{h\frac{\mu_{gas}}{\mu_{oil}}\left.\frac{\partial u_{gas}}{\partial z}\right|_{z=h}} = \frac{a_p h \rho_{oil}}{\mu_{gas}\left.\frac{\partial u_{gas}}{\partial z}\right|_{z=h}}\ .$$

Taking representative values as $a_p = \bar{a}_p \approx 1000\,\text{m/s}^2$, $h = 20\,\mu\text{m}$, $\rho_{oil} = 850\,\text{kg/m}^3$, $\mu_{gas} = \mu_{air}\,(T = 2300\,\text{K}) = 7.7\cdot10^{-5}\,\text{Pa-sec}$, and $\left.\frac{\partial u_{gas}}{\partial z}\right|_{z=h} = 3.3\cdot10^4\,\text{s}^{-1}$, this ratio evaluates to about 6.7, indicating dominance of the inertia force. Still, this ratio varies substantially throughout a cycle (for example, when a_p goes to zero), so it is not surprising that the simulation finds the gas flows to actually play an important role, as will be discussed in Section 6.

3.7   Experimental Validation of the Model

In [3], the model's validity was explored via extensive comparisons with experiments. The interested reader is encouraged to consult that document, as only a few of the main points (with a few extensions) are presented here.

Inertia driven oil flow   Figure 12, reprinted from [3], is provided as an example of the experimental film thickness measurements performed in that work. In these experiments, the oil layer was observed to form a steep front consistently. For reasons discussed in Section 3.5, this observation makes the decision to neglect surface tension (and replace it with a mechanism that captures shocks) acceptable. It is also consistent with the model's nonlinear flux; as discussed in Section 3.4.1, the presence of nonlinear fluxes mean that the wave speed is an increasing function of the film thickness, and hence steep fronts are expected to develop.

50

Oil flow induced by gas flow   In addition to studying the effect of piston inertia on the oil film, experimental studies were performed in [3] to investigate the effects of gas flows on the oil film. Circumferential oil transport was observed to take place on the second and third lands. As the direction and magnitude of the circumferential oil flow depended very distinctly on the relative locations of the ring gaps, circumferential gas flows were by far the most probable explanation for this transport mechanism. Compared to the piston inertia driven flow, the instantaneous gas flow driven circumferential oil flow rate was low. However, since the integral of the piston acceleration over the whole cycle is zero, the inertia driven flow causes the film to oscillate back and forth but not really make any net progress from one cycle to the next. The *net* inertia driven flow (say, per cycle) is actually quite small and is directly proportional to the rate at which new oil is supplied[12]. On the other hand, the circumferential gas flow can act in the same direction for quite a while, so although the circumferential oil flow rate at any instant is small compared to that of the inertia driven (axial) flow, a significant amount of *net* transport in the circumferential direction can still take place. The experimental work in [3] confirmed the importance that the crevice gas flows can have on the oil film distribution on piston lands, hence the reason for including the gas flow effect in this project.

Finally, from the strong form of the governing equation, (24), it can be seen that the wave speed associated with the gas flow terms is proportional to h. Therefore the model predicts that steep fronts will be generated for the gas flow driven oil flows (with or without piston acceleration), due again to the fact that the flux is a nonlinear function of h. Steep fronts were observed experimentally in [3] for the gas flow driven circumferential oil flows, consistent with the model's predictions.

---

[12]Additionally, the *direction* of this net flow depends entirely on the location and phase (within the cycle) at which the oil is introduced.

This page was intentionally left blank.

4   Numerical Approach

A great deal of work has been devoted to developing suitable algorithms for solving nonlinear, hyperbolic PDE's of the general form of (23) (reproduced below, for convenience) numerically.

$$\frac{\partial}{\partial x}\left(f\left(x,y,t,h\right)\right) + \frac{\partial}{\partial y}\left(g\left(x,y,t,h\right)\right) + \frac{\partial h}{\partial t} = 0$$

In fact, a substantial number of researchers (e.g. Godunov, Roe, Van Leer, Toro, LeVeque, Glimm, etc.) have made a lifelong career out of this pursuit.

In a nutshell, the possibility of naturally occurring discontinuities introduces an array of numerical difficulties. One would be correct to conclude that designing an appropriate numerical method is not trivial. Not surprisingly, the majority of the time spent developing the Top Land Oil Movement Model (TLOMM) was directed toward finding and implementing a suitable algorithm.

This part of the document builds up the numerical method used by the TLOMM; Section 5 covers details of its implementation. In all honesty, this section really only scratches the surface of numerical methods for hyperbolic conservation laws. Its main purposes are to 1) convince the reader of the existence and importance of many of the issues faced in seeking numerical solutions, 2) improve the typical TLOMM user's ability to interpret the simulation's predictions, and 3) provide a starting point for a researcher who is interested in seeking numerical solutions to their own conservation law. To appeal to an audience wider than just a small set of CFD specialists, as often as possible, mathematical descriptions are augmented with pictorial demonstrations of the issues. This way a reader should be able to get at least a qualitative understanding of the concepts without having to learn all of the math as a prerequisite. This section is not as detailed, complete, or rigorous as most of the references it cites. All the same, the author has found only a small amount of literature concerned with *variable* coefficient conservation laws, such as (22). Several numerical issues arise due to this aspect of the equation, so the work described here may actually add something to the existing body of knowledge.

4.1   Introduction

A two dimensional (two independent variables, x and t) TLOMM simulation was created first. Its satisfaction of the requirements (to be defined) was confirmed before extending it to three dimensions. The 2D simulation focused on solving this simplified version of equation (22):

$$\frac{\partial}{\partial x}\left(-\frac{a_p}{3\nu}h^3\right) + \frac{\partial h}{\partial t} = 0 \tag{51}$$

Again, partial differential equations of this form are classified as "conservation laws." To proceed, it helps to consider the various approaches that are typically used to solve PDEs numerically.

4.1.1 Approach Categories

There are many ways to approximately solve equation (51) using a computer; the most appropriate choice depends on the problem at hand, and is debatable at best. Roughly speaking, the available discretization viewpoints may be introduced as follows.

- FINITE DIFFERENCE
  This method directly approximates each derivative term in the governing PDE with a truncated Taylor series expansion. FD typically requires the least amount of time to set up, but has some negative traits that make it less than optimal when solving a nonlinear conservation law.

- FINITE VOLUME
  This type of approach stresses the integral (as opposed to differential) form of the governing PDE. Rather than point values interacting with one another, in FV the domain is divided up into cells, and it is the cell averages that interact with one another. For conservation laws, FV approaches are the most common and (arguably) the most successful.

- FINITE ELEMENT
  The Finite Element method casts the governing PDE into a variational formulation and satisfies some minimization statement via the use of basis functions. The Discontinuous Galerkin method is one modification to the standard FE method which admits solutions that can have discontinuities. DG is briefly discussed in Section 4.7.6.

In this work, a shock capturing "high resolution scheme," which is one brand of the finite volume method, was chosen. There are many other successful approaches, e.g. the Boundary Element Method ([60]); the relative merits of a few of these alternatives are discussed in Section 4.7.

The following section will begin to cultivate an intuitive feel for some of the traits needed for a numerical method to be successfully applied to equation (51) (and ultimately (22)); the finite volume viewpoint, in particular, is developed.

4.2 Concepts

To understand the impetus for using some of the more advanced approaches, it is required to at least introduce a few of the concepts relevant to seeking numerical solutions of equation $(22)^1$. Most of these concepts are discussed in standard textbooks on numerical methods for PDE's, e.g. [28], [50], [6], and [16]. The ideas developed in this section are later utilized in Section 4.4, where powerful shock capturing schemes are constructed.

4.2.1 Fundamentals

Discretization schemes have some inherent properties which are too important to go without being mentioned. In the history of numerical analysis, the fundamentals were first understood using finite difference schemes. These notions are briefly explored at present; formal presentations may be found in the references.

Truncation error A difference calculation approximates a local function value using a Taylor series expansion. As only a finite number of the terms in the series can be used in any practical computation, some error must exist between the value calculated and the exact solution.

A discretization scheme replaces a continuous differential equation with a set of algebraic equations which (one would hope) approximate its behavior. Following the approach of [16], one can consider a general partial differential equation,

$$\mathcal{L}u - f = 0 \ , \tag{52}$$

where $\mathcal{L}$ is any continuous operator, e.g. $\frac{\partial^2}{\partial x^2} + \frac{\partial^2}{\partial y^2}$, and f is the "inhomogeneous" or "source/sink" term (if any). A difference scheme for equation (52) could be expressed as

$$\hat{\mathcal{L}}\hat{\mathbf{u}} - \hat{\mathbf{f}} = 0 \ , \tag{53}$$

where the bold variables indicate vectors containing the values at each point in the computational domain. If the discrete operator $\hat{\mathcal{L}}$ is brought to act upon the continuous values u, the local truncation error at some arbitrary point i, τ_i, is defined as

$$\tau_i = \left(\hat{\mathcal{L}}u - \hat{f} \right)_i \ . \tag{54}$$

---

[1]An interesting common theme is that many of the behaviors exhibited by numerical methods (which are confined to take place within a computer, in a world of only ones and zeros) are analogous to physical phenomena, e.g. stability, dissipation, dispersion, entropy, etc.

Consistency   A result that could be deduced from the definition (54) is that any (successful) numerical scheme must satisfy

$$\lim_{\Delta x \to 0} \tau_i = 0 \quad \text{for all i} = 1, ..., n \tag{55}$$

for all smooth functions u satisfying (52). A discretization scheme possessing this property is referred to as *consistent*, because satisfaction of equation (55) implies that the difference equation generated by the scheme, (53), is in fact representative of the original (continuous) differential equation, (52).

Stability   All life forms encounter the concept of stability every day. Most physical systems may be categorized as stable or unstable. An inverted pendulum (e.g., a pencil standing upright on a table) is an example of a system in an unstable state; any perturbation to the pendulum's position is amplified by gravity and the system swings down, slowly coming to rest in its natural state. Once it has come to rest, it is now in a stable state; any perturbation to its position just results in the pendulum returning to its rest state.

One would hope that the issue of stability is confined entirely to the physical "real world." Unfortunately, in 1928 it was discovered [29] that in addition to physical systems, numerical methods for differential equations may also be categorized as stable or unstable. Some methods are unconditionally stable, some are unconditionally *un*stable (!), and many are stable provided that certain conditions are met. In essence, an unstable scheme amplifies the errors inherent to approximate methods. Examples of how this phenomenon can come to be are extremely useful; the one in Section 16.6 of [27], in particular, is very simple yet highly illustrative.

Classically, the stability of a scheme is analyzed via a Fourier stability analysis, which itself is a linear method. The references cited above cover at least linear stability analysis in detail. A key dimensionless number consistently arises in stability analyses of explicit (to be defined) schemes applied to hyperbolic PDE's. This parameter, referred to as the Courant number, represents the number of grid cells (of dimension Δx) traversed by a wave in one time step,

$$C = v \frac{\Delta t}{\Delta x} \tag{56}$$

where v is the local characteristic speed. C could also be interpreted as the ratio of the local wave speed to the maximum speed at which information could possibly travel for a discretization with grid spacing Δx and time step Δt.

Typically the outcome of a stability analysis is a restriction on the allowable values of C; for
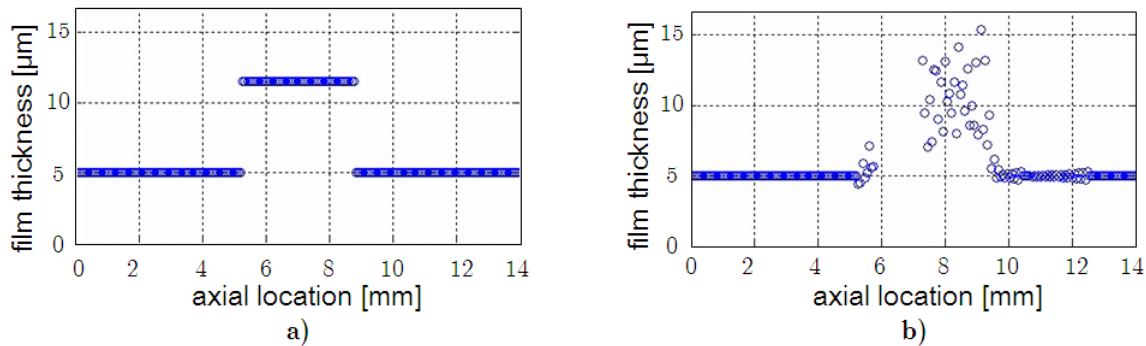
Figure 13: Numerical instability: a) initial condition; b) "solution" after some time has passed.

example, requiring that

$$|C| = \left| v \frac{\Delta t}{\Delta x} \right| \leq 1 \tag{57}$$

is very common for simple explicit schemes applied to nonlinear conservation laws. This particular requirement should make intuitive sense for the case of a simple single-stage time stepping algorithm, with a spatial discretization for which each node utilizes only information from its two nearest neighbors (one to the left, one to the right). Consider an example where a wave starts at, say, node $i - 2$ (at time level n), and arrives at node $i + 4$ (at time level $n + 1$). Since the evaluation of the derivative at node i is based only on the time level n information at the neighboring nodes, $i - 1$ and $i + 1$, node i's calculation could not possibly account for the presence of the wave, even though the wave passes *through* node i between time levels n and $n + 1$! Obviously the calculation would be erroneous.

For some schemes, no value of C will make the scheme stable, while for other schemes (usually *implicit* ones, to be discussed), stability is achieved for any value of C.

Finally, because the hyperbolic conservation law considered in this work is nonlinear, the concepts of linear stability do not entirely apply, and a stronger (nonlinear) stability condition must be met. It turns out that requiring the scheme to maintain the TVD property (presented in Section 3.4.6) of the continuous equation causes the scheme to have the desired stability characteristics, as will be discussed in Section 4.2.9.

In case this presentation has not convinced the reader of the reality of numerical stability, Figure 13 depicts a compelling example of numerical *in*stability. The initial film thickness distribution is a square wave, as shown in Figure 13a. Soon after the simulation of equation (51) begins, the solution develops a maximum Courant number that is too large, errors are amplified, and before long values start shooting off to infinity. Of course as soon as one cell's film thickness becomes infinite, the solution is destroyed, because that cell's infinite value propagates to its neighbors and so on. Figure 13b depicts the film distribution after the onset of numerical instability has

57

taken place.

Convergence Of course the end goal of going to the trouble of constructing a numerical scheme is that the approximate solution it generates should become closer and closer to the exact solution, as the spatial and temporal resolutions are refined successively. With this property, called *convergence*, a numerical solution's quality may essentially be improved to achieve any arbitrary level of accuracy. The LAX EQUIVALENCE THEOREM provides a mechanism for determing whether or not a scheme is convergent, and serves as one of the foundations of numerical analysis. Simply stated, this fundamental theorem asserts that a numerical method applied to a PDE is convergent only if it possesses both the properties of consistency and stability. The converse is also true; by definition, any convergent scheme is both stable and consistent. The *type* of stability required for a scheme to be convergent is a subtle point, and it depends on the type of equation and method; consult the references cited in the beginning of this section for details.

4.2.2 Finite Volume Method

The finite volume (FV) method is closely related to the finite difference method, but it is essentially set up with approximating the *integral* form of the governing PDE in mind. Not surprisingly, FV views the domain of a problem as being composed of a set of computational cells that have some finite volume associated with them, not simply composed of discrete points (which have zero volume) as in FD. Calculations are performed on the cell averages, not pointwise values. The definition of this instantaneous cell average, $\bar{u}_i^n$ (where u is the dependent variable, and can be a scalar or vector), at time level n in some cell i presents no surprises:

$$\bar{u}_i^n = \frac{1}{\Delta x} \int_{x_{i-1/2}}^{x_{i+1/2}} u\left(x, t^n\right) dx$$

For some applications, FV and FD calculations turn out to be identical. However, at least for nonlinear hyperbolic problems, FV has found much more success than FD, due to its abilities to admit discontinuities and to strictly enforce the conservative properties of conservation laws[2].

To keep the notation simple, for the rest of this document, the overbars are not explicitly included. u_i shall always automatically imply $\bar{u}_i$, the cell average.

4.2.3 Conservation

Unfortunately, for partial differential equations meeting the classification (Section 3.3) of equation (22), convergence of the scheme is essential, but not nearly sufficient on its own to produce

---

[2]One additional advantage of FV over FD is its much simpler adaptation to somewhat more general geometries.
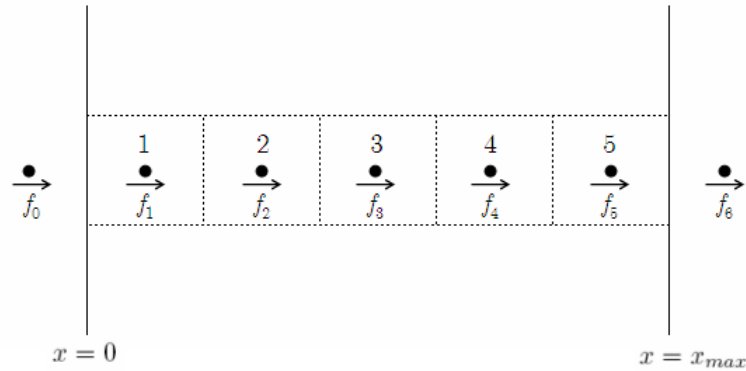
Figure 14: Example of a potentially non-conservative choice of flux evaluation points.

high quality results. Despite the fact that equation (22) is a "conservation" law, many numerical schemes applied to it will not discretely conserve mass. This phenomenon may be readily demonstrated.

Demonstration Consider equation (51) rewritten as

$$\frac{\partial h}{\partial t} + \frac{\partial f}{\partial x} = 0 \; , \tag{58}$$

where f now could be any arbitrary flux, which may be a function of h and/or x and/or and t. Naive replacement of the spatial derivatives with centered differences and the time derivative with an Euler Forward approximation gives

$$\frac{h_i^{n+1} - h_i^n}{\Delta t} + \frac{f_{i+1}^n - f_{i-1}^n}{2\Delta x} = 0 \; , \tag{59}$$

where f_{i+1}, for example, is understood to mean evaluation of the flux at the point $i+1$. Conservation arguments imply that the time rate of change of the volume contained within the boundaries must equal the net volume flow rate through the boundaries of the domain. Hence, a quick calculation can check whether or not the numerical scheme preserves this conservation property. For illustration's sake, consider a domain with just 5 nodes, as displayed in Figure 14.

Assume equation (59) is applied at each cell. Taking a sum of (59) over all cells gives

$$\sum_{i=1}^{5} 2\Delta x \frac{h_i^{n+1} - h_i^n}{\Delta t} = \sum_{i=1}^{5} f_{i-1}^n - f_{i+1}^n \; , \tag{60}$$

59

and in the limit that $\Delta t \to 0$, this equation becomes

$$2 \left. \frac{dV}{dt} \right|_{t=t_n} = \sum_{i=1}^{5} f_{i-1}^n - f_{i+1}^n \ , \tag{61}$$

where V is the volume contained within the domain. According to the argument above, mass conservation requires that $\frac{dV}{dt}$ must equal the net flow across the boundaries. In this choice of discretization, the flow *at* the boundary is not actually defined, since each boundary falls halfway between the nodes to the left and right of it. Carrying out the remaining sum in (61), one obtains

$$2 \left. \frac{dV}{dt} \right|_{t=t_n} = (f_0^n - f_2^n) + (f_1^n - f_3^n) + (f_2^n - f_4^n) + (f_3^n - f_5^n) + (f_4^n - f_6^n) \ ,$$

which obviously comes out to

$$2 \left. \frac{dV}{dt} \right|_{t=t_n} = \frac{f_0^n + f_1^n}{2} - \frac{f_5^n - f_6^n}{2} \ . \tag{62}$$

Although the terms $\frac{f_0^n + f_1^n}{2}$ and $\frac{f_5^n - f_6^n}{2}$ represent rough averages of the flow rates across the two boundaries of the domain, there is no guarantee in equation (62) that $\frac{dV}{dt}$ truly is exactly equal to the net flow across the boundaries. The values of h at the boundaries ($h_{1/2}$ and $h_{5+1/2}$) are not even calculated according to (59), so it would be impossible to prove that $f\left(h_{1/2}\right) = \frac{f_0^n + f_1^n}{2}$ for general flux functions.

Now consider a different choice of spatial discretization of the $\frac{\partial f}{\partial x}$ term in (58):

$$\frac{h_i^{n+1} - h_i^n}{\Delta t} + \frac{f_{i+1/2}^n - f_{i-1/2}^n}{\Delta x} = 0 \ . \tag{63}$$

Taking a sum over all five nodes, as before,

$$\left. \frac{dV}{dt} \right|_{t=t_n} = \sum_{i=1}^{5} f_{i-1/2}^n - f_{i+1/2}^n \ ,$$

and again writing out the sum,

$$\left. \frac{dV}{dt} \right|_{t=t_n} = \left(f_{1/2}^n - f_{11/2}^n \right) + \left(f_{11/2}^n - f_{21/2}^n \right) + \left(f_{21/2}^n - f_{31/2}^n \right) + \left(f_{31/2}^n - f_{41/2}^n \right) + \left(f_{41/2}^n - f_{51/2}^n \right) \ .$$

Cancelling out terms reduces to the desired result, which shows that this choice of discretization does strictly enforce the physical requirement that the volume accumulation rate equals the net inflow across the boundaries:

$$\left. \frac{dV}{dt} \right|_{t=t_n} = f_{1/2}^n - f_{51/2}^n \ . \tag{64}$$

Hence, it has been shown that depending on the way in which the $\frac{\partial f}{\partial x}$ term is discretized, a numerical scheme may or may not strictly conserve mass. Of course, being averages, the earlier expressions for the flow rates at the boundaries, $\frac{f_0^n + f_1^n}{2}$ and $\frac{f_5^n + f_6^n}{2}$, would be roughly accurate, and would probably be fine for linear equations. However, it turns out that nonconservative schemes do not propagate shocks at the correct speeds required by equation (37), thus for nonlinear equations it is essential to use conservative schemes which explicitly evaluate fluxes (via some sort of algorithm) *at* the boundaries.

Definition of a Conservative Scheme   As stated in [16], employing a conservative finite volume scheme yields consistency (Section 4.2.1) with the integral form of the conservation law. Formally, if an explicit difference scheme, with independent variables t and x and dependent variable u (which can be a vector), can be written in the form

$$u_i^{n+1} = u_i^n - \frac{\Delta t}{\Delta x}(F_{i-1/2} - F_{i+1/2}) \ , \tag{65}$$

where u_i is the average value of u in cell i, and $F_{i-1/2}$ is some approximation to the flux at the boundary between cells i and $i - 1$, the scheme is said to be conservative[3].

In a well posed, conservative FV scheme, a computational cell interacts with its neighbors not by directly using the pointwise values of u nearby, but by exchanging fluxes across its cell boundaries. For a given boundary between two cells, whatever flux enters one of the cells automatically gets counted as exiting from the other (obviously this is required by conservation). Of course all of the details lie in the non-trivial process of determining the flux at the boundary between two cells, given only the set of cell averages. Hence, explicit conservative FV methods differ from one another only in the manner in which the fluxes at cell boundaries are determined.

4.2.4   Dissipative and Dispersive Errors

The ways in which errors are manifested in solutions generated by approximate methods for solving PDEs vary widely, depending on scheme characteristics. Schemes often attenuate some types of error, while allowing other types of error to propagate freely. One can reliably predict which type of error will dominate based on the leading term in the truncation error; for example, diffusion is usually dominant for first order schemes, and dispersion usually dominant for second order schemes. This rule of thumb arises because of the fact that it can often be shown that a scheme under consideration actually better approximates some PDE *other* than the original PDE

---

[3]F must satisfy the consistency condition that $F(u, u, ..., u) = f(u)$, where f is the exact flux found in the continuous PDE. This requirement implies that inserting the same value "u" for all of the arguments of F must yield a value which is equal to the exact flux f evaluated at u.

for which the scheme was designed! The derivative terms in this "other" PDE (often referred to as the "modified equation") which are not present in the original PDE usually indicate the types of errors that should be expected.

Qualitative examples which demonstrate dissipative and dispersive types of error are presented in Figure 15[4]. In both images, an approximate solution is compared to an exact solution, and equation (51) is simulated. The exact solution is calculated using the method of characteristics. The approximate schemes are conservative finite volume methods. Both the exact and approximate schemes are given identical initial conditions and are allowed to run for 10 engine cycles. The lubricant moves back and forth each revolution, subject to the reciprocating inertia force. The available film thickness provided at the boundaries is simply equal to that film thickness which was uniformly covering the surface initially (not counting the initial ridge). The plots shown depict only the final states of the two schemes, at the very end of the simulation. The parameters are set up so that the viscosity is high enough to keep shock from occurring (i.e., the film moves slow enough that the piston acceleration always changes sign before the film is able to develop a shock), as can be verified from an animation of the characteristics solution.

Dissipation   Loosely speaking, a solution which spreads out over time does so due to dissipation (or, "diffusion"). This feature can be physical - for example, in the heat equation, heat diffuses locally in the opposite direction of the temperature gradient. Diffusion is usually characterized by second derivatives in space, such as $\frac{\partial^2 T}{\partial x^2}$ for the heat equation. For equations such as (22), the derivatives are first order, and no physics which would represent diffusion (other than shock, which is not occurring in this test simulation) is present. Unfortunately, the numerical scheme itself may dissipate the solutions it generates, despite the original PDE being devoid of any physical diffusion. For hyperbolic problems, numerical dissipation is typically the dominant error for first order accurate schemes; the first term in the truncation error for these methods is a second order (i.e. diffusive) derivative. Figure 15a depicts an exceedingly dissipative scheme; the spatial differences used are first order accurate. The cumulative effect of this error can most definitely be severe enough to render a numerical solution inadequate. Obviously, after only 10 engine cycles, the solution has degraded (compared to the exact solution) severely, to the point that one cannot really use it to make predictions of the oil distribution on the top land. With this scheme, it would be impossible to meet the objective described in Section 1.2 that the TLOMM should be robust enough that it can run for hundreds of cycles and still maintain the required accuracy.

---

[4]Quantitative analysis can be performed, but the references cover this topic in depth, so the discussion presented is mainly for illustrative purposes to help motivate the need for the schemes developed in Section 4.4.
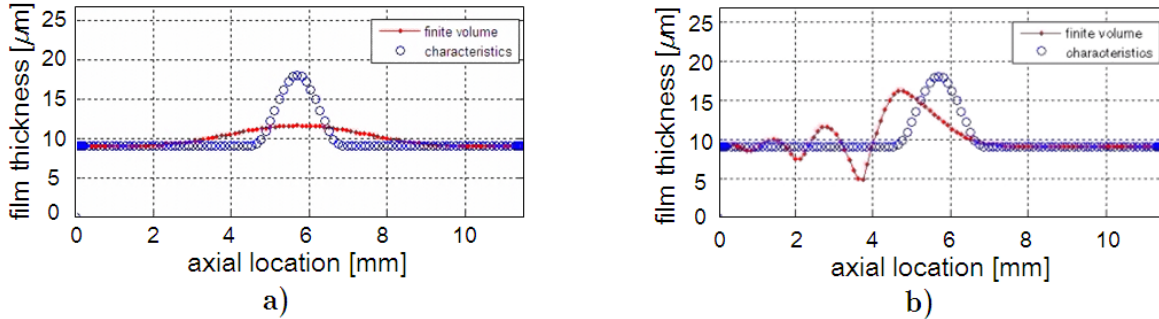
Figure 15: Approximate schemes with excessive numerical errors: a) dominant error is dissipative; b) dominant error is dispersive.

Dispersion   One can think of a solution as being comprised of a series of waveforms (generated using a Fourier series, for example). Dispersion is a phenomenon in which the propagation speed of each of these waves is a function of the wavelength. Hence the higher frequency modes travel throughout the domain at speeds different from the lower frequency modes. Dispersion is usually dominant in PDE's with third derivatives in space, e.g. $\frac{\partial^3}{\partial x^3}$. For hyperbolic problems, numerical dispersion is typically the dominant error for second order accurate schemes; the first term in the truncation error for these methods is a third order derivative. Figure 15b shows an exceedingly dispersive method, for which the spatial differences are second order accurate. Despite this scheme's formally higher order accuracy, its predictions are (again) severely off from the exact solution after only 10 engine cycles.

4.2.5   Decoupling of Discretizations

Equation (65) is written assuming the time derivative in (58) has been replaced with the simplistic "Forward Euler" approximation, $\frac{h_i^{n+1}-h_i^n}{\Delta t}$. However, there is no immediate reason why other choices of time differencing cannot be made, so (65) is more restrictive than necessary. A more flexible equation is the semidiscrete form of (65),

$$\frac{du}{dt} = \frac{1}{\Delta x}(F_{i-1/2} - F_{i+1/2}) \ , \tag{66}$$

which is a coupled set of ordinary differential equations (ODEs). This form makes the choice of time stepping algorithm modular, as its usage allows one to make use of the broad set of techniques developed in the field of ODE's. (66) can perhaps be viewed as the central governing equation used in the TLOMM simulation. The approach of using a semidiscrete form of the discretization scheme, known as the "Method of Lines" [30, 31], effectively decouples the spatial and temporal discretizations.

63

Most of the difficult features of (22) are properly dealt with by focusing on just the spatial discretization and making it satisfy some set of properties. These required properties are developed in the following sections, after briefly saying a few words about time discretization.

4.2.6   Time Stepping

This subsection is a precursor to Sections 4.5 and 4.7.1, in which temporal discretizations are discussed in more detail. The two main classes of "marching," or time stepping, methods, which are used in solving differential equations numerically, must be introduced. The methods by which the two approaches (termed explicit and implicit) advance a solution from time level n to time level $n + 1$ are fundamentally different from one another. Any introductory textbook on numerical approximations to the solutions of differential equations will discuss this topic in greater detail.

Explicit   An explicit method calculates the new solution at any cell i, u_i^{n+1}, using only information from time level n. The method is named as such because the expression used to advance the solution to time $n + 1$ is a simple, closed form (i.e., explicit) algebraic equation for u_i^{n+1}.

Implicit   An implicit method also calculates the new solution at any cell i, but the available information is not restricted only to time level n. Instead, the information at time $n + 1$ itself is used in computing the solution at time $n + 1$. In other words, the cells' values at $n + 1$ are all fully coupled to one another; computing the solution u_i^{n+1} makes use of the "new" (u^{n+1}) values at all of the other cells. The method gets its name from the fact that the resulting expression for u_i^{n+1} is not closed form; instead, it is a set of coupled, implicit algebraic equations. Typically these equations are nonlinear, and must be solved via some sort of iterative rootfinding algorithm designed for sets of nonlinear equations.

4.2.7   Convergence to a Weak Solution

An important result for the schemes considered in this work is the LAX-WENDROFF THEOREM [32]. It states:

> *If* the solution generated by a *conservative* numerical scheme converges (as Δt and Δx go to zero) to some function, then the function to which it converges is a weak solution of the conservation law.

Unfortunately, as was presented in Section 3.4.3, weak solutions to hyperbolic conservations laws are often nonunique. The LW theorem doesn't guarantee that convergence will take place, nor does it claim that the solution obtained will be physically correct. Still, according to this theorem and Sections 3.4.3 and 3.4.4, a conservative, convergent scheme satisfying an entropy condition will in fact generate the unique, correct solution we seek.

4.2.8 Entropy and Monotonicity

It turns out that all *monotone* methods preserve the entropy condition, equation (36). From [8], a monotone scheme is one that may be written in the form

$$u_i^{n+1} = H\left(u_{i-k_L+1}^n, \cdots, u_{i+k_R}^n\right) \ ,$$

with $\frac{\partial H}{\partial u_j^n} \geq 0 \ \forall j$ (H is an increasing function of each of its arguments). Unfortunately, monotone schemes suffer from poor accuracy, as observed empirically and proven by Godunov, as follows.

Godunov's theorem   In 1959, Godunov [33] proved a particularly discouraging result:

> Any monotone, linear scheme can be at most first-order accurate.

Godunov's theorem implies that if only linear schemes are considered, higher order accuracy will come with the price of violating the desired entropy priniciple. As noted in Section 3.4.4, without satisfying the entropy condition, the solution to which the scheme converges will generally not be the unique, physically correct solution. The high resolution schemes developed in Section 4.4, for the purpose of solving hyperbolic conservation laws, achieve high accuracy by exploiting a loophole in Godunov's theorem.

4.2.9 Total Variation

An important property of monotone schemes is that they do not create spurious (non-physical) extrema. In other words, the approximate solutions they create are TVD, so long as the continuous PDE is TVD (defined in Section 3.4.6). Non-TVD schemes, on the other hand, typically produce oscillations in the vicinity of discontinuities. For demonstration, consider equation (51). As long as the coefficient $\frac{a_p}{\nu}$ is not dependent on x, the total variation of solutions to equation (51) can only decrease with time[5].

---

[5]In fact, the solution's TV will stay constant except when shock occurs), which can be shown using characteristics (see Section 3.4.1).
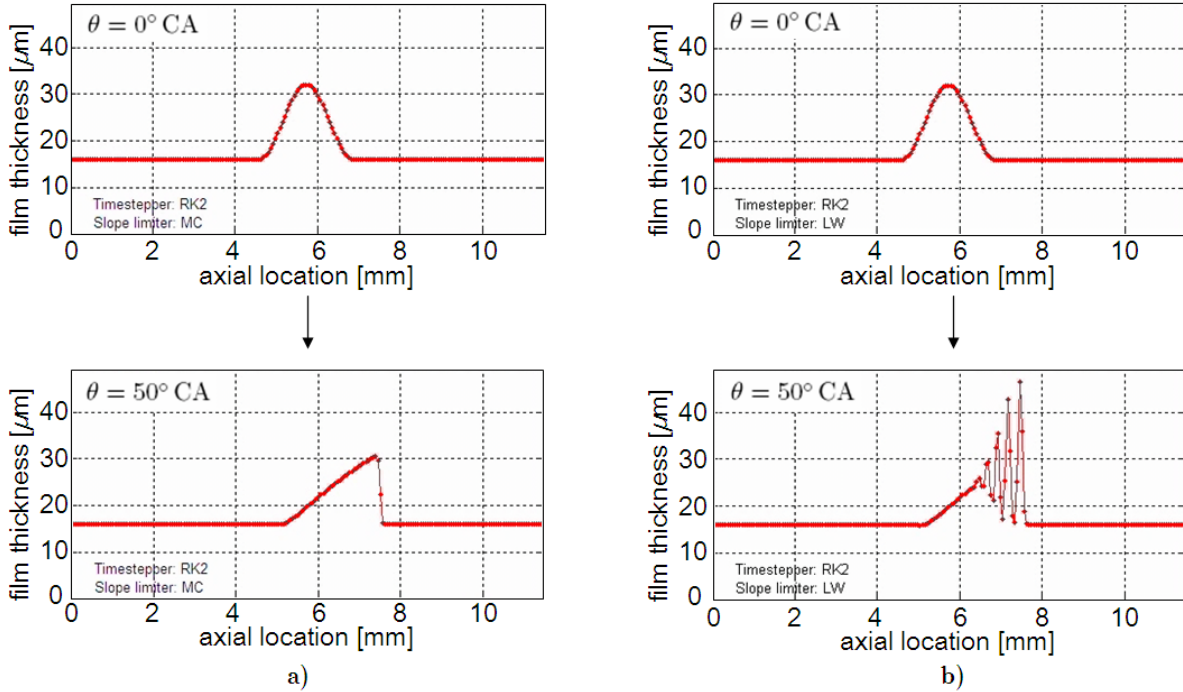
Figure 16: Solution of equation (51) for a) a TVD (shock capturing) scheme, and b) a non-TVD (Lax-Wendroff) scheme.

Figure 16 demonstrates the very different manner in which two representative schemes approximate the solution to equation (51), showing snapshots both at the (identical) initial condition and after some finite time (50 crank angle degrees) has passed. One scheme is TVD, the other is not. Unlike the simulations in Figure 15, a realistic viscosity was used and shocks occurred.

Once shock develops, the non-TVD scheme (which is formally second order accurate) behaves extremely poorly[6]. Obviously the total variation of the solution generated by the TVD scheme (Figure 16a) at $\theta = 50°$ is less than the TV of the initial condition, consistent with the TVD property inherent to the initial PDE. On the contrary, the total variation of the solution due to the non-TVD scheme (Figure 16b) at $\theta = 50°$ is certainly greater than the TV of the initial condition. Hence the non-TVD scheme does not produce solutions that maintain the total variation property of the initial PDE.

The wild oscillations exhibited in Figure 16b are undesirable for two main reasons:

- They are not physically correct, and are due entirely to weaknesses of the numerical scheme.

---

[6]One can imagine how the dispersive errors of this scheme get out of control due to the Gibbs phenomena which arise in attempting to approximate a discontinuity.

- They can easily make an otherwise stable solution unstable. From Section 3.4.1, it was shown that the characteristics move with a wave speed proportional to h^2. As seen in Figure 16, the oscillations cause some of the grid points to take on much higher values of h, where the wave speed is much larger. It is not surprising that this chain of events introduces stability issues when one recalls the typical requirement, equation (57), that the Courant number must remain less than or equal to some constant, C_{max} (often equals 1), as introduced in Section 4.2.1.

Formal Definition of TVD Schemes Consider a class of numerical schemes that can be written in the form

$$u_i^{n+1} = u_i^n - C_{i-1/2}\left(u_i^n - u_{i-1}^n\right) + D_{i+1/2}\left(u_{i+1} - u_i\right) \ , \tag{67}$$

where the coefficients C and D may, in general, depend on the data u (i.e., the scheme itself is allowed to be nonlinear). It can be shown[7] that this scheme, applied to a PDE with constant coefficients (of the form (67)), is TVD provided that the coefficients satisfy

$$\begin{aligned} C_{i+1/2} &\geq 0 \\ D_{i+1/2} &\geq 0 \\ 0 \leq C_{i+1/2} + D_{i+1/2} &\leq 1 \end{aligned} \tag{68}$$

This set of conditions is often referred to as "the TVD test."

Stability When applied to actual schemes, the third inequality in (68) actually imposes a stability-like condition which is dubiously similar to equation (57). For example (from [16]), for the *linear* advection equation,

$$\frac{\partial h}{\partial t} + a\frac{\partial h}{\partial x} = 0 \ ,$$

with a (the advection speed) being a positive constant, application of the "upwind scheme" would give

$$u_j^{n+1} = u_j^n - \frac{a\Delta t}{\Delta x}\left(u_j^n - u_{j-1}^n\right) \ .$$

Enforcing the conditions (68) leads to

$$C_{j-1/2} = \frac{a\Delta t}{\Delta x} \leq 1 \ , \tag{69}$$

which obviously is a constraint analogous to the common requirement arising in linear stability analysis, equation (57). In addition to prohibiting numerical oscillations, all explicit TVD

---

[7]HARTEN'S THEOREM, see [1, 8].

schemes which meet their CFL-type requirements such as (69) are stable (though not vice versa). This feature should not be surprising - total variation is, in a way, a measure of the "size" of the data making up a solution. Stability requires that errors are not amplified with time; similarly, a TVD requirement forces the data itself to either spread out (decay) or stay constant with time. In actuality, a scheme meeting the TVD constraint actually meets a more stringent brand of stability ("nonlinear stability"). Thus if one uses a scheme which is known to be TVD, stability is not an issue as long as the "CFL-like" TVD requirement, such as (69) for the upwind scheme, is met.

It is interesting to note that schemes have been developed which do not strictly insist that the total variation of the solution to diminish. Instead, they allow the TV to increase here and there by small amounts, but they enforce that the total variation remain bounded. These TVB schemes, though perhaps not as widely used as TVD schemes, have the property of *uniform* high order accuracy[8]. Shu presents a TVB modification in [34]. The ENO/WENO methods that will be discussed in Section 4.7 generalize this approach.

4.2.10   Summary

This section has presented many numerical concepts without much detail, let alone rigor. For a typical user of the TLOMM simulation, Section 4.2 is hopefully complete enough to provide a rough background on the issues faced when designing numerical methods for hyperbolic conservation laws.

According to Godunov's theorem, it may seem that one must make a choice between a low-accuracy (first order) monotone scheme and a high accuracy (higher order) non-monotone scheme which suffers from severe oscillations. Fortunately, as will be shown in Section 4.4, schemes which are both high order and nonoscillatory can in fact be constructed. This achievement takes place by making the coefficients C and D of equation (67) nonlinear functions of the data u itself, which allows Godunov's theorem to be circumvented.

Classification of the various methods   There are quite a few classes of conservative finite volume methods. As discussed in this section, properties of schemes include order of accuracy, monotonicity, total variation consistency, entropy satisfaction, etc. Understanding the underlying genealogy of these methods (e.g. which schemes are subsets of other schemes) is, itself, a daunting task. Figure 17, adapted from [16], makes some attempt to decode these relationships. One can see, for example, that all linear TVD methods ("TVD (L)") are necessarily first order, but this

---

[8]Formally, all TVD schemes' spatial accuracies degenerate to first order in the vicinity of extrema, see Section 4.4.4.
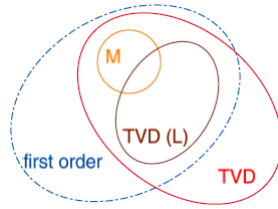
Figure 17: Venn diagram of various schemes, adapted from [16].

restriction does not apply to nonlinear TVD methods. "M" here denotes the class of monotone schemes.

Of course, nothing has been said yet as to how these schemes are actually constructed in practice. The following section gets one step closer to developing the powerful high resolution scheme(s) used by the TLOMM.

4.3   Scheme Requirements

Section 4.2 should surely have motivated the notion that the scheme used in the TLOMM simulation should meet some minimum set of requirements. This brief section simply organizes these requirements into a concise and coherent list. From the presentations of Sections 1.2 and 4.2, the numerical method employed to solve equation (51) (and eventually (22)) should meet the following criteria:

1. ACCURACY
   The numerical dissipation and dispersion errors inherent in the scheme must be minimized to the point that their cumulative effects, even after hundreds of engine cycles (i.e. hundreds of thousands of time steps), are so small that they have practically no effect on the TLOMM's ability to accurately predict the distribution of lubricant on the top land. Of course their severity will be a function of the spatial and temporal resolutions chosen by the user.

2. CONSISTENCY
   As for all PDE simulations, the numerical scheme must be consistent with the original PDE, according to the definition (55).

3. STABILITY
   As for all PDE simulations, the numerical scheme must be stable.

4. CONSERVATION
   Due to the fact that methods which are not strictly conservative often propagate information at incorrect speeds, the numerical scheme must be strictly conservative.

5. ENTROPY

   The scheme must satisfy the entropy condition, so that the weak solution to which it converges is the physically correct solution.

6. TOTAL VARIATION

   The scheme must produce solutions whose total variation is (at least) bounded[9], in order to generate physically correct predictions and to meet the stability requirements of nonlinear conservation laws. This requirement does make requirement number 3 redundant, but both are listed in order to emphasize that not only must the scheme satisfy the classical stability requirement, it must possess a more restrictive type of stability and match the total variation property of the original PDE.

7. SHOCKS

   The numerical scheme must be able to handle a general, arbitrarily curved shock front in the $x - y$ plane, which must be allowed to occur naturally without having to provide the scheme with extra information *a priori*.

8. GRID

   The TLOMM should possess a computational grid which does not introduce unnecessary complications. As the simulation is intended to be augmented with additional physics by future workers, its grid should allow for straightforward incorporation of additional models. Essentially, an *Eulerian* grid with Cartesian coordinates is required.

Reasoning   Items 2 and 3 are common to all PDE simulations, as discussed in Section 4.2.1. Requirements 4 through 7 are specific to the *type* of PDE (see Classification, Section 3.3) being solved in this work. Finally, requirements 1 and 8 are specific the TLOMM in particular (according the project objectives, Section 1.2).

4.4   Shock Capturing

It was the discovery (usually credited to Harten [1]) of a class of higher order schemes which maintain the TVD property that ushered in an explosive period of development in the CFD field back in the 1980's. Numerical methods meeting the requirements outlined in Section 4.3 do in fact exist. These finite volume approaches, usually termed "high resolution schemes," revolutionized several areas of PDE numerical methods during the late 1970's and early 80's. Some of the

---

[9]As was mentioned in Section 3.4.1, solutions to equation (22) itself do *not* necessarily have a diminishing total variation, since gradients in the gas flows show up as source terms (i.e. the coefficients of equation (22) are variable). To be more precise, what is essential is that any increase in the total variation of the approximate solution calculated by the TLOMM simulation be *physically* induced, and not be caused by the numerical scheme.

many approaches ([35]) for resolving flows with shock waves prior to the introduction of these schemes included 1) using a first order Godunov method, 2) using a second order scheme to solve a modified governing equation, to which an artificial viscosity term is explicitly added, and 3) applying an *anti*diffusive term to a known first order accurate scheme. The development of high resolution methods can more or less be followed from the following central papers: [35] (review), [25], [1], [37] (review), [36], and the references contained therein.

High resolution schemes designed for hyperbolic conservation laws which admit discontinuous solutions are generally grouped into the categories of "shock capturing" or "shock fitting" (sometimes referred to as "front tracking"). Only the class of shock capturing schemes will be discussed in this section; Section 4.7.3 addresses the reasons behind avoiding shock fitting.

Shock capturing schemes are finite volume numerical algorithms which allow discontinuities (shocks) to occur without creating spurious oscillations in the solution. They are conservative, according to the definition (66). Additionally, the methods are better than first order accurate. Typically, the solutions they create are second order accurate throughout most of the domain. An unfortunate reality of all TVD methods is that they degenerate to first order accuracy in the immediate vicinity of extrema, as discussed in Section 4.4.4.

There are several classes of modern shock capturing schemes. Perhaps the two most widespread are the flux limiting and slope limiting approaches; both were applied to equation (51) in this work. The reader is reminded that a 2D simulation of equation (51) meeting the requirements of Section 4.3 was first developed, and then the simulation was extended to three independent variables by adding the circumferential direction. Ultimately, slope limiting was chosen as the final method, for reasons which will become clear in Section 5.1.3.

Some schemes do achieve *uniformly* high order accuracy, by slightly relaxing the TVD constraint. These (newer) shock capturing methods, such as Discontinuous Galerkin and ENO/WENO, are gaining in popularity and are discussed briefly in Section 4.7. Despite the wide variety of choices that exists, shock capturing schemes all share the common theme of *limiting*.

4.4.1   Limiting

Shock capturing methods involve discretization schemes which are, out of the necessity to avoid the limitations of Godunov's theorem, nonlinear. That is, the exact form of a given shock capturing scheme (itself) varies from cell to cell, depending on the values of the dependent variable u. Schemes implement this feature via the process of limiting.

As was seen in Figure 15b, unacceptable oscillations can result from using a standard higher order accurate scheme which is not TVD, while no oscillations whatsoever (but extreme amounts of

dissipative error) can result from using a basic first order scheme[10]. *Limiting* is a generic name for an approach made by shock capturing algorithms, which essentially constructs some sort of weighted compromise between the two extremes. The methods available limit various aspects of the scheme; for example, the magnitudes of the fluxes (Section 4.4.2), the slopes of some sort of polynomial reconstruction within each cell (Section 4.4.3), and the magnitudes of the waves being propagated (see Section 4.7.2) all have been limited in various schemes.

4.4.2   Flux Limiting

Classical higher order schemes often violate the TVD property of the governing PDE by overestimating the magnitude of the fluxes being exchanged between computational cells. Hence one standard shock capturing line of attack is to limit the fluxes directly.

In this approach (following [7]), the numerical flux $F_{i+1/2}$ at the boundary between cells i and $i+1$, at time level n, is interpreted as a weighted combination of low and high resolution fluxes, F_L and F_H:

$$F_{i+1/2} = F_L\left(u_i^n, u_{i+1}^n\right) + \phi_{i+1/2}^n\left[F_H\left(u_i^n, u_{i+1}^n\right) - F_L\left(u_i^n, u_{i+1}^n\right)\right]\ ,\tag{70}$$

where u_i is the average value of u within cell i and ϕ is a flux limiter, to be defined. Obviously if $\phi = 0$, the flux used by the scheme is the low resolution flux, whereas if $\phi = 1$, the scheme uses the high resolution flux. The high resolution flux chosen is one which typically works well for smooth data, but cannot handle discontinuities, such as the Lax-Wendroff flux:

$$F_{i+1/2,\ \mathrm{LW}} = \frac{1}{2}\left(f\left(u_i^n\right) + f\left(u_{i+1}^n\right)\right) - \frac{\Delta t}{2\Delta x}\left\{\frac{\left[f\left(u_{i+1}^n\right) - f\left(u_i^n\right)\right]^2}{u_{i+1}^n - u_i^n}\right\}\ .\tag{71}$$

Several variations on equation (71) exist; the form shown here (from [7]) is general enough for use with nonlinear equations such as (51), and is employed in the 2D flux limiting version of the TLOMM.

One must also choose a low resolution flux, which usually is known to have poor accuracy in smooth regions but handles discontinuities without issue (due to being monotone). A common example is the Upwind flux:

$$F_{i+1/2,\mathrm{UW}} = \begin{cases} \min_{u_i \leq u \leq u_{i+1}}\left(f\left(u\right)\right) & \text{if } u_i \leq u_{i+1} \\ \max_{u_{i+1} \leq u \leq u_i}\left(f\left(u\right)\right) & \text{if } u_{i+1} \leq u_i \end{cases}\tag{72}$$

---

[10]As was demonstrated in Figure 15.

Name	Expression	Ref
Upwind	$\phi = 0$	[6]
Lax-Wendroff	$\phi = 1$	[6]
Beam-Warming	$\phi = \theta$	[6]
Fromm	$\phi = \frac{1}{2}(1 + \theta)$	[6]
Minmod	$\phi = \mathrm{minmod}(1, \theta)$	[26]
Superbee	$\phi = \max(0, \min(1, 2\theta), \min(2, \theta))$	[26]
Monotonized Centered	$\phi = \max\left(0, \min\left(\frac{1+\theta}{2}, 2, 2\theta\right)\right)$	[25]
Van Leer	$\phi = \frac{\theta + |\theta|}{1 + |\theta|}$	[24]

Table 2: Common flux limiters, adapted from [6].

The upwind flux gets its name from the feature that it ensures that a flux evaluation only uses information which is "upwind" of the location at which the flux is required. For example, if the local flow is taking place from left to right, the upwind flux uses only information from the left.

The flux limiter, ϕ, is typically chosen to be a function of the local smoothness. Before ϕ can be defined, a local smoothness monitor, θ, must be introduced (following [6]):

$$\theta^n_{i-1/2} = \frac{\Delta u_{i^*-1/2}}{\Delta u_{i-1/2}} \ , \tag{73}$$

where $\Delta u_{i-1/2} = u_i - u_{i-1}$ and $i^* = i - \mathrm{sign}(C)$, where C is the local Courant number ($C = v\frac{\Delta t}{\Delta x}$, equation (56)).

Common flux limiters   Many different flux limiter functions have been constructed. Table 2, adapted from [6], shows some of the most common choices. The first four methods listed are classical linear methods which are subject to Godunov's theorem. Upwind is formally first order accurate, while the Lax-Wendroff, Beam-Warming, and Fromm[11] schemes are second order accurate. The last four in the table are *non*linear methods: high resolution schemes which are TVD and (almost uniformly) high order accurate. For these, references to the original paper in which they were proposed are given.

Sweby diagram   In [36], Sweby showed that on a plot of $\phi(\theta)$ vs. θ, there exists a region in which all values of flux limiter functions must fall, in order for the resulting scheme to be both TVD and second order accurate. One version of this "Sweby Diagram," along with three of the high resolution limiters, is shown in Figure 18 (adapted from [16]). The shaded area on this plot is the 2nd order, TVD region. One will note that none of the top four limiter functions in Table

---

[11]Fromm's scheme is just an average of LW and BW.
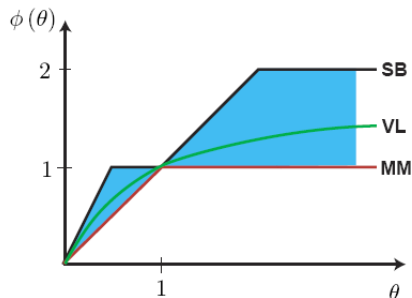
Figure 18: Sweby diagram, with several popular flux limiters, adapted from [16].

2 would fall within the second order, TVD region for all θ, while the bottom four all remain within this region.

4.4.3 Slope Limiting

Rather than attempting to achieve better accuracy by focusing solely on the fluxes between the cells, higher accuracy can also be attained by applying some sort of reconstruction algorithm to the set of cell averages. This reconstruction could, say, attempt to describe the variation of the dependent variable *within* each cell, obtaining values of the solution at the boundaries between cells. Most commonly, the reconstruction approach takes piecewise constant data (cell averages) as input and outputs a piecewise polynomial function[12]. One could then apply some limiting procedure to the reconstruction, to ensure that it is TVD.

This basic sketch of a numerical method, involving functional reconstruction of the data within the cells, is the foundation of the slope limiting method. In this work, piecewise linear functions were chosen to describe the sub-cell variation, since this practice results in sufficient accuracy, but higher order polynomial reconstructions have been performed as well (e.g. Piecewise Parabolic method in [38]). Figure 19 depicts the main idea. Several cells are shown, with arbitrary values of the cell averages (piecewise constant) shown in a solid line. The dotted line represents a linear reconstruction operation that has been performed on the cell averages.

Letting σ_i denote the slope within cell i, the reconstructed values of u at the left and right boundaries of cell i are $u_i - \frac{\Delta x}{2}\sigma_i$ and $u_i + \frac{\Delta x}{2}\sigma_i$, respectively. At a typical cell boundary, say $i - 1/2$, there are in general two values (since the reconstruction is only piecewise continuous) - one from cell i and one from cell $i - 1$. Defining both values at both boundaries of cell i is

---

[12]Of course the number of "pieces" to the function is simply the number of cells.
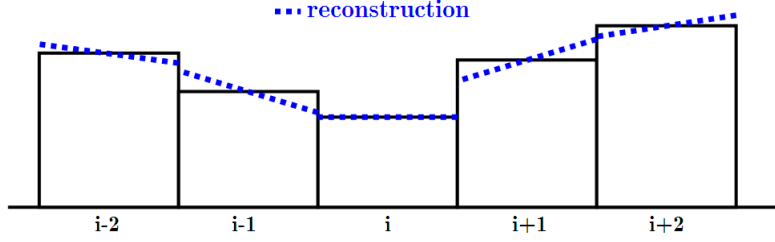
Figure 19: Cell averages with linear reconstruction performed.

Name	Expression	Ref
Upwind	$\sigma_i = 0$	[6]
Lax-Wendroff	$\sigma_i = \frac{Q_{i+1}-Q_i}{\Delta x}$	[6]
Beam-Warming	$\sigma_i = \frac{Q_i-Q_{i-1}}{\Delta x}$	[6]
Fromm	$\sigma_i = \frac{Q_{i+1}-Q_{i-1}}{2\Delta x}$	[6]
Minmod	$\sigma_i = \text{minmod}\left(\frac{Q_i-Q_{i-1}}{\Delta x}, \frac{Q_{i+1}-Q_i}{\Delta x}\right)$	[6]
Superbee	$\sigma_i = \text{maxmod}\left(\sigma_i^{(1)}, \sigma_i^{(2)}\right)$ where $\sigma_i^{(1)} = \text{minmod}\left(\frac{Q_{i+1}-Q_i}{\Delta x}, 2\frac{Q_i-Q_{i-1}}{\Delta x}\right)$ $\sigma_i^{(2)} = \text{minmod}\left(2\frac{Q_{i+1}-Q_i}{\Delta x}, \frac{Q_i-Q_{i-1}}{\Delta x}\right)$	[6]
Monotonized Centered	$\sigma_i = \frac{1}{\Delta x}\begin{cases} \min\left(2\left|Q_i-Q_{i-1}\right|, \frac{1}{2}\left|Q_{i+1}-Q_{i-1}\right|, 2\left|Q_{i+1}-Q_i\right|\right) \\ \quad \text{if } \text{sign}\left(Q_i-Q_{i-1}\right) = \text{sign}\left(Q_{i+1}-Q_i\right) = \text{sign}\left(Q_{i+1}-Q_{i-1}\right) \\ 0 \quad \text{otherwise} \end{cases}$	[25]
Van Leer	$\sigma_i = \frac{1}{\Delta x}\begin{cases} \frac{2(Q_i-Q_{i-1})(Q_{i+1}-Q_i)}{Q_{i+1}-Q_{i-1}} & \text{if } \text{sign}\left(Q_i-Q_{i-1}\right) = \text{sign}\left(Q_{i+1}-Q_i\right) \\ 0 & \text{otherwise} \end{cases}$	[23]

Table 3: Common Slope Limiters.

straightforward:

$$\begin{aligned}
u_{i-1/2}^{L} &= u_{i-1} + \frac{\Delta x}{2}\sigma_{i-1} \\
u_{i-1/2}^{R} &= u_i - \frac{\Delta x}{2}\sigma_i \\
u_{i+1/2}^{L} &= u_i + \frac{\Delta x}{2}\sigma_i \\
u_{i+1/2}^{R} &= u_{i+1} - \frac{\Delta x}{2}\sigma_{i+1}
\end{aligned} \tag{74}$$

where superscripts L and R denote the value on the left and right "side" of each boundary, respectively. As there is for flux limiting, there exists an array of choices of slopes. Table 3 highlights some of the common slope functions. The top four are the slopes associated with classical, linear schemes which either have poor accuracy or violate the TVD requirement. The bottom four reconstructions use limiting; as such, they are TVD and second order accurate (except near extrema). All of the slopes in Table 3 are analogous, and closely related, to the flux limiters listed in Table 2 by the same names.

Riemann Solver   As with all conservative finite volume schemes, the flux at all cell interfaces must be computed. With a reconstruction algorithm, the common way to determine the interface fluxes is via a Riemann solver, but first the notion of a Riemann problem (RP) must be introduced. An RP is simply the problem of solving the PDE itself on an infinite domain x, with the special initial conditions of a jump discontinuity at $x = 0$:

$$u\left(x, t = 0\right) = \begin{cases} u_{0_L} & \text{for } x < 0 \\ u_{0_R} & \text{for } x > 0 \end{cases} \tag{75}$$

A Riemann *solver* is simply some numerical technique which approximately solves the Riemann problem (i.e., determines $u\left(x\right)$ for all time). It is essentially the core calculating engine of a slope limiting simulation, for it calculates all of the fluxes at all of the cell boundaries, at each time step. For a scalar PDE, there are only a few possibilities; the solution could be a shock, a contact discontinuity, or a rarefaction wave. Using the notation of [6], let $u^{\downarrow}\left(u_{0_L}, u_{0_R}\right)$ denote the value of u at $x = 0$ and $t = 0^+$, as calculated by a Riemann solver. $t = 0^+$ is used simply so that an infinitesimally small time has passed and has allowed the solution u at $x = 0$ to become single valued (if the solution is a shock wave, the discontinuity will propagate either to the left or right; if it is a rarefaction wave, the discontinuous initial condition will immediately become continuous).

A simulation doesn't really need *all* of the data calculated by a Riemann solver; it only needs the instantaneous flux at the boundary, $f\left(u^{\downarrow}\left(u_{0_L}, u_{0_R}\right)\right)$, in order to evolve the cell averages using equation (66). This flux may be evaluated explicitly, using Osher's closed form expression [40]

$$f\left(u^{\downarrow}\left(u_{0_L}, u_{0_R}\right)\right) = \begin{cases} \min_{u_{0_L} \leq u \leq u_{0_R}}\left(f\left(u\right)\right) & \text{if } u_{0_L} \leq u_{0_R} \\ \max_{u_{0_R} \leq u \leq u_{0_L}}\left(f\left(u\right)\right) & \text{if } u_{0_R} \leq u_{0_L} \end{cases} \tag{76}$$

which, importantly, is valid for nonconvex fluxes in addition to convex fluxes[13] (see below). Of course, the striking similarity between this equation and (72) is not coincidental; what makes equation (76) correct is that it uses correctly upwinded information to calculate the instantaneous flux.

Luckily for this work, as the TLOMM involves solving a scalar PDE, the Riemann solver is not difficult to derive or nor is the solution all that costly to compute. However, for the case where u is a vector, and the governing PDE is a *system* of conservation laws, finding the solution to the Riemann problem is generally not trivial; it involves a costly eigenvalue decomposition which transforms the system into its characteristic variables. To avoid the prohibitively high cost of solving a Riemann problem for a vector PDE, workers often use "approximate" Riemann solvers

---

[13]Though it is only applicable to scalar equations.

(it is not essential that the solution to the Riemann problem be exactly correct if the rest of the discretization scheme has inherent errors anyway).

Flux convexity A flux function f is termed "convex" if the sign of its second derivative with respect to the dependent variable u, $\frac{\partial^2 f}{\partial u^2}$, is constant. A convex flux is simpler to work with than an arbitrary (nonconvex) flux because for the former, the wave (or "characteristic") speed varies monotonically with respect to u. Unfortunately, the axial flux in equation (22),

$$f = -\frac{a_p}{3\nu}h^3 + \frac{1}{2}\frac{\mu_{gas}}{\mu_{oil}} \left.\frac{\partial u_{gas}}{\partial z}\right|_{z=h} h^2 \quad , \tag{77}$$

is *non*convex. There are two terms in this expression, and they scale with different powers of h. One can imagine a case where the inertia and gas flow forces act in opposite directions. For small enough values of h, the gas flow effect would be dominant and waves (characteristics) would travel in the direction imposed by the gas flow force. For large values of h, the inertia effect would be dominant and characteristics would propagate in the direction of the inertia force. Thirouard [3] noted this fundamental possibility from the different h scalings present in equation (77), without need for any of the analysis in this section. Of course the flux need not always be nonconvex; for example, whenever these two axial forces act in the same direction, the flux is convex and all is well. However, with only a minimal amount of effort, the author was able to identify at least one occurrence within an engine cycle, using some contrived (abnormally strong) gas flows, of the flux function at a point becoming nonconvex. Figure 20 demonstrates this situation - the direction of the flux depends on the magnitude of h^{14}. The fluxes and film thicknesses are plotted on dimensionless axes because the internal workings of the TLOMM simulation use only dimensionless variables, as is standard practice.

Osher [40] derived the Riemann solver, (76), which correctly captures the behavior of a general nonconvex flux for a scalar PDE. To be absolutely correct, the Riemann solver implemented in the TLOMM should take the min and max functions indicated in equation (76) literally; e.g. it should seek the minimum value of f over the range $u_{0_L} \leq u \leq u_{0_R}$ (for the first case in equation (76)) using some sort of minimization algorithm. Unfortunately, due to the desire to run the simulation for hundreds of engine cycles, it was decided to simply consider the end points rather than implement a true, yet potentially costly, minimization (or maximization, in the case $u_{0_R} \leq u_{0_L}$) routine.

Commonalities and differences between slope limiting and flux limiting Flux limiting and slope limiting are very similar concepts. Seeing in Tables 2 and 3 that there are slope limiters

---

[14] An additional consequence of nonconvexity here is that it means the governing equation is not strictly TVD, even when there are no gas velocity gradients.
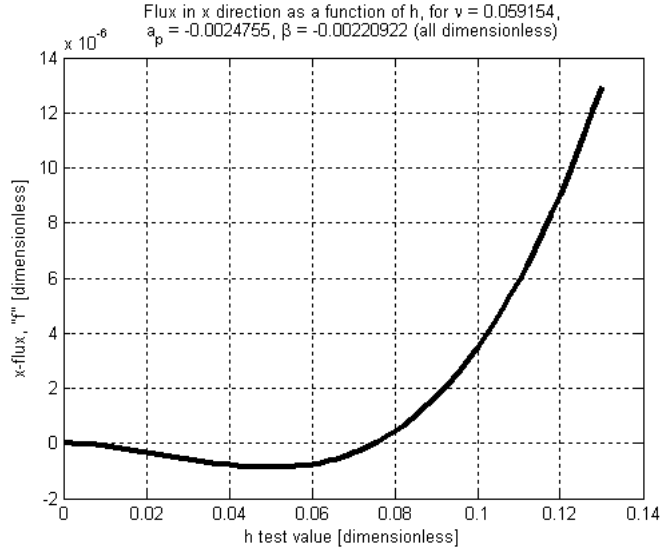
Figure 20: Example of the axial flux becoming nonconvex when the inertia and gas flow forces oppose one another.

and flux limiters that go by the same names, one may wonder if the two approaches actually yield identical results and differ in interpretation only. It turns out that they are in fact identical for scalar, linear equations. However, though analogous [8], the two approaches are fundamentally different, and they produce different results for general nonlinear equations[15]. Only slope limiting fits "into the Godunov-type framework" ([23]).

The two approaches place different constraints on the available temporal discretization choices. Flux limiting discretizes both time and space; "the numerical flux function...plays the role of an average flux through $x_{i+1/2}$ over the time interval $[t_n, t_{n+1}]$" [7]. Slope limiting, however, produces an instantaneous flux at the beginning of the time interval, which in turn translates to the availability of an instantaneous derivative. Slope limiting only dictates the spatial discretization, and a scheme designer is given the flexibility of using the Method of Lines for the temporal discretization.

4.4.4 Degeneration to First Order Accuracy

Considering the definition of the smoothness monitor θ (equation (73)), the Sweby diagram (Figure 18), and equation (70), one can see why all TVD flux limiting schemes degenerate to first order accuracy in the immediate vicinity of extrema (including *smooth* extrema). At an extremum, $\theta < 0$ because the first derivative in u changes sign. The Sweby diagram does not

---

[15]The empirical error analyses presented in Section 5.1 largely confirms this notion.

show it, but to have the TVD property, a requirement is that $\phi = 0$ if $\theta < 0$. Slope limiting schemes are of course formulated differently, and do not rely explicitly on a smoothness monitor, but they too suffer from this limitation as a direct implication of being TVD. Still, despite this local loss of accuracy, in practice the error typically stays confined to a small region. As a result, both flux limiting and slope limiting methods have met with a great deal of success, and are sufficiently accurate for most purposes.

4.5 Temporal Discretization

The method of lines presented in Section 4.2.5 allows the tasks of spatial and temporal discretization of the governing PDE to be uncoupled from one another. In addition to being presented with many choices for spatial discretization, when the method of lines can be used, one must choose a time stepping algorithm. As was discussed in Section 4.4.3, the flux limiting method imposes both choices of discretizations and does not allow them to be uncoupled. Hence, this discussion is really applicable only to algorithms such as slope limiting, which provides instantaneous time derivatives and does not automatically specify the time stepper.

One may of course choose to use a simple, first order accurate time discretization. When all that is seeked is a steady state solution, first order accuracy in time may well be sufficient enough. However, for time dependent problems, if the time accuracy is still only first order, the improvement in spatial accuracy obtained by using the slope limiting scheme discussed in Section 4.4.3 is usually not advantageous [7]. Since the TLOMM simulates a highly time dependent situation, it would make sense to consider augmenting the slope limiting method with a high order time discretization.

Although decoupling of the spatial and temporal discretizations produces a set of ODE's, one cannot quite use any of the hundreds of ODE solvers available in ODE literature. If the TVD requirement is to be strictly enforced, only certain ODE timesteppers may be used. These so-called "TVD time discretizations," discovered by Shu in [41], are typically variants on the well known Runge-Kutta algorithms. It was demonstrated "with numerical examples that non-TVD but linearly stable Runge-Kutta time discretizations can generate oscillations even for TVD spatial discretizations" [42]. Several 2nd through 5th order Runge-Kutta time discretizations, modified to ensure the TVD property, are presented in [46]. In the TLOMM, the time discretization is modularized; one has a choice of the one, two, or three stage TVD Runge-Kutta algorithms given in [46] (the number of stages is typically the expected order of accuracy). Writing equation (66) as $\frac{du}{dt}\big|_{t=t_n} = \mathcal{L}\left(u^n, t^n\right)$), with $\mathcal{L}$ being the spatial discretization operator, the TVD formulae are as follows.

- RK1 (Forward Euler):
$$u^{n+1} = u^n + \Delta t \mathcal{L}\left(u^n, t^n\right) \qquad (78)$$

- RK2 (Heun's Method):
$$
\begin{aligned}
u^{(1)} &= u^n + \Delta t \mathcal{L}\left(u^n, t^n\right) \\
u^{n+1} &= u^n + \tfrac{1}{2}\Delta t \left[\mathcal{L}\left(u^n, t^n\right) + \mathcal{L}\left(u^{(1)}, t^n + \Delta t\right)\right]
\end{aligned}
\qquad (79)
$$

- RK3:

$$
\begin{aligned}
u^{(1)} &= u^n + \Delta t \mathcal{L}\left(u^n, t^n\right) \\
u^{(2)} &= u^n + \tfrac{1}{4}\Delta t \left[\mathcal{L}\left(u^n, t^n\right) + \mathcal{L}\left(u^{(1)}, t^n + \Delta t\right)\right] \\
u^{n+1} &= u^n + \tfrac{1}{6}\Delta t \left[\mathcal{L}\left(u^n, t^n\right) + \mathcal{L}\left(u^{(1)}, t^n + \Delta t\right) + 4\mathcal{L}\left(u^{(2)}, t^n + \tfrac{\Delta t}{2}\right)\right]
\end{aligned}
\qquad (80)
$$

Thus far the discussion has been confined solely to explicit schemes. The reasoning for avoiding implicit schemes is explained in Section 4.7.1.

At this point, two complete numerical methods for simulating equation (51) have been specified. One is a flux limiting approach and requires equations (65), (70), (71), (72), and (73), as well as Table 2 and lower level expressions for material properties, etc. The second is a slope limiting approach and requires equations (66), (74), (76), and (78) or (79) or (80), as well as Table 3 and the same lower level expressions as the former scheme. Predictions of the two schemes are compared to each other (to some extent) in Sections 5.1.2 and 5.1.3.

4.6   Extension to Three Independent Variables

Until this section, essentially all discussions have focused exclusively on numerical methods for solving equation (51), a simplified version of the full governing equation, (22). It was decided to use the slope limiting method described in the previous sections for the final version of the TLOMM. The TVD slope limiting algorithm, equipped with higher order time stepping, generates time dependent solutions on a one dimensional (x, the axial direction) domain only. The remaining algorithmic ingredient required is an extension to a two dimensional domain, bringing the number of independent variables to three. It turns out that most shock capturing schemes, especially of the type used in the TLOMM, can be readily generalized to higher dimensions.

Before delving into the multidimensional schemes available, an important (though discouraging) result must be presented. The GOODMAN & LEVEQUE THEOREM, which was proven in 1985 [4], states that any conservative, TVD scheme for solving scalar conservation laws in two space

dimensions is at most first order accurate. Despite this development, many workers have found in practice that high resolution schemes work very well in two space dimensions[16].

Choices   When adding more dimensions to the base scheme, one is confronted with several choices. There are essentially two main multidimensional algorithms.

- Split Method

  The dimensional splitting approach splits the task of evolving the solution from time n to time $n + 1$ into several one dimensional "sweeps." Each sweep simply applies the base scheme in one of the spatial directions. There are two often used approaches, Godunov splitting and Strang splitting. The schemes are very straightforward: from [6], Godunov's (first order accurate) approach is

$$
\begin{array}{rcl}
u_{ij}^* & = & u_{ij}^n - \frac{\Delta t}{\Delta x}\left(F_{i+1/2,j}^n - F_{i-1/2,j}^n\right) \\
u_{ij}^{n+1} & = & u_{ij}^* - \frac{\Delta t}{\Delta x}\left(G_{i,j+1/2}^* - G_{i,j-1/2}^*\right)
\end{array}
\tag{81}
$$

  where the notation of equation (23) has been followed; G is the approximate (or "numerical") flux in the circumferential direction, just as F has been the approximate axial flux. Strang's method, which is formally second order accurate ([56]), is

$$
\begin{array}{rcl}
u_{ij}^* & = & u_{ij}^n - \frac{\Delta t}{2\Delta x}\left(F_{i+1/2,j}^n - F_{i-1/2,j}^n\right) \\
u_{ij}^{**} & = & u_{ij}^* - \frac{\Delta t}{\Delta x}\left(G_{i,j+1/2}^* - G_{i,j-1/2}^*\right) \\
u_{ij}^{n+1} & = & u_{ij}^n - \frac{\Delta t}{2\Delta x}\left(F_{i+1/2,j}^{**} - F_{i-1/2,j}^{**}\right)
\end{array}
\tag{82}
$$

  Dimensional splitting may seem ad hoc, and its convergence properties suspect, but it was proven [59] that, at least for monotone schemes, this method is in fact convergent. Dimensional splitting is usually cited as the easiest approach, but it does introduce a splitting error due to the decoupling it creates. In practice, however, it is commonly reported (e.g. [6]) that the splitting error added is no worse than the errors contained in the rest of the numerical scheme. Additionally, from the forms of equations (81) and (82), it does not appear that the method of lines may be used, as the splitting approach itself takes control of the time stepping. It would seem that one may lose flexibility in the time stepper choice by using split methods.

- Unsplit Method

  Construction of an accurate method which does not split the dimensions into several sweeps

---

[16]The author would speculate that this is because the coefficient k_1 of the first order error term, $k_1 \Delta x$, is so small that, while the scheme is formally first order accurate in the limit that Δx and Δt go to 0, the first and second order *components* of the error cross over each other at some extremely small grid size, meaning for most practical computations the second order error component dominates and the apparent order of accuracy is 2.

is more difficult than applying a dimensional splitting method. As noted in [6], "In order to achieve high-order accuracy it is still necessary to use information from several grid cells nearby in defining the fluxes, typically by some multidimensional interpolation method."

Unfortunately, the literature for truly multidimensional limiters is scarce. [8] contains a fully discrete multidimensional MUSCL-Hancock approach. Zalesak ([58]) presented a multidimensional flux limiter. However, it was decided that the best scheme would be semidiscrete (so that higher order time steppers, such as equation (79), could be used), ruling out both flux limiting and dimensional splitting. The slope limited reconstructions that have been published (e.g. [54] and [57]) are quite involved. Hence, a new approach was undertaken.

In this work, no special multidimensional slope limiter was developed; instead, the normal one dimensional slopes are used, and just calculated twice (one for each spatial dimension). The reconstructed film thickness within each cell is piecewise planar. The reconstructed values within cell i, evaluated at the cell boundaries, are calculated as before

$$
\begin{array}{rcl}
u_{i-1/2,j} & = & u_{i,j} - \frac{\Delta x}{2}\sigma^x_{i,j} \\
u_{i+1/2,j} & = & u_{i,j} + \frac{\Delta x}{2}\sigma^x_{i,j}
\end{array}
\tag{83}
$$

with two new additions,

$$
\begin{array}{rcl}
u_{i,j-1/2} & = & u_{i,j} - \frac{\Delta y}{2}\sigma^y_{i,j} \\
u_{i,j+1/2} & = & u_{i,j} + \frac{\Delta y}{2}\sigma^y_{i,j}
\end{array}
\tag{84}
$$

A second Riemann solver, identical to (76) but with f replaced by g is used to calculate the circumferential flux, G. One will note that this approach does not calculate truly multidimensional fluxes, despite the advice quoted from [6] above. However, when a higher order time discretization is used, the flux calculation *does* in fact use multidimensional information. This argument can be quantified by comparing the computational stencils[17] of first and second order Runge-Kutta time stepping, as shown in Figure 21. The figure shows that for first order time stepping (Figure 21a), any flow coming into cell i,j at an angle to the grid will not be captured accurately, since cell i,j only uses information from its neighbors aligned along the grid. However, for the two stage Runge-Kutta algorithm (79), the second stage, in effect, applies the first order stencil of 9 cells recursively to each one of the 9 cells used in the first stage. Hence, RK2 should end up calculating a correctly upwinded flux, even if it is in a direction not aligned with the grid. It is in this way that the author expects the spatial order of accuracy of the TLOMM's flux calculations to depend on the temporal order of accuracy chosen.

---

[17]The stencil for any cell i,j is the set of nearby cells which provide time level n information used by the scheme in evolving the solution at cell i,j from time n to $n+1$. The amount of information that is used scales with the number of stages in the time stepper.
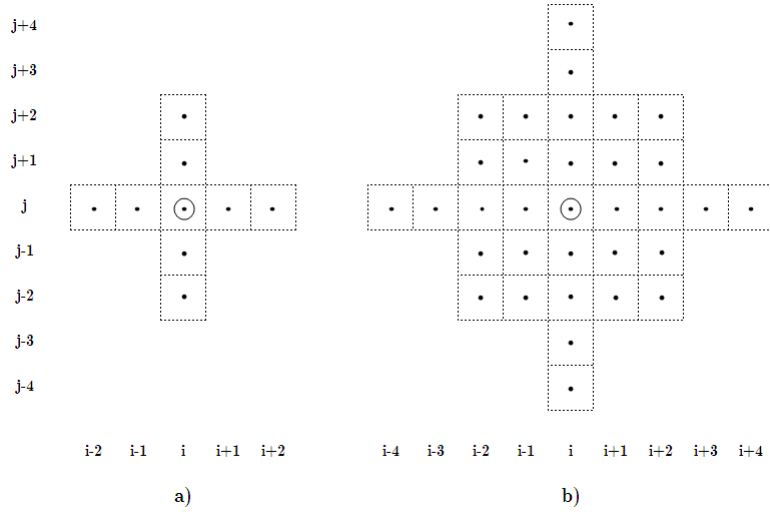
Figure 21: Comparison of multidimensional stencils, for a) single stage time stepping, and b) two stage time stepping.

4.7 Some Alternatives

Although the TLOMM makes use of a number of numerical methods, there are still many other approaches that have been applied to solving hyperbolic conservation laws which were not used in this work. This section's purpose is to briefly point the reader in the directions of alternative solution methods.

4.7.1 Time Stepping

In this work, explicit time stepping was chosen. Explicit algorithms are much more commonly used than implicit methods when applied to hyperbolic problems.

Often the choice of implicit or explicit timestepping which makes the most sense is problem dependent. For example, explicit algorithms for hyperbolic problems, due to the need to satisfy a stability constraint such as $|C| = \left| v\frac{\Delta t}{\Delta x}\right| \leq 1$ (equation (57)), usually require that

$$\Delta t = \mathcal{O}\left(\Delta x\right) \ . \tag{85}$$

However, for explicit schemes applied to *parabolic* PDE's, the analogous stability constraint usually imposes a time step requirement of the form

$$\Delta t = \mathcal{O}\left(\Delta x^2\right) \ . \tag{86}$$

83

According to this equation, any refinement in the spatial resolution of the grid necessitates a substantial decrease in the time step, which usually means that a parabolic PDE requires unacceptably small time step sizes. For this type of PDE, implicit algorithms are usually employed. Although these schemes require more time per time step (because they usually have to solve a set of nonlinear algebraic equations), the time step size can be much larger, since most (though not all [49]) implicit methods are unconditionally stable. One defining characteristic of parabolic PDEs is that the information at one location is "felt" immediately throughout the entire domain; unlike the hyperbolic case, there is no finite time between an event occurring and information about the event reaching some other point in the the domain. So, it should be no surprise that implicit methods make more sense for parabolic equations than for hyperbolic PDE's.

4.7.2   Wave Limiting

Rather than limiting the fluxes or the slopes of a linear reconstruction step, one can also limit the magnitudes of the waves propagated throughout the solution. See [6] and [39].

4.7.3   Front Tracking

Front tracking, also referred to shock fitting ([50]), is a technique which assumes the existence of discontinuties and explicitly tracks each front. Since it requires a bit of *a priori* knowledge about the solution, shock fitting is not as broadly applicable as shock capturing techniques [16]. However, it can be quite fast; the time step is not bounded by a CFL condition such as equation (57), so CFL numbers in the vicinity of $10-15$ can be achieved without loss in accuracy. Requirement #7 of Section 4.3 more or less rules out the approach of front tracking[18]. [13] and [51] are probably good starting points for interested readers.

4.7.4   Random Choice

This method is actually quite old, but it has an interesting feature: despite being a shock capturing approach, it resolves discontinuities with infinite resolution[19]! However, due to the randomness of the method, the actual positions of the discontinuities are inexact. The approach has not seen success in general purpose codes, probably because efforts to extend the scheme to problems in "more than two independent variables...have so far proved unsuccessful" [8]. That it is conservative on average, but not instantaneously, is another interesting property ([35]). Two of the key papers are [52] and [53].

---

[18]It also rules out a scheme based on a generalization of the method of characteristics to 3D, due to inability, at least on the part of the author, to generalize the equal area rule to three independent variables.

[19]i.e. discontinuities are not smeared out over a few grid cells.

4.7.5 ENO/WENO

Essentially Non-Oscillatory (ENO) methods are shock capturing schemes which are uniformly high order accurate and avoid the limitations discussed in Section 4.4.4. They accomplish this feat by using a scheme which is even more nonlinear than the standard TVD limiting approaches discussed in Section 4.4.1: the computational stencil *itself* depends on the local smoothness. By using a variable stencil and not allowing the stencil to span across discontinuities, the computed solution is almost always high order accurate. Of course, ENO methods are essentially, not strictly, TVD. Weighted ENO methods are a more recent development which achieve even higher accuracy, by using a weighted combination of all possible stencils. The important developments of these schemes may be found in [44], [45], [46], [47], and [48].

4.7.6 Discontinuous Galerkin

The Discontinuous Galerkin (DG) approach is a variant on the finite element method which allows for the solution values to be discontinuous at the element boundaries. It is related to the simulation approach undertaken in this work, as it involves the usage of piecewise polynomial basis functions (of a user-set order), limiters, and Riemann solvers. Its popularity has been growing in recent years; see [54] and [55].

This page was intentionally left blank.

5 Implementation

Extensive program validation efforts were performed, in hopes of building evidence-based confidence in the Top Land Oil Movement Model's predictions, as well as verifying its robustness and reliability. This section describes some of these undertakings. It also discusses the all-important topic of the oil supply rate, as well as a few of the important but non-obvious aspects of the program itself.

Empirical Determination of TVD requirement   One will recall that explicit algorithms, which are used in the TLOMM, usually must satisfy some Courant number constraint such as equation (57) in order to be stable. As discussed in Section 4.2.9, schemes must also satisfy the TVD test in order to have a total variation which diminishes with time. As long as variable coefficients are avoided (requiring that any gas flows are uniform), and the flux is convex (requiring that the axial gas velocity is in the same direction as the inertia force), the exact solution itself should be TVD. The CFL-like TVD condition was not derived analytically for the TLOMM schemes, so it was determined empirically.

The simulation keeps track of the instantaneous maximum film thickness and plots it at the end of a simulation. It also calculates the instantaneous (though approximate) x and y Courant numbers in every cell, and keeps track of the maximum values at each time step. The simulation does not, however, actually loop over all cells and calculate the true total variation. A series of careful tests designed to detect the scheme's total variation property using the instantaneous maximum film thickness plots were performed using the 3D simulation with RK1 timestepping. The max film thickness was observed to monotonically decrease with time (i.e., preserve the TVD property, at least in a rough sense) when the Courant number always remained less than roughly 0.5. For values of C_{max} greater than 0.5, however, the max film thickness occasionally increased with time, which certainly violates the TVD property. Hence, it is safe to conclude that the schemes used in the TLOMM must satisfy the Courant number requirement $C \leq 0.5$[1] in order to remain TVD (and hence nonlinearly stable)[2]. Two of these plots are shown in Figure 22, for reference. The increase in maximum film thickness in Figure 22b is circled, as it may be difficult to observe.

---

[1]From the literature, this 0.5 constraint is common for unsteady simulations with two dimensional spatial domains. However, it may be different for RK2 and RK3, which were not tested.

[2]Again, the full governing equation, (22), is not strictly TVD itself, due to both flux nonconvexity and gradients in the gas flows. The objective is that any increase in total variation be physically based, and not be due to the numerical scheme.
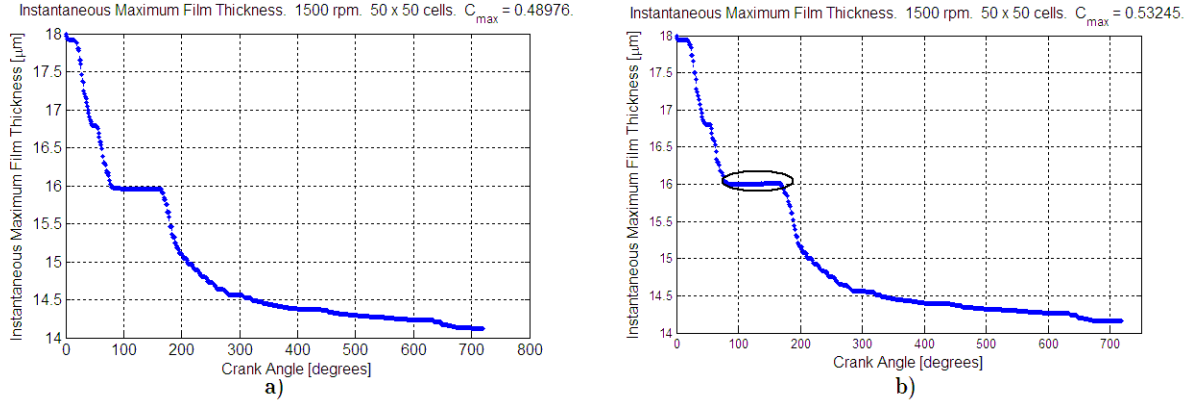
Figure 22: Adherence to the TVD principle as a function of Courant number. a) $C_{max} \approx 0.49$: scheme still TVD; b) $C_{max} \approx 0.53$: scheme no longer TVD.

5.1 Validation

A sizeable amount of validation was performed on the TLOMM. Both the 2D and full 3D simulations were confirmed to be strictly conservative. In addition, detailed qualitative and quantitative analyses were carried out using the 2D simulation, which approximates the simplified governing equation, (51).

5.1.1 Conservation

Using a scheme which is strictly conservative is important for several reasons, as discussed in the previous sections. In addition to assuring physically correct solutions, rigorously checking conservation of the scheme also provides a mechanism for seeking out programming bugs. The 2D flux limiting, 2D slope limiting, and 3D simulations were heavily tested.

Programming logic is in place within the program to check global mass conservation. All TLOMM simulations keep track of the instantaneous total volume contained within the domain, as well as the instantaneous flow rates through all of the domain's boundaries. The program automatically calculates the error between the instantaneous net inflow rate through the boundaries and the rate of change of the oil volume, the latter being calculated using a simple backwards difference approximation.

A variety of inflow boundary conditions, both constant and varying in time, were simulated, along with all choices of limiter functions. The tests also covered various initial conditions. Some tests used a modified piston acceleration which made the inertia force constant in time. Conservation checks on the 3D program also included introducing a variety of gas flow patterns. All tests

came out positive; the simulation does not spuriously gain or lose mass. With confidence in the scheme's conservative properties established, error analyses were performed to compare the performance of the various available formulations (flux vs. linear reconstruction), time stepping algorithms, and limiter functions.

5.1.2 Qualitative Error Analyses

To develop an intuitive feel for the ways in which the various shock capturing schemes developed in Section 4.4 behave, a series of tests were carried out. A lot can be gained from qualitative comparisons between schemes, and snapshots comparing approximate and exact solutions are easily interpreted with only a small investment of effort[3].

The 2D simulation approximates equation (51) and was used to generate the results. The finite volume predictions were then compared pictorially with exact calculations made using the method of characteristics. The initial film thickness profile was a uniform $9\,\mu$m layer plus a $\sin^2$-shaped bump in the middle, having maximum height of $18\,\mu$m (exactly like the initial thickness profiles pictured in Figure 16). 100 cells made up the grid, and the time step size was 3 crank angle degrees (CAD). All runs used a crankshaft speed of 2000 rpm and a top land height of 11.5 mm. The film thickness available at both boundaries was the initial base film thickness, $9\,\mu$m. The oil viscosity was set to 0.006 Pa-sec. The parameters were chosen in order to get the film to come somewhat close to creating shock, but never quite get there, so that the non-TVD schemes such as Lax-Wendroff (which is formally second order accurate) could be included in the comparison[4].

Figures 23, 24, and 25 present comparisons between the exact solution and many approximate solutions. Note that the set of limiters chosen for comparisons varies slightly from one figure to the next. Figure 23 compares various limiter functions within the flux limiting framework. Likewise, Figure 24 compares limiters using the slope limiting approach with single stage timestepping (RK1, equation (78)), while Figure 25 makes practically the same comparisons but with two stage timestepping (RK2, equation (79)) instead. The snapshots depict the exact and approximate solutions after 10 full engine cycles, with the piston position returned to TDC ($0°$ CA). The initial conditions for the exact (characteristics) and approximate (finite volume) calculations were always identical, so they are not shown. The simulations were always started at $0°$ CA. Since shock never occurs, the film basically rocks and forth in place but never actually spreads out, so the *correct* solution at TDC after 10 cycles is actually the same as the initial condition. This observation makes it easy to pick out the underperforming scheme combinations.

---

[3]The reader who is seeking rigor will find it in the next section.

[4]Recall from Figure 16b what happens to the unmodified Lax-Wendroff scheme if the film is allowed to develop shock.
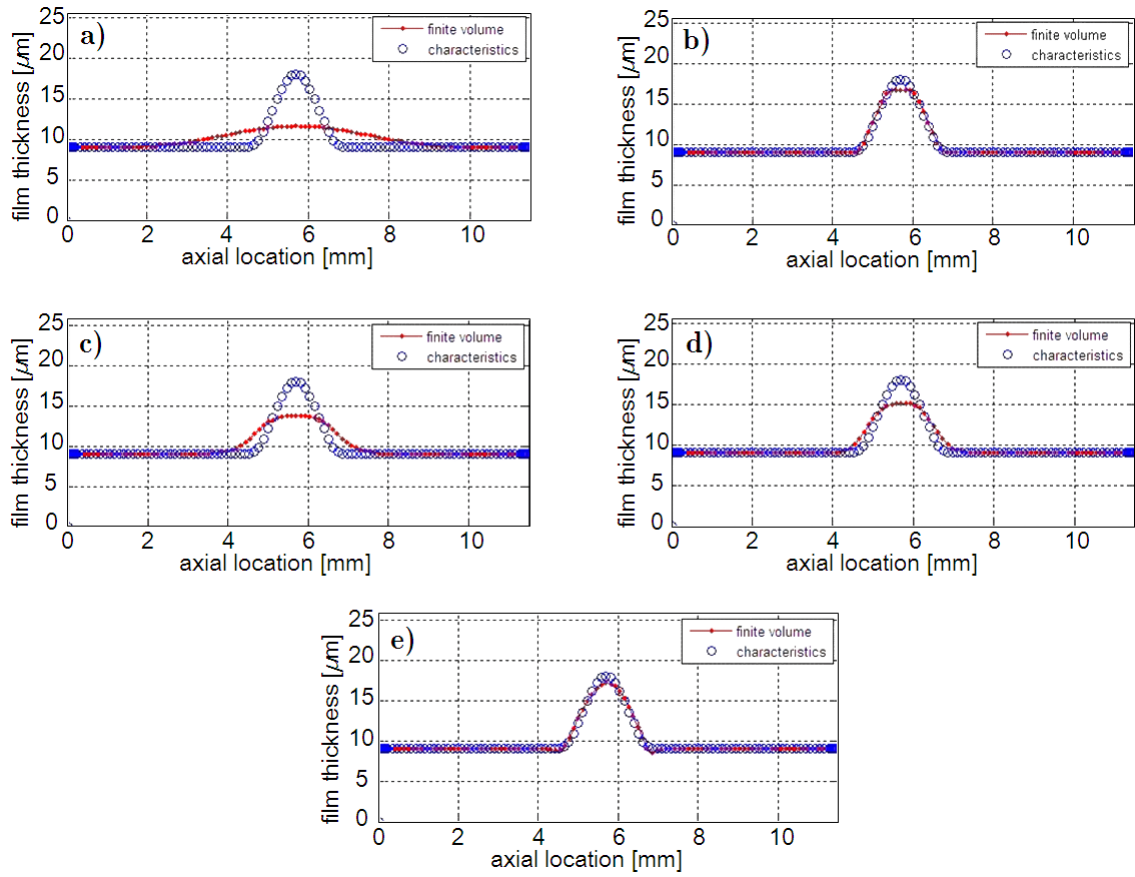
Figure 23: Qualitative comparisons for the flux limiting formulation, using various limiters: a) Upwind, b) Superbee, c) Minmod, d) Van Leer, and e) Lax-Wendroff.

Figure 24: Qualitative comparisons for the slope limiting formulation, using RK1 timestepping, for various limiters: a) Upwind, b) Superbee, c) Minmod, d) Van Leer, e) Monotonized Centered, and f) Lax-Wendroff.

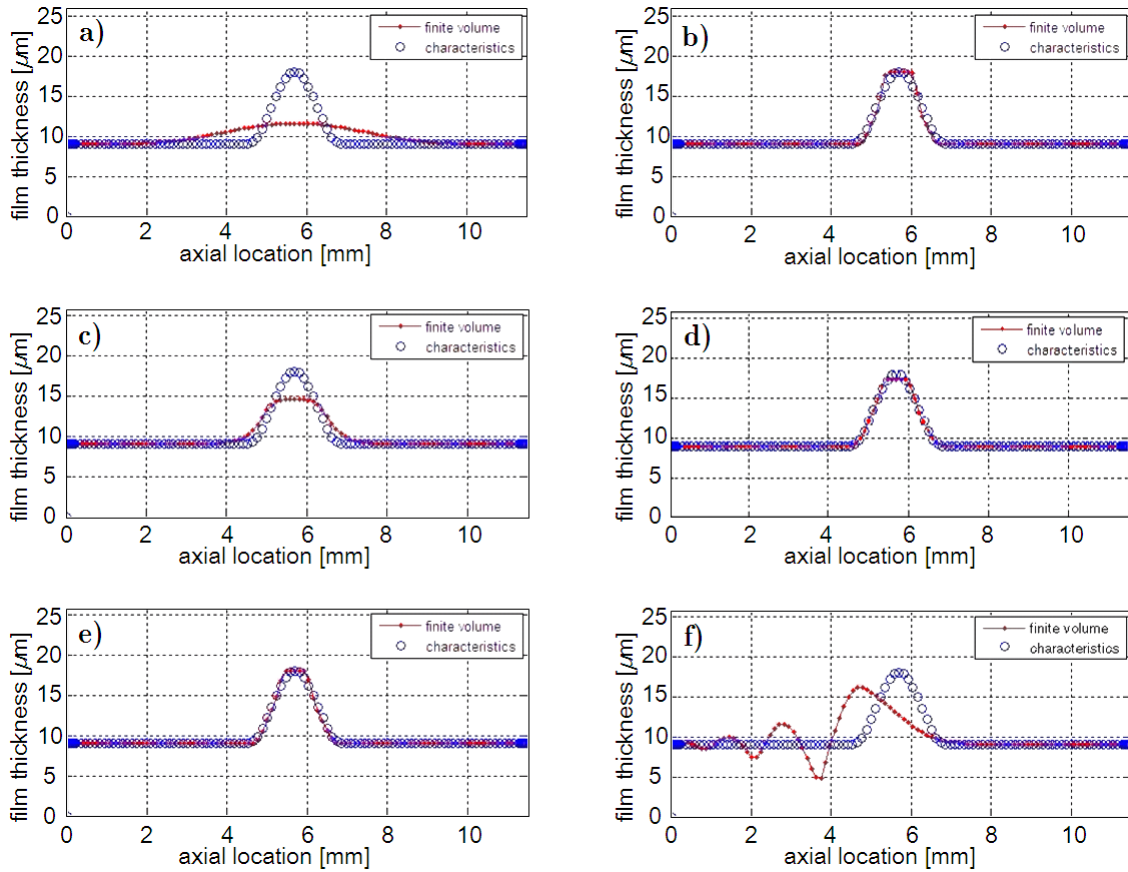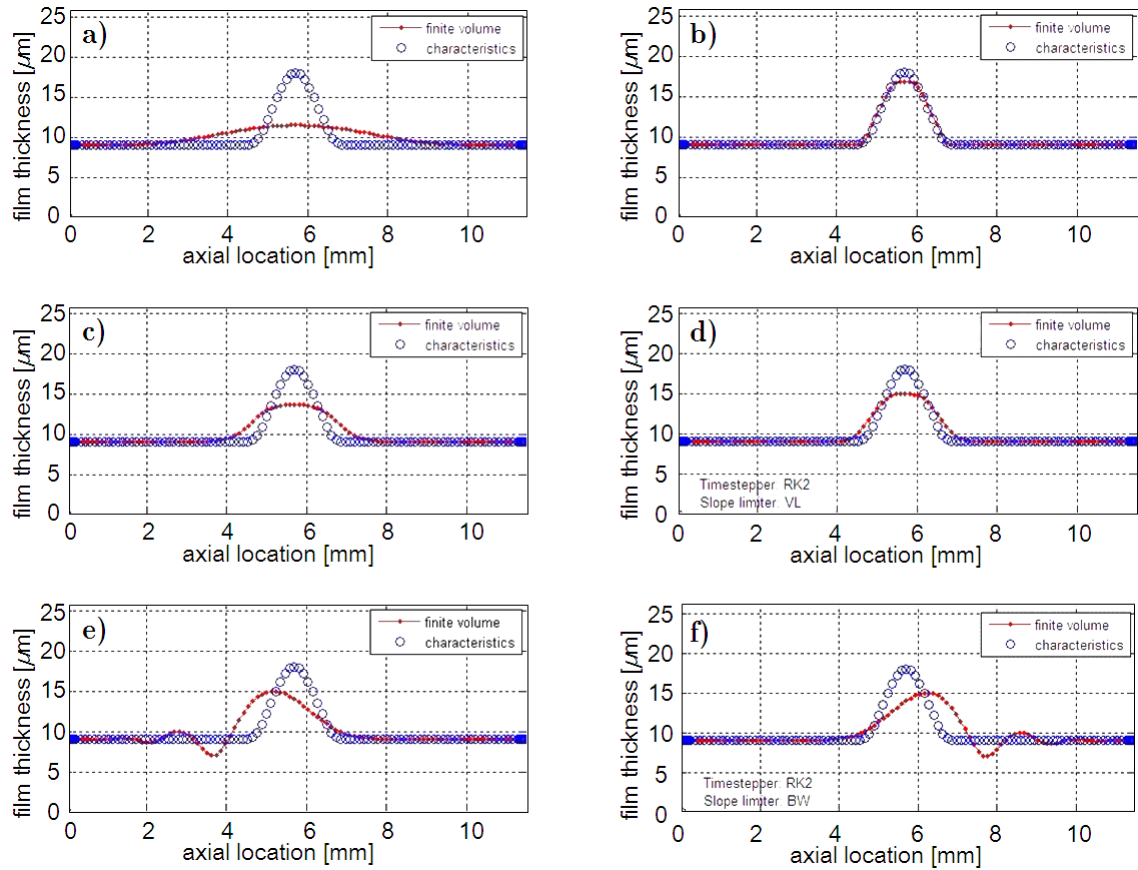Figure 25: Qualitative comparisons for the slope limiting formulation, using RK2 timestepping, for various limiters: a) Upwind, b) Superbee, c) Minmod, d) Van Leer, e) Lax-Wendroff, and f) Beam-Warming.
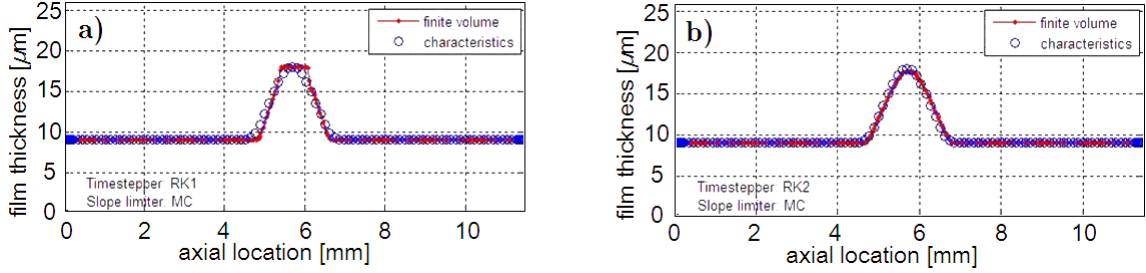
Figure 26: Qualitative comparison for slope limiting, using the MC limiter, showing effect of time stepper at finer grid resolutions: a) RK1, b) RK2.

Discussion It can be seen that the qualitative behaviors of some of the schemes (e.g. LW) vary from one implementation to another, while others do not. The upwind scheme, which is expected to converge at only a first order rate, certainly exhibits large amounts of dissipative error, in essence behaving identically regardless of the formulation and temporal discretization. To the contrary, the slope limiting LW scheme behaves very differently than the flux limiting LW scheme, to which it is supposedly analogous (they are identical when applied to linear PDEs). In Figure 25, the LW and BW schemes behave very much alike, except their dispersive errors are biased in opposite directions (this phenomenon is commonly observed for these two sister schemes, e.g. [16]). The Superbee limiter lives up to its reputation for being over-compressive and is observed to be the least diffusive of the limited schemes. As one would hope, the high resolution schemes built up in Section 4.4 behave relatively well overall, at least when shock has not occurred[5].

One may have noticed that the slope limiting results using the RK1 timestepper actually appear to be substantially closer to the exact solution than the slope limiting results using the RK2 timestepper. Usually intuition would say that it works the other way around; after all, upgrading one component of a scheme to a higher order version is done because "higher order" is usually synonymous with "better." Unfortunately, whether or not it is actually better can turn out to depend on the degree of refinement in the grid. Figure 26 depicts a special comparison: RK1 and RK2 are again evaluated, for a slope limiting formulation equipped with the MC limiter, but this time 300 cells and one CAD timesteps have been used instead[6]. With this level of grid refinement, the second order time stepper clearly wins over the first order one. Why does RK2 outperform RK1 for a fine grid, but the opposite happens for a coarse grid? This interesting quirk of the error analysis is investigated in more depth, and explained, in the following section[7].

---

[5]They behave well when shock is present too. Figure 16a already showed the slope limiting 2D TLOMM, using the MC limiter and RK2 timestepper, allowing a shock to occur naturally and without incident.
[6]Note that $\frac{\Delta t}{\Delta x}$ is still the same.
[7]I'm sure you were planning to read it anyway.

5.1.3   Quantitative Error Analyses

Qualitative analyses are highly illustrative, but analyzing the error quantitatively adds rigor to the approach and verifies that a scheme is truly as accurate as it is rated to be. To this effect, quantitative error analyses were carried out using the 2D TLOMM simulation of equation (51). Several choices of flux limiters, slope limiters, and time steppers were compared to one another. With the exception of the number of cells and the time step size, all simulations presented here used the same set of parameters as the qualitative analyses in Section 5.1.2.

The process went as follows. For each choice of scheme, results were first obtained using a very highly refined grid (5120 cells), which would serve as the (pseudo) "exact" case. Then the simulation was run using 160, 320, 640, and 1280 cells. All of the data was filtered down to only 160 cells, in order to consistently make comparisons at the same spatial locations. The mesh ratio $\frac{\Delta t}{\Delta x}$ was held constant, to avoid any Courant number effects; hence, the results were also filtered temporally in order to only make comparisons at identical values in time.

For each combination of algorithm (flux vs. slope limiting) and time stepper, the instantaneous global, filtered error between each of the "coarse" discretizations and the exact case was calculated using three different norms, L^1, L^2, and L^{∞}[8]. In addition, the standard assumption that the error scales with Δx raised to some power p, i.e.

$$e = c\Delta x^p \tag{87}$$

was made[9].

Plots of error vs. grid refinement   Figures 27 and 28 plot the L^1 error as a function of mesh refinement, for a variety of test cases[10]. These plots contain some interesting results. It can be observed from Figure 27 that the choice of timestepper does matter - for both the unmodified Lax-Wendroff (LW) slope and the Monotonized Centered (MC) slope, the slope of the plot trendlines are larger for the RK2 time stepper than for RK1. Being on log-log axes, the slope of the trendline is the computational order of accuracy, p, according to the assumed error scaling, equation (87). These schemes benefit from the higher order time stepper because, being second order accurate (or close to it) spatially, the error due to temporal discretization is dominant when only first order time stepping is used. The upwind (UW) scheme, on the other hand, does not benefit from the improvement in time stepping, because it is still bottlenecked by

---

[8]The various 5120 cell reference cases were compared to one another as a consistency check; the differences between them were small compared to a typical 1280 vs 5120 cell error.

[9]Yes, $e = c_1\Delta x^{p1} + c_2\Delta t^{p2}$ would be more complete, but we kept $\frac{\Delta t}{\Delta x}$ constant.

[10]Readers are advised to view these figures in color.
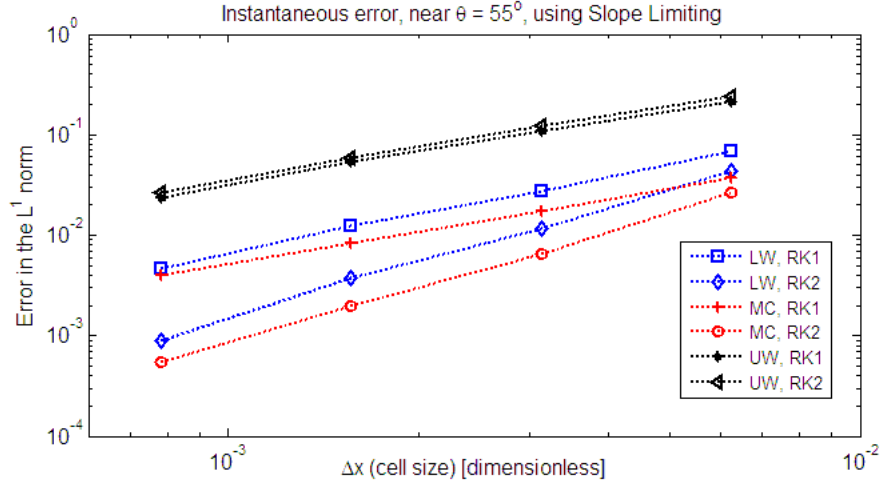
Figure 27: L^1 error comparison between first and second order time stepping, for various slope limiter choices.

its paltry first order accuracy in space. In addition, it is surprising to see that the magnitudes of the MC scheme's errors actually came out to be lower than the LW scheme's errors, despite the MC scheme's slightly smaller expected order of accuracy. Finally, one may note that at the coarsest discretization (160 cells), the RK2 MC scheme's measured error is just slightly less than that of the RK1 MC scheme; certainly the extrapolations of their trendlines would cross at some coarser grid resolution. Yet from the qualitative error analyses presented in Section 5.1.2, in which all of the grids had only 100 cells, it certainly appeared that the RK1 MC scheme was *more* accurate than RK2 MC. It would be reasonable to postulate that somewhere between 100 and 160 cells, the errors for the two schemes are equal. The implication of the existence of a crossover point is that for grids any coarser than the grid resolution at which crossover takes place, one is better off using the first order time stepper than the higher order stepper! This counter-intuitive result confirms the notion alluded to in Section 5.1.2, and represents one of the unfortunate tradeoffs one must face when using underresolved grids. A somewhat analogous situation involving crossover of schemes with different orders of accuracy is discussed in [6].

Figure 28 presents an error comparison between some flux limiting and slope limiting schemes. All of the calculations used RK1 (Euler Forward) time stepping. For both types of limiting, the use of either Superbee (SB) or LW gives quite a large reduction in the magnitude of the error, compared to the basic UW scheme. The SB slope limiter's characteristics seem to improve substantially (the error stops increasing!) as coarser grids are used, but this anomalous behavior is not observed for the SB flux limiter. It would appear from this plot that flux limiting might perhaps be more accurate than slope limiting, as the flux limiting results are generally shifted downward[11]. However, both methods on this plot use only first order time stepping; one of the

_____

[11]This occurrence is probably because the flux calculated by a flux limiting scheme attempts to be an average
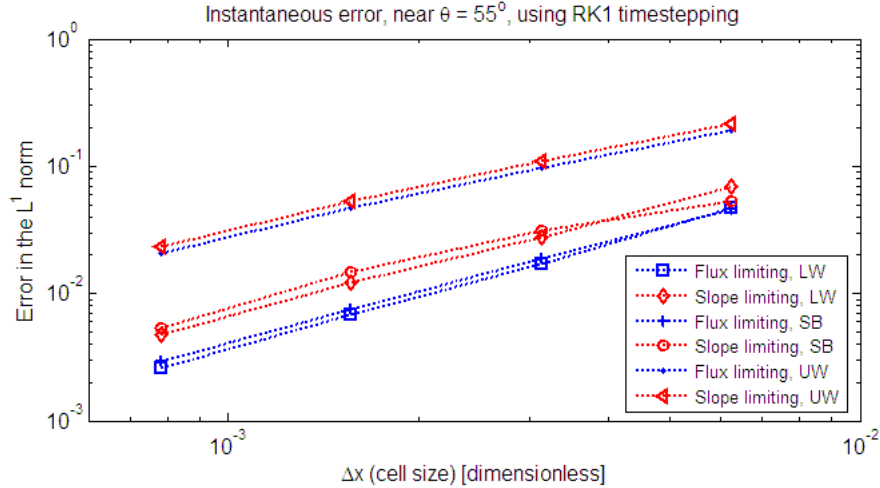
Figure 28: L^1 error comparison between flux limiting and slope limiting, for various limiter choices.

beauties of the slope limiting approach is that it can be used with higher order time steppers, unlike the (fully discrete) flux limiting method, as discussed in Section 4.4.3. With that in mind, it can be seen that for the same amount of grid refinement (1280 cells), the minimum amount of error achieved using flux limiting (about 0.0025, from Figure 28) was not nearly as low as the minimum amount of error observed using RK2 slope limiting (about 0.00055, from Figure 27). Hence, flux limiting may indeed have a lower value of c in equation (87), but it is constrained (and observed) to have $p \approx 1$ due to a low order timestepper, unlike slope limiting.

Computational order of accuracy   Each scheme's instantaneous computational order of accuracy was calculated according to equation (87). According to this assumption, p can be determined by forming the ratio of two error measurements,

$$\frac{e_2}{e_1} = \left(\frac{\Delta x_2}{\Delta x_1}\right)^p \; , \tag{88}$$

where Δx_1 represents a more refined grid (everything else constant) than Δx_2, or vice versa. Note that since $\frac{\Delta t}{\Delta x}$ was kept constant, equation (88) can be interpreted as the ratio of errors as both Δx and Δt uniformly go to zero. As mentioned above, p can be interpreted graphically as the slope of the plot trendlines in Figures 27 and 28.

The set of instantaneous order of accuracy data, given at each time step, for all three (1280 vs. 640, 640 vs. 320, and 320 vs. 160) grid refinements, for several combinations of limiters and time steppers, in all three norms, is of course too large of a data set to make much sense of on

---

over the time interval $t_n \rightarrow t_{n+1}$, which would most likely be more accurate than simply using the instantaneous flux at t_n, which is what RK1 slope limiting does.

its own. Hence, for each combination of slope limiter and time stepper, this data was averaged over all time steps and grid refinements. These results are presented in Table 4. There is one sub-table per norm, but all three norms considered gave similar results.

The numbers in Table 4 essentially confirm what is expected from the scheme *design* considerations discussed in Sections 4.4 and 4.5, as well as the plots in Figures 27 and 28. Upgrading from first to second order time stepping gives a large order of accuracy improvement to all but the Upwind scheme, as expected. However, upgrading the temporal discretization from second to third order did *not* improve the results, probably because the spatial discretization (at no better than second order accuracy) had become the bottleneck[12]. Interestingly, when using RK2, the MC limiter outperforms the Superbee limiter and exhibits an order of accuracy on the same level as Lax-Wendroff (and of course LW cannot capture shocks)[13].

Though Table 4 contains a convenient summary, the plots in Figures 27 and 28 show information that is not displayed in the table: they highlight the fact that the various schemes distinguish themselves from one another via their error magnitudes, in addition to their orders of accuracy. In other words, the table only shows p, whereas the plots give a rough idea of both p and c in equation (87).

5.2   Boundary Conditions

Though some simulation results have been presented in previous sections, the implementation of boundary conditions has not actually been discussed. Both the 2D and 3D versions of the simulation incorporate boundary conditions using the ghost cells approach described in [6]. This method allows the same formulas to be used anywhere, regardless of how close to the boundary any interior cell is.

5.2.1   Grid Setup

Figure 29 depicts a schematic of a prototypical TLOMM grid having only six cells in both the axial and circumferential directions. The interior cells which span the full top land are shaded, while others (the ghost cells) are not. Two layers of ghost cells are needed around the full boundary because at any given time the derivatives are calculated, the nine point stencil of Figure 21a is used. The TLOMM uses the convention that i denotes a cell's x direction index and j denotes a cell's y direction index (both starting at 1 at the cell located at the coordinate system's origin).

---

[12]One would expect that if the spatial discretization was third order accurate, going from RK2 to RK3 would have shown a large improvement.

[13]It is for this reason that the TLOMM is set up to use the MC limiter by default.

97

Computational Order of Accuracy (averaged over all choices of I, and all time), in L^1 norm.				
Slope **Timesteps**	Upwind (1st order)	Lax-Wendroff (2nd order)	Superbee	Monotonized Centered (MC)
RK1 (Euler Forward)	1.06	1.29	1.21	1.12
RK2 (Heun)	1.05	2.00	1.71	2.07
RK3	-	1.99	1.71	2.09

Computational Order of Accuracy (averaged over all choices of I, and all time), in L^2 norm.				
Slope **Timesteps**	Upwind (1st order)	Lax-Wendroff (2nd order)	Superbee	Monotonized Centered (MC)
RK1 (Euler Forward)	1.00	1.35	1.14	1.10
RK2 (Heun)	0.99	1.96	1.57	1.97
RK3	-	1.96	1.56	1.98

Computational Order of Accuracy (averaged over all choices of I, and all time), in L^∞ norm.				
Slope **Timesteps**	Upwind (1st order)	Lax-Wendroff (2nd order)	Superbee	Monotonized Centered (MC)
RK1 (Euler Forward)	0.91	1.35	1.10	1.16
RK2 (Heun)	0.91	1.94	1.46	1.92
RK3	-	1.95	1.45	1.93

Table 4: Order of accuracy results for 2D slope limiting simulation in various norms.
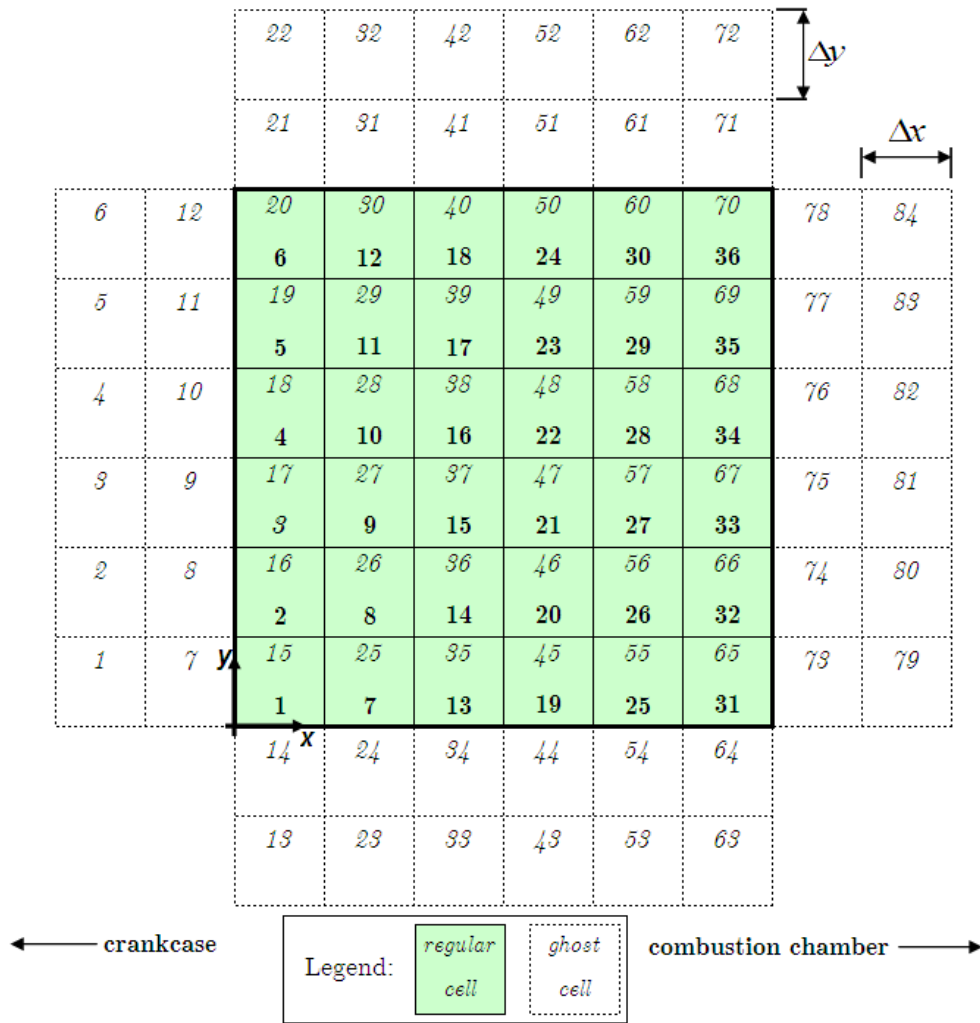
Figure 29: Schematic of the computational grid, including ghost cells, for the choice of $I = J = 6$.

Moreover, the total number of cells in the axial [circumferential] direction is represented by I [J]. The figure shows the two cell addressing schemes which are used simultaneously within the program: the italic, non-bold numbers indicate the cell identifiers using the numbering scheme which includes the ghost cells, and the bold numbers indicate the cell identifiers using the numbering scheme which does not include ghost cells (only the cells within physical domain of the problem are assigned one of these numbers). Note the location of the origin of the coordinate system.

5.2.2 Boundaries at the Circumferential Extremes

Periodic boundary conditions are imposed along the outer boundaries of cells $(i, j = 1)$ and $(i, j = J)$. That is, whatever flows out through the boundary at $j = 1$ flows in through the boundary at $j = J$, and vice versa[14]. The equations for the values within the ghost cells are simply

$$
\begin{aligned}
h_{i,j=0} &= h_{i,j=J} \\
h_{i,j=-1} &= h_{i,j=J-1} \\
h_{i,j=J+1} &= h_{i,j=1} \\
h_{i,j=J+2} &= h_{i,j=2}
\end{aligned}
\tag{89}
$$

These BC's do not allow any net amounts of mass to flow in or out of the domain; they only allow mass to move to different areas within the grid.

Justification   Recall from earlier sections that the TLOMM simulates only a portion of the top land, based on the gas flow data which is provided. Typically only one fuel injector's spray segment is simulated; for example, the TLOMM will have a domain of one fifth of the full top land if there are five fuel sprays. The implicit assumption in choosing periodic BC's is that each spray segment of the top land has identical conditions as all the other segments. Of course this assumption is one that an engine manufacturer makes when it chooses to simulate only one spray.

5.2.3 Boundaries at the Axial Extremes

The boundary conditions imposed along the outer boundaries of cells $(i = 1, j)$ and $(i = I, j)$ are a bit more complicated than the previous case. At any of these boundary cells, either outflow or inflow may be taking place across the boundary. The machinery for setting up the ghost cells for outflow vs. inflow is discussed first, followed by a presentation of an algorithm which was created to choose the appropriate flow condition at any point along the boundary.

Setting outflow conditions   When outflow is taking place, ghost cells are assigned values which are essentially an extrapolation of the interior domain. Computing the ghost cell values based on the interior values, during outflow, forces the information to be properly upwinded. In this work, first order extrapolation was used. For a point along the boundary at the bottom of

---

[14]Readers familiar with the arcade game PAC-MAN may view periodic BCs as being practically identical to the portals on the sides of the game map which allow a player to jump to the other side of the screen.

the top land, the equations activated when outflow is detected are

$$
\begin{aligned}
h_{0,j} &= 2h_{1,j} - h_{2,j} \\
h_{-1,j} &= 2h_{0,j} - h_{1,j} \ ,
\end{aligned}
$$

and for a point along the boundary at the top of the top land,

$$
\begin{aligned}
h_{I+1,j} &= 2h_{I,j} - h_{I-1,j} \\
h_{I+2,j} &= 2h_{I+1,j} - h_{I,j} \ .
\end{aligned}
$$

[6] discusses these conditions in more detail.

Setting inflow conditions   The inflow boundary conditions are straightforward. When inflow is taking place at the outer boundary of one of the cells for which $i = 1$ or $i = I$, the program simply sets the associated ghost cell values to the user-defined available film thickness contained in the TLOMM input data file. This input file includes data fields for the film thickness available both above and below the top land[15].

BC Algorithm   The previous two paragraphs have discussed the low-level procedure for applying BCs, assuming that one knows whether inflow or outflow is occurring locally. However, the task of actually *detecting* the appropriate flow condition is a bit subtle, and turns out to require a well thought-out algorithm.

To avoid overspecifying the system, one cannot simply impose arbitrary BCs regardless of the local conditions. Since the axial gas flows' open-ended spatial variation makes it possible for different flow conditions (in vs. out) to take place along several segments of the same boundary, the choice of inflow vs. outflow must be made on a cell by cell basis.

Due to the possibility of the gas flow induced shear forces competing with the piston acceleration force (nonconvexity of the flux, see Section 4.4.3 and Figure 20), it is not always immediately evident whether inflow or outflow boundary conditions are appropriate. The cause of this uncertainty is that for some values of film thickness, inflow would be correct, while for other values, outflow would be correct. Unfortunately, one does not know the film thickness at the boundary *a priori*; the average value in the interior cell is known but its reconstructed value at the boundary depends on the slope σ within that cell, which *itself* depends on the cell average in the ghost cell

---

[15]Obviously most users would reasonably assume that any oil which flows out through the top of the top land is not available to re-enter, and would leave the film thickness available "above" the top land set to zero. The functionality is there to allow for easy adaptation of the program to other piston lands, if desired.

adjacent to the boundary (revisit Table 3, if needed). The value within the ghost cell depends on whether inflow or outflow boundary conditions are being set. Hence, one cannot determine the direction of the flux at the boundary without knowing the cell average in the ghost cell, yet simultaneously one cannot know the cell average in the ghost cell without knowing the direction of the flux at the boundary. The problem is ill posed, but a solution was found.

A custom algorithm was developed to ensure consistent selection of the correct boundary conditions, and works as follows. At each of the $2J$ locations along the "top" and "bottom" boundaries ($x = 0$ and $x = $ [top land height]) respectively, the TLOMM does the following:

```
1.  Evaluate the axial gas flow-induced shear stress at the center of the
segment of the boundary being considered.

2.  Assign a value to the ghost cell adjacent to the boundary using the
"inflow" BC.

3.  Calculate the x-direction slopes in both the interior cell adjacent
to the boundary and the ghost cell adjacent to the boundary.

4.  Calculate the tentative flux through the boundary using reconstructed
values (using the slopes from step 3) on both sides of the boundary, via
the Riemann solver.

5.  If the flux is into the domain, use the inflow BC chosen and exit the
BC selection algorithm.  Otherwise, continue:

6.  Assign a value to the ghost cell adjacent to the boundary using the
"outflow" BC instead.

7.  Calculate the x-slopes and flux at the boundary as before.

8.  If the flux is out of the domain, use the outflow BC chosen and exit
the BC selection algorithm.  Otherwise:

9.  Assign inflow BC's[16].
```

To some this effort may seem like overkill, but after extensive testing, certain pathological cases were found to exploit loopholes in an earlier, simpler algorithm. In these cases, outflow was "detected," yet the extrapolated ghost cell values were large enough that the Riemann solver calculated an inflow flux instead. This process repeated itself and soon the solution became

---

[16]If the algorithm reaches this point, then the situation is degenerate; attempts of choosing inflow and outflow both produced fluxes which were inconsistent with the BC assumed. If this happens, it is safest to assign inflow BC's, because the Riemann solver will end up calculating an outward flux. Allowing the opposite to happen, in which the Riemann solver calculates an inward flux despite outflow BC's being prescribed, can spuriously increase the film thickness within the regular cell adjacent to the boundary, leading to potential numerical instability!

numerically unstable due to the interior film thickness adjacent to the boundary shooting off to infinity. Hence, some amount of rigor was in fact necessary to ensure reliability of the simulation. The algorithm presented above makes certain that this process does not happen.

Since all interface fluxes (at least in the final 3D TLOMM) are calculated using the Riemann solver, equation (76), the information at the boundaries is always properly upwinded. That is, in any computational cell adjacent to a boundary, the ghost cell data is only used if the Riemann solver detects *inflow* as taking place. This feature is important because usage of ghost cell data when information is flowing out through the boundary would a mathematical crime. As the reader is probably aware, overspecifying a system's boundary conditions can destroy a solution. For example, the PDE considered here is first order; in the case of the 2D TLOMM with constant coefficients, only one boundary condition could legitimately be specified at any given time (either at the top *or* the bottom of the top land, depending on the flow direction, not both). With variable coefficients and more dimensions, the situation is of course more complicated, but the general rule is that BCs may only be imposed if characteristics are locally *entering* the domain.

5.3 Oil Supply Considerations

Though the boundary conditions algorithm developed above is robust, it still does not address a fundamental question:

> In an operating IC engine, what film thickness of lubricant is available to flow into the top land region, from both below and above the domain, at any given time within the cyle?

It is probably realistic to assume that once oil flows out through the *top* of the top land, it is gone and cannot come back. This assumption would imply that the film thickness available above the top land is zero. However, the amount of oil available at the bottom of the top land is much more complicated.

What lies just below the bottom of the computational domain is the upper lip of the top ring groove, from which studies have shown oil to emerge, via a mechanism in which the rings' natural motion slowly pumps lubricant upward [3]. Hence, one route by which lubricant can enter the top land is the through the top ring groove.

A second mechanism exists as well; under high load conditions, the forces on the top ring can be sufficient enough to cause it to scrape off some of the oil sitting on the liner during upstrokes.

This oil may then get transported to any of several locations, one of which being the top land. Detailed calculations made by another simulation developed by the author[17] estimated the top ring upscraping rate for a diesel engine with $2.3\,\mathrm{L}$ of displacement per cylinder, at max load, to be about $53\,\mathrm{g/hr}$ per cylinder. The ringpack in this calculation consisted of a top ring and oil control ring both with symmetric barrel profiles, and a second ring with a wedge shape. Although this estimate may roughly quantify the upscraping rate, only some of the upscraped oil actually makes it to the top land, so it does not really tell the whole story. The following section describes the approach taken in this work, which avoids these ambiguities.

5.3.1 TLOMM's Oil Supply Mechanism

Ideally, a TLOMM user would not have to try to guess the film thickness available below the top land. This parameter depends on many competing factors; it is typically unknown and would be speculative at best. However, engine oil consumption rates *are* typically available, being routinely measured in the laboratory.

In an operating engine, it would not be unreasonable to expect that the amount of oil introduced to the top land each cycle is proportional to the engine oil consumption rate. Under this assumption, a better way to implement a supply mechanism would be to simply require a user to input 1) the oil consumption rate, and 2) a multiplication factor, and then have the program introduce a *fixed* volume of oil each revolution according to these inputs. This is the approach taken in the 3D TLOMM[18]. The multiplication factor mentioned represents the ratio $\frac{\text{oil added to the top land of each cylinder per cycle}}{\text{oil consumed by each cylinder per cycle}}$.

The simulation introduces this fixed volume per cycle by brute force. It does not attempt to introduce it through the boundaries. One might think that the bottom of the top land boundary condition required in order to introduce the desired volume per cycle could be back-calculated, by integrating the piston acceleration component of the inflow rate over the part of the cycle for which the inertia force is upward. However, the presence of gas flows makes analytical calculation impossible; their strength, duration, location, etc. are completely arbitrary - they can change the flow conditions at the boundary, meaning the actual amount of oil introduced per cycle would not necessarily match the attempted amount.

To ensure that the amount of oil added each cycle truly matches the user input (*desired* amount of oil to be added), the simulation adds a small ridge of oil every revolution on top the existing oil distribution, phased to occur at the beginning of the upward inertia force period. The location

---

[17]This work was not published, but it used a formulation not unlike that found in the friction simulation developed in [2].

[18]As well as in [3].

at which the ridge is added, as well as its axial width, is a user input (but by default the ridge is set up to be added just slightly above the bottom of the top land, with an axial width of 5% of the top land height). This approach is obviously adapted from [3]. For any given invocation of the oil addition subroutine, the existing film is simply augmented by some additional film thickness, i.e.

$$h\left(x\right) = h\left(x\right) + h_{add}\left(x\right) \ . \tag{90}$$

For the TLOMM, the ridge is given a sinusoidal shape,

$$h_{add}\left(x\right) = \sin\left(\frac{x - x_{rs}}{x_{re} - x_{rs}}\pi\right)k_m \ , \tag{91}$$

where x_{rs} is the x-coordinate of the ridge's starting location and x_{re} is the x-coordinate of the ridge's ending location. k_m specifies the magnitude of the ridge, and is related to the amount of oil it is desired to add,

$$k_m = \frac{\pi V_{add}}{2y_{max}\left(x_{re} - x_{rs}\right)} \ , \tag{92}$$

where V_{add} is the volume the program needs to add and y_{max} is the circumferential width of the domain (e.g. one fifth of the top land circumference if each cylinder has five fuel sprays). The sensitivity of the results to the shape of the ridge was not investigated in this work, since [3] established that it is not an issue.

Due to the nature of discretization, adding oil according to equations (90) through (92), which are analytically derived, still does not quite introduce the amount of oil desired[19]. Hence, in practice, the TLOMM follows the above equations, calculates the oil it just added, and then makes a correction to the affected cells so that the amount added only differs from the amount desired by an error on the order of machine precision.

The procedure described above can perhaps be made clearer by an example. Figure 30 depicts snapshots of the oil film thickness immediately before and after a ridge of oil was introduced.

5.4   Some Programmatic Details

This section is intended mainly for future TLOMM developers who need to understand some of the "how" and "why" behind the source code. A user's guide (separate from this document) is also available.

The TLOMM was written in MATLAB, a high level programming environment offering a wide variety of built in mathematical functionality as well as post processing capabilities. The actual

---
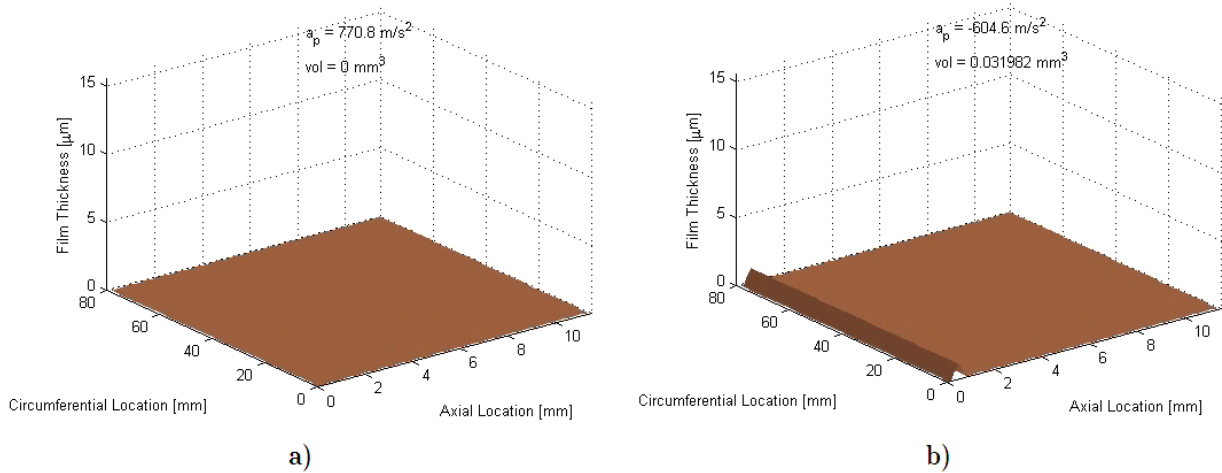
[19]Typically off by a few percent.

Figure 30: Demonstration of oil supply mechanism: a) before oil added, b) after oil added.

simulation is deployed as a standalone executable file; this way one does not need to have MATLAB installed to be able to run the program (the executable runs 2-3 times faster too).

5.4.1 Gas Velocity Input Data

The simulation developed in this project will not simply automatically work with any set of gas velocity data. As with all simulations, the input data files must follow a certain format, as outlined in Appendix B. The gas velocity data is assumed to follow a right handed, cartesian coordinate system which moves with the piston and has its origin at or slightly above the center of the piston crown. Additionally, it assumes that there are only two layers of grid points in the radial direction. It can tolerate some small variations in the regularity of the grid, but the more uniformly distributed the grid points (as measured in the TLOMM's own polar coordinate system), the less chance there is of incompatibility[20]. The simulation takes this input data and does several things, including rejecting cells which lie outside of the domain of interest (top land), adjusting for slight irregularities in the input data grid, and of course performing velocity and coordinate transformations. If all goes well, the data ends up in the framework of the TLOMM (Table 1) and is contained within a few large matrices.

5.4.2 Optimization

Since the TLOMM is designed to run for hundreds, perhaps even thousands, of engine cycles[21], it should come as no surprise that optimization was necessary to bring the program execution time

---

[20]Though it was never actually tested, it is doubtful that data falling on a completely unstructured grid could be successfully imported by the program's existing code.

[21]and it was written in MATLAB, which is generally viewed to be slower than Fortran or C code...

down to a reasonable value. Compared to other languages, MATLAB has the advantage of its built in Profiler, which can help a user quickly identify bottlenecks and speed up the code. For the sake of quoting some approximate time scales, the simulations whose results are presented in the following section had 6400 computational cells and used 0.4 degree time steps, and took roughly 2.5 minutes of real (not processor) time per engine cycle. The simulations were run on the author's Windows-based dual core Pentium IV PC, using the standalone executable version of the TLOMM. Countless tweaks were performed; only a select few of the optimizations are herein discussed.

In practice, one of the largest time sinks in the simulation was the task of calculating the gas velocities. This job requires the program to calculate the instantaneous velocity values (both circumferential and axial) at all cell interfaces, at each time step for which gas flow input data is available. In other words, since the data provided is fully variable over all three space dimensions and time, the program requires four dimensional interpolation each time step. The MATLAB interpolation routines, and hence the TLOMM, take advantage of substantial speed gains which are realized when the interpolated values are all requested at once on a uniformly spaced grid, but in the end the task is still costly compared to the rest of the code. To nearly completely take this bottleneck out of the loop, the simulation calculates all of the needed gas flows only once and just reuses those values every cycle, so long as the user chooses a time step which is a factor of 720^{22}. The program automatically detects whether or not this is true, and if it is, it stores all of the needed calculations into matrices at the beginning of the simulation. Adoption of this practice yielded a very significant speed increase. As such, users are *highly* recommended to choose a time step size which is a factor of 720.

The TLOMM simulation utilizes time steps whose sizes are fixed, but algorithms which automatically vary the time step size throughout a simulation certainly do exist. Variable size time steps can force the simulation to proceed very slowly through a time period in which the physics are taking place on very small time scales, yet quickly jump across large time periods when very little is happening. The usage of variable time step sizes often make sense for IC engine simulations. For example, in the author's own experience (as well as in [2]), ringpack friction simulations encounter most of their highly transient activity close to the bottom and top positions of the piston stroke, while near midstroke the conditions are practically steady state, allowing large time steps to be taken without difficulty. Nonetheless, the TLOMM simulation uses fixed size time steps. The main reason for this choice is that using adaptive time steps would have completely ruled out the possibility of the optimization discussed above, involving re-using the same gas flow data every cycle. Another reason is that the animations created by the post-processing routines would

---

[22]If 720 is an integer multiple of the time step size, then every cycle the simulation will request gas flow data at the exact same set of crank angle values. There is no need to recalculate them every cycle unless the input data itself is cycle-dependent (the author does not currently know of any engine manufacturers whose combustion CFD simulations run for a full cycle, let alone several cycles).

appear jumpy, due to the irregularity of the time step size, without some interpolation logic to regularize the time scale.

Another optimization which merits discussion is that of inlining some of the functions used by the subroutine "derivs.m," which calculates the instantaneous derivatives $\frac{dh}{dt}$ at all of the cells. The MATLAB Profiler identified function call overhead as being a large bottleneck. By replacing calls to various functions, such as the function that calculates the slopes for use in the linear reconstruction, with the functions' code (*directly* entered in derivs.m), function call overhead was reduced significantly and execution speed increased by a factor of about 2.5. Unfortunately, though it speeds up the progam, this practice introduces a painful loss in modularity. For example, the newest version of the TLOMM does not allow the user to choose the slope limiter; the MC limiter is simply hard-coded into derivs.m. It is most likely that in the near future, one inlined version of derivs.m for each available slope limiter will be created, to restore modularity while retain the speed increase.

5.4.3 Memory

The TLOMM is built to run long simulations, upwards of hundreds or even thousands of engine cycles. Unfortunately, storing the instantaneous film thickness distribution each time step, for simulations of this length, requires far more memory than what is installed in current PCs. The underlying data behind the results presented in Section 6, which span 200 engine cycles with 6400 computational cells and 0.4 degree time steps, comprises about $2 \cdot 10^9$ data points and takes up about 10 GB (gigabytes) of space per simulation. This data cannot simply be deleted on the fly during the simulation, since the post-processor needs it to create the assorted animations and plots of the results[23]. Nor can it all be stored in RAM.

To get around these limitations, algorithms which automatically save the data to disk at regular intervals were created. The frequency at which data is saved is a user setting, to make the TLOMM at least slightly machine independent. After saving the data to the hard drive, the TLOMM resets the matrix of film thicknesses; hence, by continually reusing its largest variable, the simulation never ends up needing more than around 100 MB of memory. The data is stored in a compressed format, and is retrieved automatically when necessary (during post-processing). Once postprocessing has been carried out and all animations and plots generated, one may delete the data so long as it is certain that it will not be needed again. The program does save a main output file once a simulation has completed, which includes much of the simulation data (with the exception of instantaneous film thickness distribution matrices).

---

[23]One *could* integrate the animation generation code into the main simulation itself, but it is better practice to keep the tasks modular and perform post-processing separately.

6 Results

The Top Land Oil Movement Model was applied to a real engine, and the evolution of the top land oil film thickness distribution was studied in detail. This section presents illustrative sample results which both demonstrate the TLOMM's capabilities and realistically predict the top land oil transport for the diesel engine studied in this project. A key physical mechanism which controls oil transport on the top land was discovered, and is discussed at length.

6.1 Sample Simulation

A full scale simulation using real engine data was performed. It spanned 200 engine cycles, and a cycle-to-cycle steady state was reached. The results are presented and analyzed in what follows.

6.1.1 Settings

Gas velocity data   A full set of CFD data of the combustion gas velocities in the top land crevice of a modern diesel engine was obtained. The input data spans a 180 degree portion of the engine cycle, and is fully three dimensional, spanning the crevice volume. A few representative plots of the input velocity field are shown in Figure 31. These plots are set at a viewing angle which is a side view of the piston; of course only one fifth of the full top land is shown, because in this case, there were five in-cylinder fuel sprays. Only the axial and circumferential velocity components are displayed (the TLOMM has no use for the radial velocities anyway). The gases are essentially quiescent at the beginning and end ($\sim 300°$ and $\sim 480°$) of the data range; combustion gases enter the crevice volume around $385°$.

Material properties   The dynamic viscosity of the oil on the top land was calculated as described in Section 3.6.1, using SAE 15w40A lubricant, assumed to be at a constant temperature of $325°$ C. A gas temperature of $2000°$ C was used, which is consistent with the maximum temperature of the combustion gases entering the top land (obtained from the combustion CFD simulation). The gas dynamic viscosity was set to $76.6 \cdot 10^{-6}$ Pa-sec, which is that of air at this temperature [19].

Engine   The 12 liter, six cylinder engine chosen was simulated at 1500 rpm. The piston acceleration was directed downward during $0° \le \theta \le 75°$, upward for $75° \le \theta \le 285°$, and downward again for $285 \le \theta \le 360°$ (roughly).
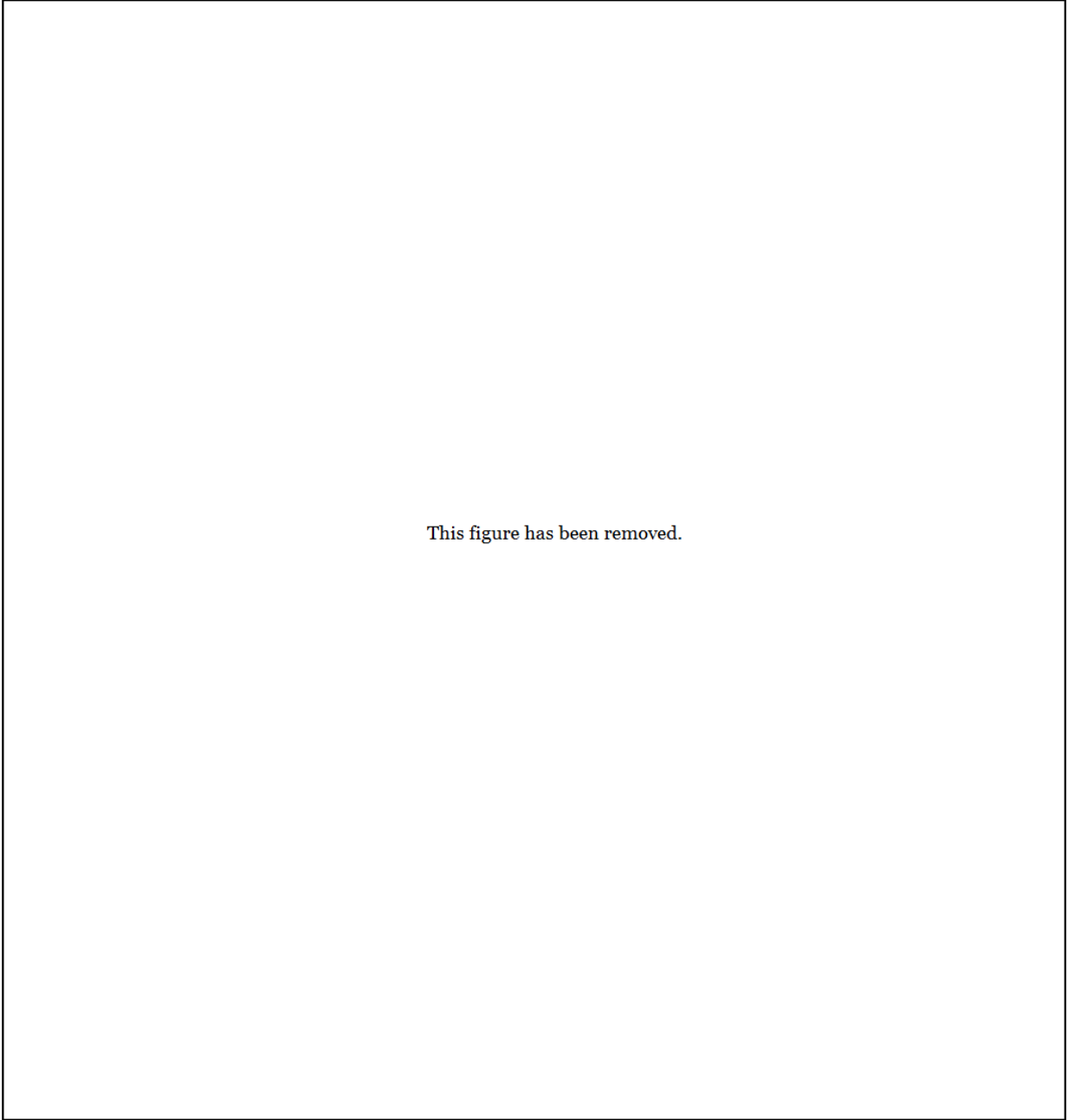
This figure has been removed.

Figure 31: Selected snapshots of gas velocity input data.

Computation settings The spatial discretization chosen had 80 cells in both the axial and circumferential directions (6400 cells in total). The time step size was 0.4 crank angle degrees. Timestepping was accomplished using the RK1 algorithm (78)[1], and the MC slope limiter was used.

Oil supply An engine oil consumption rate of 75 g/hr was assumed, and the amount supplied to the top land was set to twice this amount, knowing that some unknown percentage of the oil which reaches the top land would be ejected downward onto the top ring and into its groove[2]. Of course the amount introduced into the TLOMM domain was less, since the program accounts for the fact that it is only simulating one fifth of one of the cylinders rather than the entire engine. The initial film thickness distribution was set to zero in all cells, and a fixed volume of oil was introduced each revolution, according to the procedure described in Section 5.3.1. The sinusoidal ridge of new film had its lower edge 0.2 mm above the bottom of the top land, and its width was 0.58 mm. The boundary conditions at the top and bottom of the domain were set to zero, so that new mass could be added only via the sinusoidal ridge oil supply mechanism.

6.1.2 Results

In addition to creating a large volume of numerical data, the program developed communicates its results visually, using both animations and static plots.

Animations Animations of the oil film thickness distribution over the course of the 200 cycle sample simulation were generated[3]. Figures 32 and 33 show selected images taken from this animation. It is highly recommended that these figures be printed, or at least viewed, in color. For consistency, all of these snapshots are shown at the same phase within a cycle (TDC of exhaust/intake). Each pair of plots displays the film thickness distribution every 20 cycles. One can clearly see the film grow and spread up the land. The gas flows make the film growth very non-uniform circumferentially. After about 120 cycles, the distribution appears to be close to reaching steady state, and a large dry patch has formed, in which no oil resides.

---

[1]RK2 was not used because it was expected that the 80 x 80 grid is too coarse to take advantage of the higher order time stepper (see Sections 5.1.2 and 5.1.3).

[2]Another simulation was performed with only half of the oil supplied here; the results ended up being practically identical. It just took longer to reach steady state.

[3]The TLOMM has many postprocessing capabilities; the film distribution snapshots presented here do not do justice to the animations created by the program.

Initial

After 20 Cycles

After 40 Cycles

After 60 Cycles
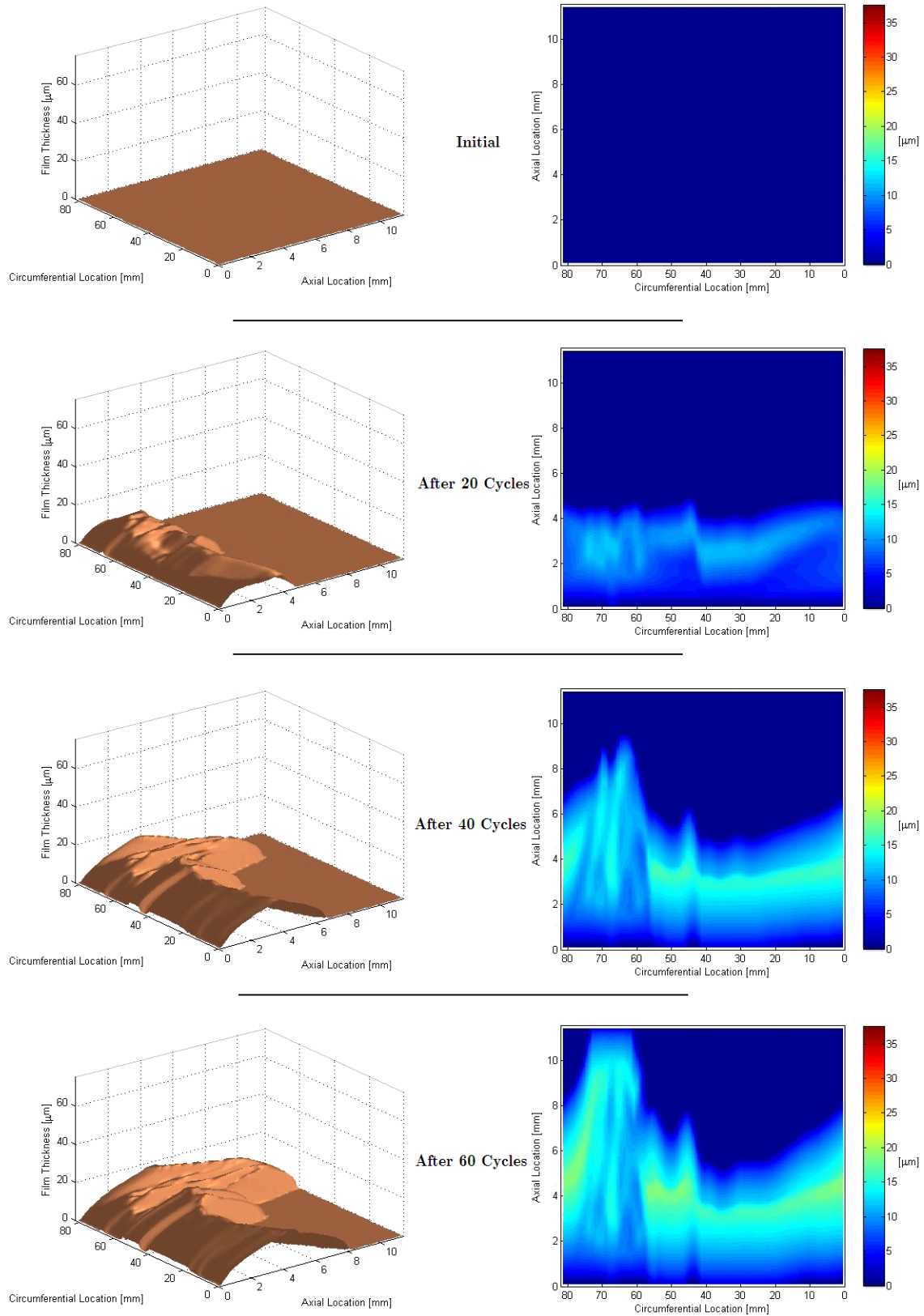
Figure 32: Evolution of oil film thickness for the sample simulation, part 1. All images correspond to piston at TDC position.
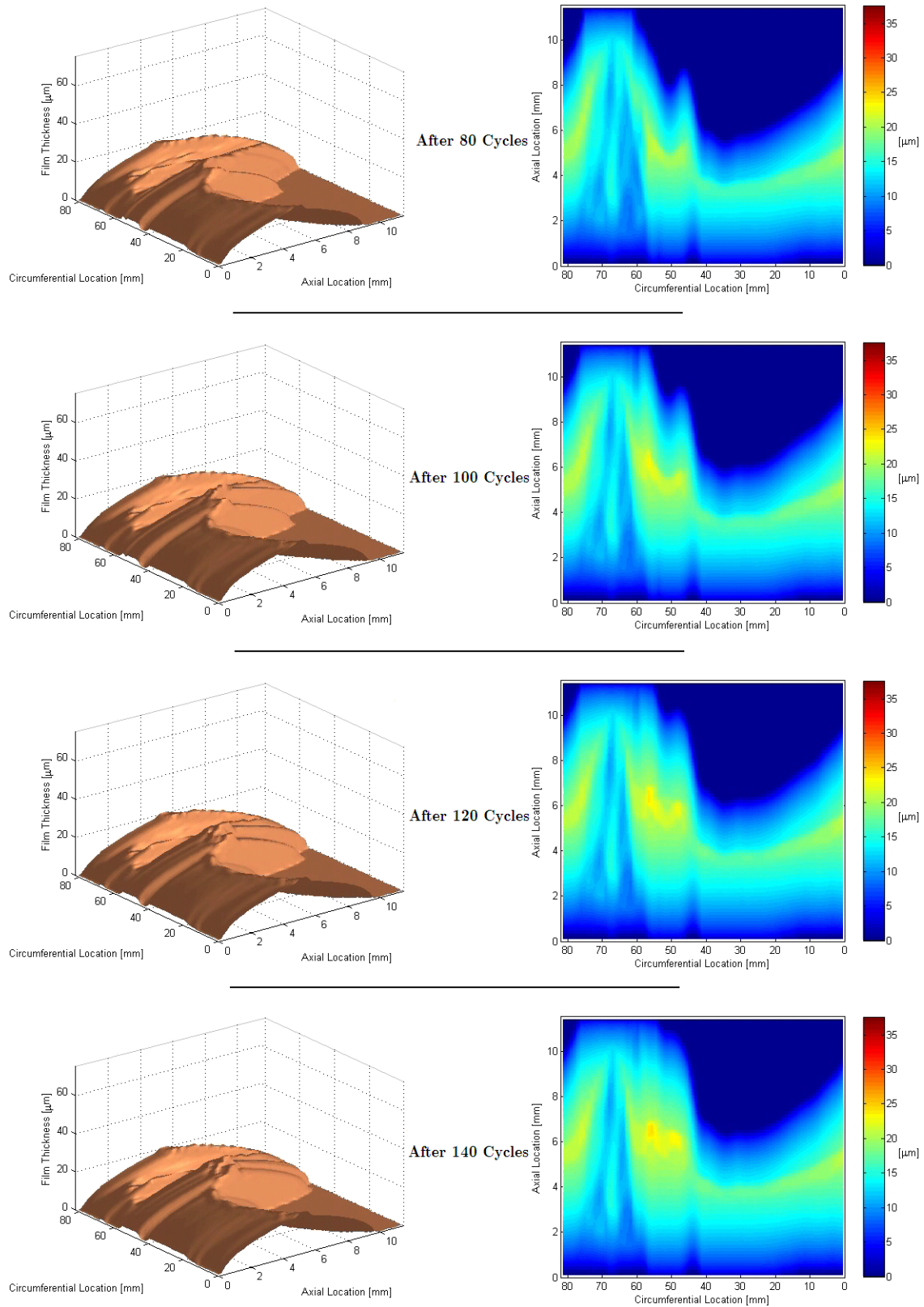
Figure 33: Evolution of oil film thickness for the sample simulation, part 2. All images correspond to piston at TDC position.

As the images shown in Figures 32 and 33 are all at the same phase within a cycle, it may appear to some that the oil film does not move back and forth, and simply grows in place. However, this (of course) is not the case; *within* each cycle the oil moves back and forth. To demonstrate some representative movement within one cycle, Figure 34 depicts a few images of the film distribution within just one period of the inertia force, clearly indicating that the oil is moving.

Figure 34: Demonstration of oil movement within one period of the inertia force.
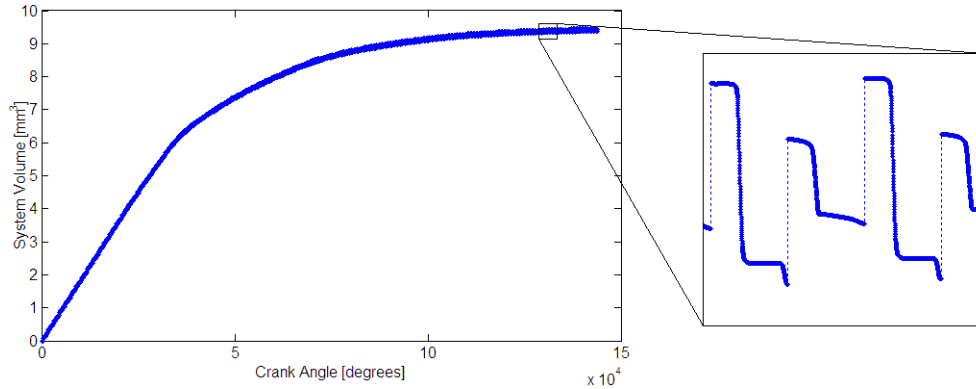
Figure 35: Instantaneous volume throughout sample simulation.
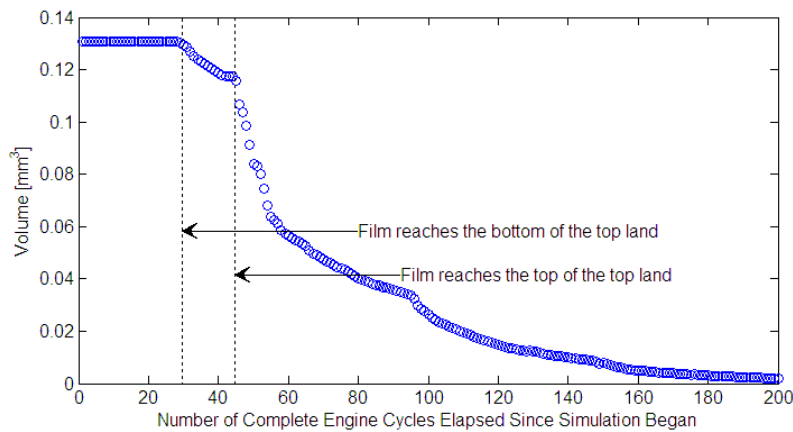
Figure 36: Net change in oil volume per cycle.

Summary Plots   A plot of the instantaneous volume of oil on the top land is displayed in Figure 35. The inset on this image shows that the volume fluctuates within each cycle, though on the whole it slowly tapers off and approaches an average value per cycle. Of course the large discontinuities that can be observed each r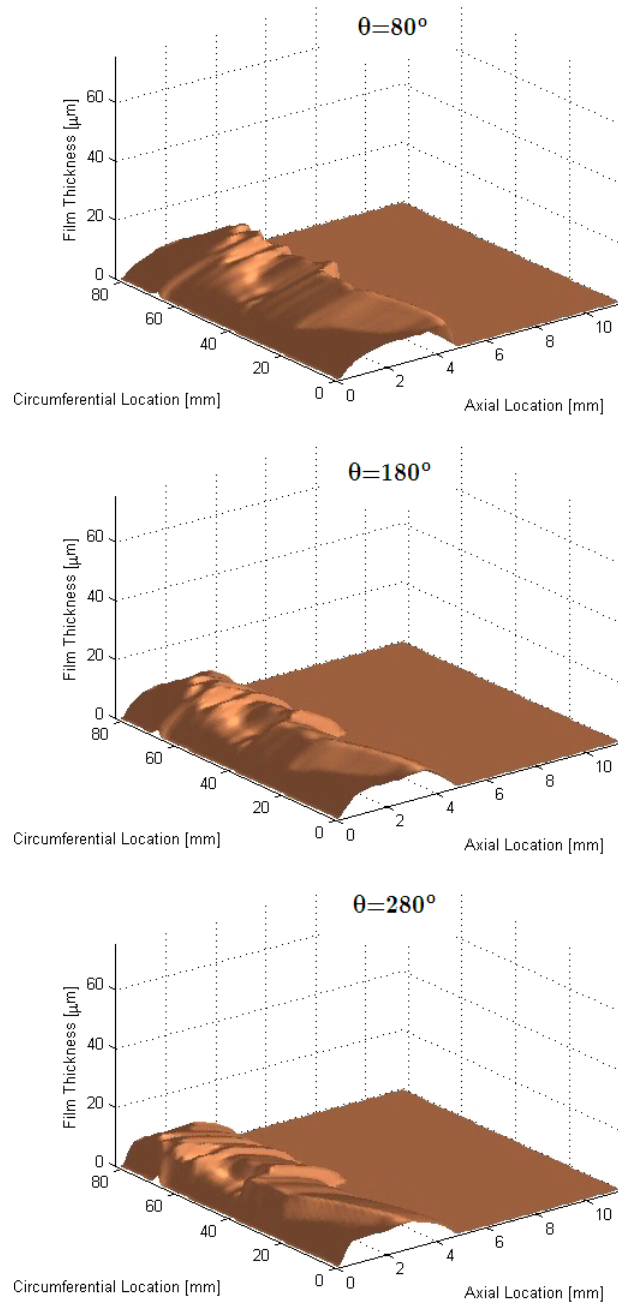evolution are simply due to the instances of a new ridge of oil being added at the bottom of the top land, as discussed in Section 5.3.1. The volume decreases whenever oil flows out through either the top or bottom of the domain.

Figure 36 presents the net change in oil volume per cycle. Since the oil will continue to move back and forth within each cycle, for as long as the simulation is run, there is no such thing as a true (i.e. quiescent) steady state. However, the TLOMM results do eventually reach a cycle-to-cycle steady state, which may be detected from this plot[4].

For the reader looking for more detail about where the oil went, Figure 37 displays a plot of the volume of oil ejected out of both of the domain's "real" (i.e. non-periodic) boundaries each cycle.

---

[4]Cycle-to-cycle steady state can also be roughly observed from the animations, but the cyclic mass balance plots are more quantitative indicators.

Figure 37: Volume ejected both above and below the top land per cycle.

It is in this way that the TLOMM can be used to do more than just study carbon deposits - it can also be used to study oil consumption, by predicting where the top land oil goes. Figure 37 shows t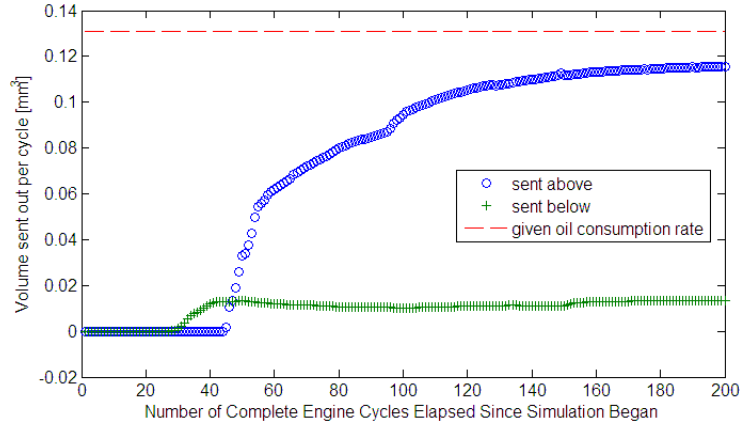he volume of oil which gets ejected out of the top of the top land (presumably sent into the combustion chamber) per cycle, as well as the amount of oil which flows out through the bottom of the top land (onto the top ring or into the top ring groove) per cycle. The dotted line simply represents the oil consumption rate originally entered by the user, for reference. It is corrected to account for the fact that the TLOMM only simulates a small part of the engine.

Figure 38 shows the instantaneous maximum film thickness throughout the entire simulation. A user can zoom in on parts of this plot and examine times during which shocks and gas flow bunch up effects (to be discussed) occurred, if desired[5]. Due to the compressive scale of the abscissa axis, the plot appears to have multiple values for each crank degree increment. However, this is not the case; the plot's two insets depict the typical instantaneous maximum film thickness variations within a cycle. Again, the jump discontinuities are due to the ridge of new oil being introduced every revolution. The sharp increases in maximum film thickness which are *not* discontinuous (more evident at later times), however, are due to a physical effect induced by the gas velocities.

6.1.3   Comparison to a Case Without Gas Flows

The simulation run which created the above results used detailed gas velocity data from a CFD simulation of the combustion chamber (and top land crevice) gases. The profound effect that the gas flows have on the oil film distribution may be immediately observed by comparing the above results with those of a simulation run in which the gas flows were turned off. Figure 39 presents

---

[5]Recall that in the case of the 2D simulations (which did not include gas flows), shocks were the only mechanism by which the film could spread out and $h_{\max}$ could ever change. In the 3D case, gas flows can do this as well.
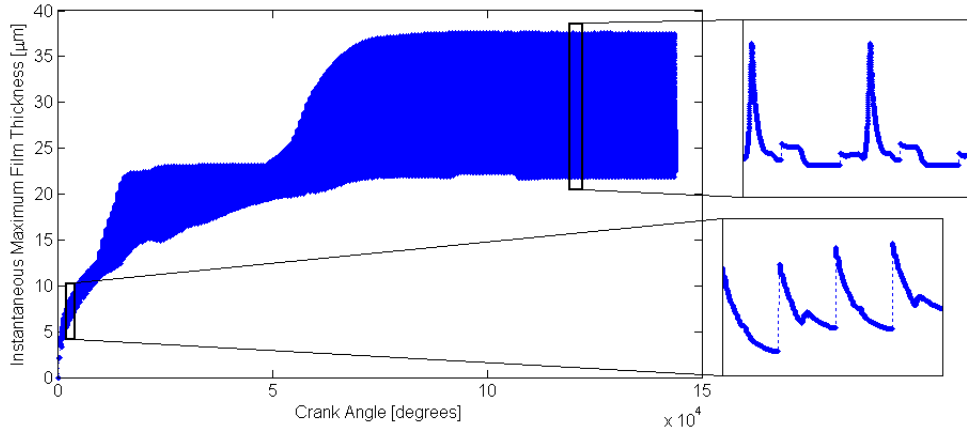
Figure 38: Instantaneous maximum film thickness throughout the full simulation.

the film thickness distribution at TDC once cycle-to-cycle steady state has been reached, for a simulation using just the inertia force. That is, the simulation settings for the results pictured in Figures 32 through 38 and Figure 39 were identical except for the "set all gas velocities equal to zero" toggle.

As one might expect, the results which do not include any gas flow effect are very different from the results including gas velocities. Without gas flows, the oil added every revolution is still spread by the inertia force, but nothing is there to create any circumferential non-uniformity. Comparison of the two simulation runs indicates that the gas flows can actually prohibit the oil from spreading up the top land; even by the time it reached steady state, the simulation which used gas flows had a large dry region covering roughly one third of the whole top land. The same cannot be said in the inertia-only case (Figure 39), for which the whole top land is eventually covered with oil.

From Figure 39, it may seem that there are two horizontal bands, at the top bottom of the top land, where there is no oil at all. This appearance exists simply because all of the images show the film distribution when the piston is at its TDC position. Even at cyclic steady state, the oil still moves back and forth within a cycle. Three snapshots taken roughly at the middle and both ends of one piston inertia period[6] are displayed in Figure 40. Being that a downward inertia period is examined here, the peak of the film starts at the top of the top land and moves to the bottom of the top land. The small bump protruding above the main puddle is due to the sinusoidal new ridge of oil introduced every revolution. Certainly the film thickness distribution at either end of the inertia period is very different from the distribution in the middle of the inertia period. In this "no gas flows" case, almost every point on the top land has a film thickness in excess of $20\,\mu$m at least twice per revolution.

---

[6]For this engine, a_p switches sign at roughly 75 and 285 CAD.
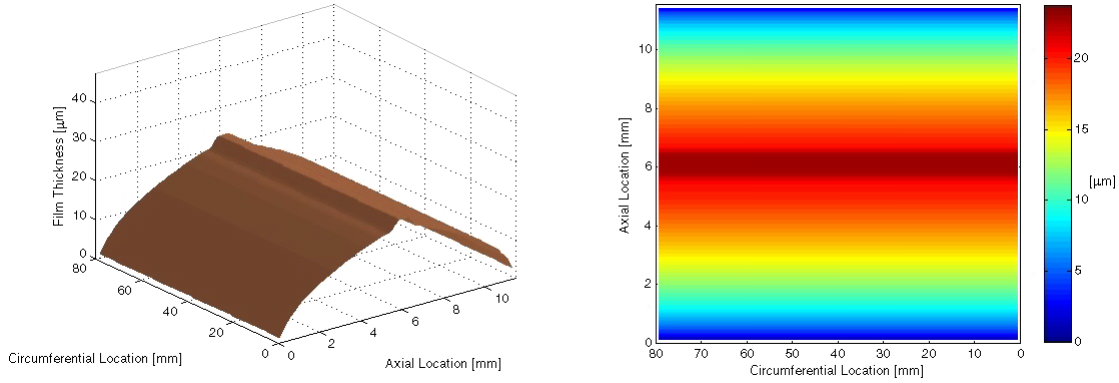
Figure 39: Steady state film thickness distribution when gas flows are disabled, with piston at TDC position.
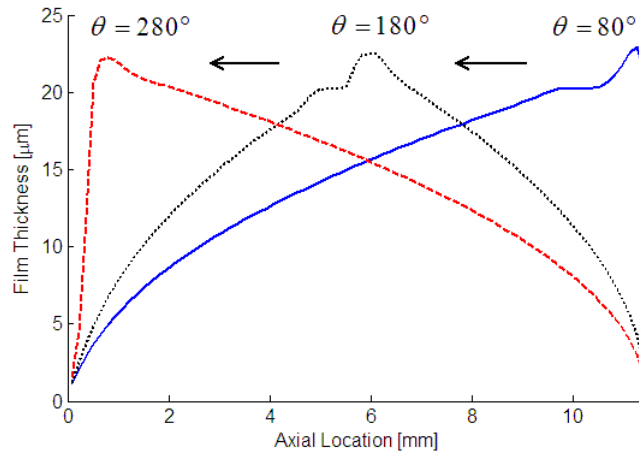
Figure 40: Film thickness distribution at various times within one inertia period, after cycle-to-cycle steady state has been reached, with gas flows disabled.

Overall, this comparison suggests that a higher percentage of the top land would be covered with oil for an engine which (somehow) has no crevice gas flows than one which has the crevice gas flows shown in Figure 31.

6.2   Key Finding of the Project

The "physical effect induced by the gas velocities" alluded to above, in the discussion of the mechanisms responsible for increasing the instantaneous maximum film thickness, deserves its own section. This effect turns out to be a subtle, yet powerful, oil rejection mechanism.

One implication of the variable coefficients in equation (22) is that source/sink-like terms (the

right hand side of equation (24)) exist. The terms are repeated here for convenience:

$$-\frac{1}{2}h^2\left[\frac{\partial}{\partial x}\left(\frac{\mu_{gas}}{\mu_{oil}}\left.\frac{\partial u_{gas}}{\partial z}\right|_{z=h}\right)+\frac{\partial}{\partial y}\left(\frac{\mu_{gas}}{\mu_{oil}}\left.\frac{\partial v_{gas}}{\partial z}\right|_{z=h}\right)\right]\ . \tag{93}$$

This expression represents a sum of the rate of change (in the axial direction) of the axial gas shear force and the rate of change (in the circumferential direction) of the circumferential gas shear force. It is *not* a sum of the full spatial gradients; the rate of change of the axial gas shear force in the circumferential direction, for example, is not present. At some point (x, y), when it is nonzero, expression (93) causes the local film thickness to either decrease or increase, depending on its sign. Of course, as the equation and numerical scheme are both conservative, the term is not a "source" in the classical sense[7].

The result of equation (24) having a nonzero right hand side is that in addition to simply altering the wave speeds, $-\frac{a_p}{\nu}h^2+h\frac{\mu_{gas}}{\mu_{oil}}\left.\frac{\partial u_{gas}}{\partial z}\right|_{z=h}$ and $h\frac{\mu_{gas}}{\mu_{oil}}\left.\frac{\partial v_{gas}}{\partial z}\right|_{z=h}$, the gas flows also have the ability to bunch up or flatten out local regions of the oil film. This observation may not seem very important, and one would probably assume that this effect is secondary to the augmenting of oil transport speeds. However, the gas flow gradient-driven bunch up effect ends up having vast consequences on the top land oil film distribution; the reason there is a large dry patch holding no oil in the steady state distribution shown in Figure 33 is exactly because of this gradient mechanism[8].

One can see from Figure 31 that the velocity field from the crevice gas CFD simulation contained some significant gradients. Specifically, there is a small region on the $\theta = 400°$ CA plot, roughly defined as $15 \leq y \leq 55$ and $x \approx 4$, in which there obviously exists a strong axial gradient in the axial gas velocities. There is also a region on the same plot, with $6 \leq x \leq 9$ and $y \approx 70$, in which there is a variation among the circumferential gas velocities along the circumferential direction. The gradients in both of these regions are convergent ($\frac{\partial}{\partial x}(u_{gas})$ and $\frac{\partial}{\partial y}(v_{gas})$ are both negative). According to the arguments and equations above (or simple intuition), one would expect the oil film to bunch up in these regions. This behavior is in fact what takes place.

Demonstration To illustrate the effects of the gas velocity gradients on the oil film, another simulation was prepared. The gas flow data set was the same as that used in the above simulation. The initial film thickness covering the top land was set to a completely uniform $9\,\mu$m layer. The film thickness available to flow in through the upper and lower boundaries was also $9\,\mu$m. In addition, the act of adding a new ridge of oil each revolution was deactivated. Hence, the only

---

[7]It neither adds nor removes mass from the system; it only pulls up or presses down (i.e., changes the shape of) the existing film.

[8]Of course spatial viscosity gradients could technically make expression (93) nonzero as well, but in this investigation viscosities were constant.

physics present which could possibly give texture to the initially flat film thickness distribution was the source/sink term, expression (93).

Figure 41 displays eight carefully chosen snapshots of the film distribution for this simulation. Sure enough, within the first 100 crank angle degrees of gas flow data[9], the flat oil film begins to acquire texture. At 400°, the beginnings of an oil ridge may be found, located in the region containing the large, convergent axial gas flow gradient already observed on the middle plot of Figure 31. This gradient is sustained in that same location for a while, which cannot be told from Figure 31. By 440°, the ridge is more pronounced and has stopped growing, and the inertia force has just begun acting on it. The piston acceleration is directed upward (and the inertia force downward) from then until about $\theta = 646°$, when it switches sign. The plot of the thickness distribution at $\theta = 646°$ shows that the inertia force has pushed the ridge down, close to the bottom of the top land. The following plot ($\theta = 75°$) displays the extent to which the film moves during the upward inertia period; at this crank angle the inertia force has just switched signs again and now begins to act downward (sending the ridge back toward the bottom of the top land).

Skipping ahead a few cycles, one readily observes from Figure 41 that the film continues to become more textured as the simulation progresses. The gas flows repeat the same "bunching up" effect every other revolution (once per cycle), and the size of the ridge grows[10]. The film distribution plot at $\theta = 504°$ in the 4th cycle is right in the heart of the downward inertia period. Examination of the slope of the oil film near the front edge of the ridge indicates that the film has indeed developed the curved **shock front** for which so much effort was invested in the numerical methods portion of this work. From the plot, the numerical algorithm shows no signs of shock-induced problems. Rather, it allows any number of shocks and rarefactions to interact with each other in arbitrary ways, as one would expect anyway from the validation efforts in Section 5.1. On the next image ($\theta = 646°$), the peak of the ridge is not as high as it was at $\theta = 504°$; it should be no surprise that the ridge spread out, since shock acts as a physical dissipation mechanism (discussed in previous sections). Mass has not been lost in the spreading out process; rather, once the ridge is set in motion, its leading edge develops into a shock and a rarefaction wave follows behind.

One should compare Figure 41's two $\theta = 646°$ plots, from the first and fourth cycles. In the first cycle, the ridge created by the gas gradients was not big enough to reach the bottom of the top land. However, by the fourth cycle, this ridge *was* tall enough to make it to the bottom of the top land. The reason the two cases are different is because the wave speed component due to the inertia force scales with h^2. Due to the boundary conditions applied in this particular

---

[9]Recall that in this simulation, crevice gas velocity data was only provided from 300° to 480°.

[10]The various textures which are forming might appear unphysically steep; keep in mind that the scale of the z axis (μm) is 3 orders of magnitudes smaller than the other axes (mm).

simulation, if the ridge does not make it to either of the axial boundaries, the mass contained in the ridge stays constant. However, if the ridge reaches one of these boundaries, some of its mass flows out, which is *not* reclaimed during the following (inflow through the boundaries) inertia period. The available film height flowing in through the boundary is less than the film height of the ridge when it was flowing out; the net effect is that mass is lost.

Of course, *where* the mass goes strongly depends on the *timing* of the occurrence of these gas velocity gradients. If they take place right before the crank angle when the piston inertia force begins to act downward, as was the case in this simulation, then the bunched up oil gets spread downward and eventually reaches the bottom of the domain, ejecting some amount of mass per cycle. However, if the gradients occur right before the crank angle when the piston inertia force begins to act upward, then obviously the ridge of oil created would be transported in the opposite direction, and eventually the oil would start being ejected out of the top of the domain (presumably into the combustion chamber).
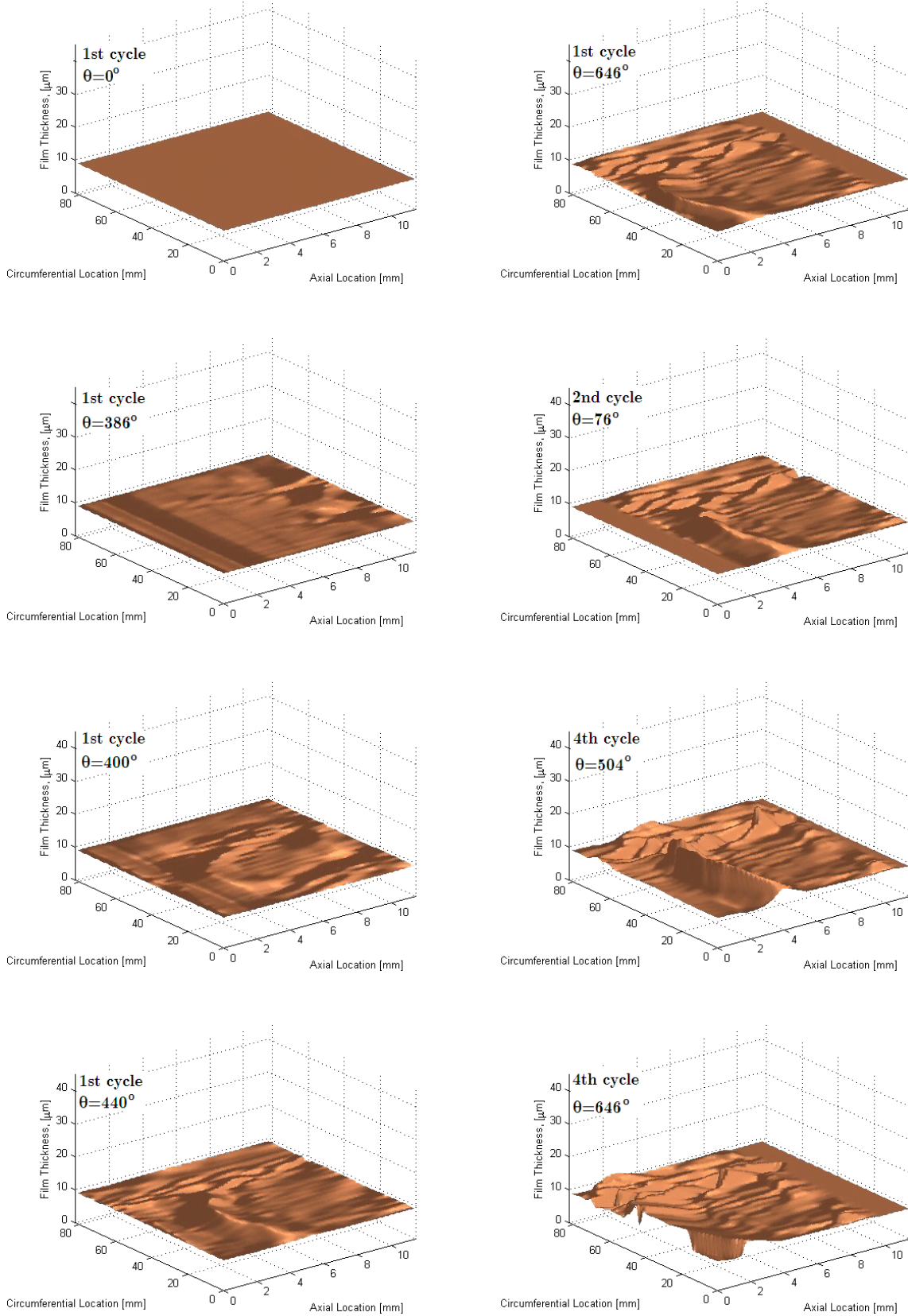
Figure 41: Demonstration of the oil ejection mechanism induced by gas velocity gradients.

By now it should be clear why there was a large dry patch in the images from the "realistic" simulation results shown earlier in Figure 33. The location of this dry patch was more or less above the previously discussed region on Figure 31, in which there was a strong convergent axial velocity gradient around $\theta = 400°$. The gradient effect continually bunched up some of the oil which happened to lie in this region just at the right time so that it got spread downward by the inertia force, rather than being allowed to spread upward much. In effect, by prohibiting the oil from spreading upward in one area, this large gradient caused the dry patch to exist (despite a consistent supply of new oil every revolution)[11]. Note that the dry patch is much larger than the region containing the large convergent gradient itself.

It is in this way that the gas velocity gradients act as a mechanism which indirectly controls the oil distribution on the land[12]. It is not the gradients themselves which spread oil in any preferential direction - that role is played by the inertia force. The gradients bunch up the oil, but the inertia force is essentially what is responsible for getting rid of the oil. Basically, by collecting the film into a ridge, the gradients amplify the effect of the inertia force, since the wave speed induced by inertia scales with h^2. The gas velocity gradient effect and inertia force complement each other; without one, the other could not be responsible for reducing the volume of oil on the top land.

Experimental Validation   According to the widely accepted notion that diesel piston land deposits come from the lubricant, one would postulate that for the engine being simulated, no carbon deposit formed on the dry patch observed above. As it turns out, in experiments performed on the engine studied in this work (for which the CFD data shown in Figure 31 was generated), a large patch on the top land was observed to have no carbon deposit. This deposit-less region was located in very much the same area as the dry patch predicted on Figure 33. At least for this engine, the TLOMM simulation results correlate surprisingly well with the carbon deposit distribution found experimentally[13].

Importance of the finding   The results presented indicate that the velocity gradients arising from the combustion gases in the top land crevice can have quite an effect on the distribution of oil on the top land. It is worth repeating the fact that all of the settings used for the simulation depicted in Figures 32 through 38 are realistic and represent an operating engine, to the best of our knowledge[14]. If one does have the ability to change the gas flow patterns in the top land

---

[11] In addition, the gradient also reduced the amount of oil being ejected into the combustion chamber.

[12] It should be stressed again that "gradients" is being used here as shorthand for "variation of the axial gas velocity in the axial direction and/or variation of the circumferential gas velocity in the circumferential direction." It does not mean the full gradient $(\frac{\partial}{\partial x} + \frac{\partial}{\partial y})$ of each component u and v; rather, it is the dot product $\vec{\nabla} \cdot \left(u\hat{i} + v\hat{j} \right)$.

[13] Unfortunately, photographs of the top land from these experiments cannot be displayed due to proprietary reasons.

[14] We of course assume that the input gas velocity data is at least somewhat accurate.

crevice, this gradient mechanism can be used to *control* top land oil transport and the resulting film distribution. Given the combination of the TLOMM simulation tool and a combustion CFD simulation (or experimental data, acquired using PIV for example), one could iteratively alter their power cylinder design in order to create a gas velocity field that forces most of the oil which happens to make it to the top land to be harmlessly ejected out of the bottom boundary[15]. This control strategy could be applied not only to mitigate carbon deposits, but also to reduce oil consumption.

---

[15]Predicting where oil exiting out of the bottom of the computational domain ends up going (e.g. top ring, top ring groove, etc.) is beyond the scope of this project.

7 Summary

A comprehensive simulation tool (the TLOMM) for predicting the oil film distribution on IC engine piston lands was created. The physical model was discussed at length, with topics including its strengths, weaknesses, and experimental validation. Quite a bit of effort was devoted to developing a numerical algorithm which could meet this project's requirements. The numerical approach taken was subjected to several validation exercises, and appropriate oil supply considerations were discussed. The simulation was applied to the top land of a modern heavy duty diesel engine, in order to better explain the nonuniform carbon deposit patterns observed in engine tests. The predicted steady state oil distribution corresponded surprisingly well with the experimentally observed carbon deposit distribution. New findings concerning gas flow-induced oil transport mechanisms were made.

The most important contributions of this project are summarized as follows:

- It was found that at least for some engines, gas velocities arising from the combustion process can in fact have a profound effect on the top land oil film distribution, and hence the distribution of carbon deposits.

- The underlying physical mechanism by which the gases can play such an important role is not brute force. Rather, it was discovered that the existence of spatial gradients in the gas velocity field cause the oil film to bunch up, allowing the powerful inertia force to easily eject oil from the land.

  - Careful placement and timing of these gradients could be used to control engine oil consumption.

- The TLOMM proved that modern high resolution shock capturing schemes may be successfully applied to free surface problems for which it is known *a priori* that the fluid film creates steep fronts.

- A robust algorithm for setting appropriate boundary conditions when solving scalar, hyperbolic conservation laws having variable coefficients and nonconvex fluxes was created, as described in Section 5.2.3.

One will note that the results of this project do in fact answer the central question posed in Section 1.1. The answer to "are the crevice gas velocities sufficient to push oil off of the top land" is yes. The *mechanism* by which they do so is not that which was initially expected.

Although the simulation works and its results so far seem to correlate well with a limited amount of experimental data, it should be stressed again that the physical mechanisms underlying carbon deposit formation are complex. Many important degradation processes, such as evaporation and oxidation, were not included in the model (though the existing model and simulation could easily be expanded to include these). The TLOMM's predictions can probably be trusted so long as the effects of degradation-induced flow property alteration are small relative to the driving forces modeled herein.

8 Future Work

There is enough complexity in carbon deposits to fill up at least a dozen (unique) Ph.D. theses. Many different approaches could be taken to attack the problem. Quantifying all of the kinetic pathways and rates would be an important achievement[1], as it would allow for simulations of the chemical degradation processes to be carried out with confidence. Repeating the fluid mechanics calculations *with* the inclusion of surface tension would shed more light on the accuracy of the predictions made by this model. Molecular dynamics simulations of the oil and additive molecules are certainly alluring; unfortunately, molecular vibration frequencies are so high that time steps on the order of 10^{-15} sec are typically required, making simulation of even one engine cycle highly infeasible.

Large scale ideas aside, there are more immediate things which could be done in continuation of this project.

- The sub-models for the material properties are not very general and could easily be improved. Currently the simulation just uses constant values for the oil and gas viscosities throughout the simulation, at all locations. However, dynamic viscosity is a relatively strong function of temperature. Gas temperature data is available from the combustion CFD simulation; it varies quite a bit spatially, as well as throughout each cycle. Various levels of improvement, proportional to the workload incurred, are possible:

  - One could account for only the temporal temperature variation, assigning some spatially averaged temperature each time step.
  - Some basic heat transfer calculations which predict the temperature at the oil/gas interface could be built into the TLOMM, for the sake of calculating both the air and oil viscosity at this location[2].
  - Since the oil film temperature certainly varies in the radial direction, so does the oil viscosity, meaning that equation (22) is an approximation. To fully capture the effect of the radial viscosity gradient[3], due to the temperature gradient, one could increase the number of independent variables by one and discretize in the z direction. This modification would not be trivial.

- The quantitative error analyses in Section 5.1.3 were only carried out on the 2D TLOMM. It would be interesting to see if the full 3D version is also essentially second order accurate,

---

[1] Certainly the open literature does not include the full knowledge base held by corporations, e.g. additive companies.

[2] Note that even if the gas flow is uniform, a viscosity gradient could trigger the "bunch up" mechanism. Consult equation (93).

[3] Which, the author feels, is overkill, given the other modeling assumptions made.

to determine the legitimacy of the argument defending the unsplit method chosen (Section 4.6).

- Strictly speaking, the Riemann solver shown in Section 4.4.3, equation (76), calls for the numerical algorithm to seek for an extremum in the flux between the values u_{0_L} and u_{0_R}. The program developed ignores this notion and simply chooses the minimum (or maximum, depending on the case) given the two endpoints, $f(u_{0_L})$ and $f(u_{0_R})$. Of course this practice does not change the scheme's conservative property; it only inhibits certain exceptional wave structures from being manifested. The author wonders how different the results would be if the Riemann solver was modified to use a true extremum-seeking algorithm.

- In this project, little was done to examine the importance of the shape and phasing of the oil ridge introduced by the supply mechanism every revolution.

  - Many tests performed in [3] found that the shape of the bump does not matter once the volume introduced per revolution is small compared to the volume on the top land. This project simply used those results, and assumed that the calculations are insensitive to the added puddle's shape.

  - It is certain that the phasing *does* matter - if the new oil was always introduced at the beginning of the downard inertia period, transport would be downward - but introducing the oil within the upward inertia period makes the most physical sense. Expecting oil to be splashed *up* to the top land during the downward inertia period would be unrealistic.

References

[1] Harten, A., 1983, "High Resolution Schemes for Hyperbolic Conservation Laws," *Journal of Computational Physics*, Vol. 49, p. 357.

[2] Tian, T., 1997, "Modeling the Performance of the Piston Ring-Pack in Internal Combustion Engines," Ph.D. Thesis, Department of Mechanical Engineering, MIT. Available at `http://dspace.mit.edu/`.

[3] Thirouard, B., 2001, "Characterization and Modeling of the Fundamental Aspects of Oil Transport in the Piston Ring Pack of Internal Combustion Engines," Ph.D. Thesis, Department of Mechanical Engineering, MIT. Available at `http://dspace.mit.edu/`.

[4] Goodman, J., and LeVeque, R., 1985, "On the Accuracy of Stable Schemes for 2D Scalar Conservation Laws," *Mathematics of Computation,* Vol. 45, No. 171, p. 15.

[5] Yokozeki, A., 2001, "Solubility of Refrigerants in Various Lubricants," *International Journal of Thermophysics*, Vol. 22, No. 4, p. 1057.

[6] LeVeque, 2002, *Finite-Volume Methods for Hyperbolic Problems*, Cambridge University Press.

[7] LeVeque, 1992, *Numerical Methods for Conservation Laws*, Birkhäuser Verlag.

[8] Toro, E., 1999, *Riemann Solvers and Numerical Methods for Fluid Dynamics*, Springer.

[9] Kevorkian, J., 1993, *Partial Differential Equations: Analytical Solution Techniques*, Chapman & Hall.

[10] Farlow, S., 1993, *Partial Differential Equations for Scientists and Engineers*, Dover.

[11] Clay Mathematics Institute. Available at `http://claymath.org/millennium/Navier-Stokes_Equations/`. Accessed 20 May, 2007.

[12] National Nanotechnology Initiative. Available at `http://www.nano.gov/html/facts/faqs.html`. Accessed 20 May, 2007.

[13] Holden, H., and Risebro, N., 2002, *Front Tracking for Hyperbolic Conservation Laws*, Springer.

[14] Guinot, V., 2003, *Godunov-type Schemes*, Elsevier.

[15] Oleinik, O., 1957, Discontinuous solutions of nonlinear differential equations, *Amer. Math. Soc. Transl. Ser.* 2, 26, p. 95.

[16] Peraire, J., et. al., 2003, MIT Course 16.920 Lecture Notes. Available at `http://ocw.mit.edu/index.html`. Accessed 20 May, 2007.

[17] Taylor, R.I., et. al., 1994, "The Influence of Lubricant Rheology on Friction in the Piston Ring-Pack," SAE Paper 941981.

[18] Heywood, J., 1988, *Internal Combustion Engine Fundamentals*, McGraw Hill Inc.

[19] Incropera, F., and DeWitt, D., 2002, *Fundamentals of Heat and Mass Transfer*, Wiley.

[20] Mills, A., 1999, *Heat Transfer*, Prentice Hall.

[21] Takata, R., 2006, "Effect of Lubricant Viscosity and Surface Texturing on Ring-pack Performance in Internal Combustion Engines," S.M. Thesis, Department of Mechanical Engineering, MIT. Available at `http://dspace.mit.edu/`. Accessed 20 May, 2007.

[22] Jocsak, J., 2005, "The Effects of Surface Finish of Piston Ring-pack Performance in Advanced Reciprocating Engine Systems," S.M. Thesis, Department of Mechanical Engineering, MIT. Available at `http://dspace.mit.edu/`. Accessed 20 May, 2007.

[23] Sweby, P., 2001, "Godunov Methods," in *Godunov Methods: Theory and Applications*, edited by Toro, E., Springer.

[24] Van Leer, B., 1974, "Towards the Ultimate Conservative Difference Scheme: II. Monotonicity and Conservation Combined in a Second-Order Scheme," *Journal of Computational Physics*, No. 14, p. 361.

[25] Van Leer, B., 1979, "Towards the Ultimate Conservative Difference Scheme: V. A Second-Order Sequel to Godunov's Method," *Journal of Computational Physics*, Vol. 32, No. 1, p. 101.

[26] Roe, P., 1986, "Characteristic-Based Schemes for the Euler Equations," *Ann. Rev. Fluid Mech.*, No. 18, p. 337.

[27] Press, W., et. al., 2002, *Numerical Recipes in C++*, Cambridge University Press.

[28] Tveito, A., and Winther, R., 1998, *Introduction to Partial Differential Equations, A Computational Approach*, Springer.

[29] Courant, R., Friedrichs, K., and Lewy, H., 1928, "On the Partial Difference Equations of Mathematical Physics," *Mathematische Annalen*, No. 100, p. 32.

[30] Schiesser, W., 1991, *The Numerical Method of Lines Integration of Partial Differential Equations*, Academic Press.

[31] Schiesser, W., 2000, *Lecture Notes on Partial Differential Equations.* Available at `http://www.lehigh.edu/~wes1/apci/`. Accessed 20 May, 2007.

[32] Lax, P., and Wendroff, B., 1960, "Systems of Conservation Laws," *Commun. Pure Appl. Math.*, No. 13, p. 217.

[33] Godunov, S., 1959, "A Difference Method for Numerical Calculation of Discontinuous Solution of the Equations of Hydrodynamics," *Math. Sbomik*, No. 47, p. 271.

[34] Shu, C., 1987, "TVB Uniformly High-Order Schemes for Conservation Laws," *Mathematics of Computation*, Vol. 49, No. 179, p. 105.

[35] Sod, G., 1978, "A Survey of Several Finite Difference Methods for Systems of Nonlinear Hyperbolic Conservation Laws," *Journal of Computational Physics*, No. 27, p. 1.

[36] Sweby, P., 1984, "High Resolution Schemes Using Flux Limiters for Hyperbolic Conservation Laws," *SIAM Journal on Numerical Analysis*, Vol. 21, No. 5, p. 995.

[37] Woodward, P., and Colella, P., 1984, "The Numerical Simulation of Two-Dimensional Fluid Flow with Strong Shocks," *Journal of Computational Physics*, No. 54, p. 115.

[38] Woodward, P., and Colella, P., 1984, "The Piecewise Parabolic Method for Gas-Dynamical Simulations," *Journal of Computational Physics*, No. 54, p. 174.

[39] LeVeque, R., 1997, "Wave Propagation Algorithms for Multidimensional Hyperbolic Systems," *Journal of Computational Physics*, No. 131, p. 327.

[40] Osher, S., 1984, "Riemann Solvers, The Entropy Condition, and Difference Approximations," *SIAM Journal on Numerical Analysis*, Vol. 21, No. 2, p. 217.

[41] Shu, C., 1988, "Total-Variation-Diminishing Time Discretizations," *SIAM J. Sci. Stat. Comput.*, Vol. 9, No. 6, p. 1073.

[42] Gottleib, S., and Shu, C., 1998, "Total Variation Diminishing Runge-Kutta Schemes," *Mathematics of Computation*, Vol. 67, No. 221, p. 73.

[43] Gottlieb, S., et. al., 2001, "Strong Stability-Preserving High-Order Time Discretization Methods," *SIAM Review*, Vol. 43, No. 1, p. 89.

[44] Harten, A., and Osher, S., 1987, "Uniformly High-Order Accurate Nonoscillatory Schemes, I.," *SIAM Journal on Numerical Analysis*, Vol. 24, No. 2, p. 279.

[45] Harten, A., et. al., 1987, "Uniformly High Order Accurate Essentially Non-oscillatory Schemes, III," *Journal of Computational Physics*, Vol. 71, No. 2, p. 231.

[46] Shu, C., and Osher, S., 1988, "Efficient Implementation of Essentially Non-oscillatory Shock-Capturing Schemes," *Journal of Computational Physics*, No. 77, p. 439.

[47] Liu, X., et. al., 1994, "Weighted Essentially Non-Oscillatory Schemes," *Journal of Computational Physics*, Vol. 115, p. 200.

[48] Jiang, G., and Shu, C., 1996, "Efficient Implementation of Weighted ENO Schemes," *Journal of Computational Physics*, Vol. 126, p. 202.

[49] Tadjeran, C., 2007, "Stability analysis of the Crank-Nicholson method for variable coefficient diffusion equation," *Communications in Numerical Methods in Engineering*, No. 23, p. 29.

[50] Tannehill, J., Anderson, D., and Pletcher, R., 1997, *Computational Fluid Mechanics and Heat Transfer*, Taylor & Francis.

[51] Glimm, J., 1998, "Three-Dimensional Front Tracking," *SIAM J. Sci. Comput.*, Vol. 19, No. 3, p. 703.

[52] Glimm, J., 1965, "Solutions in the large for nonlinear hyperbolic systems of equations," *Comm. Pure Appl. Math*, No. 18, p. 697.

[53] Chorin, A., 1976, "Random Choice Solution of Hyperbolic Systems," *Journal of Computational Physics*, No. 22, p. 517.

[54] Cockburn, B., and Shu, C., 2001, "Runge-Kutta Discontinuous Galerkin Methods for Convection-Dominated Problems," *Journal of Computational Physics*, Vol. 16, No. 3, p. 173.

[55] Cockburn, B., and Shu, C., 1998, "The Local Discontinuous Galerkin Method for Time-Dependent Convection-Diffusion Systems," *SIAM Journal on Numerical Analysis*, Vol. 35, No. 6, p. 2440.

[56] Strang, G., 1968, "On the construction and comparison of difference schemes," *SIAM Journal of Numerical Analysis*, Vol. 5, p. 506.

[57] Chavent, G., and Jaffré, J., 1986, *Mathematical Models and Finite Elements for Reservoir Simulation*, North-Holland.

[58] Zalesak, S., 1979, "Multidimensional Flux-Corrected Transport Algorithms for Fluids," *Journal of Computational Physics*, No. 31, p. 335.

[59] Crandall, M., and Majda, A., 1980, "The method of fractional steps for conservation laws," *Mathematics of Computation*, Vol. 34, p. 285.

[60] Shuaib, N., et. al., 2006, "A numerical study of wave structures developed on the free surface of a film flowing on inclined planes and subjected to shear," *International Journal for Numerical Methods in Engineering*, Vol. 68, p. 755.

[61] Burgess, J., et. al., 1968, "Chemistry of diesel deposit formation - a study of a model process," *Amer. Chem. Soc., Petrol. Chem. Div.*, Vol. 13, No. 2, p. B91.

[62] Burnett, P., 1992, "Relationship Between Oil Consumption, Deposit Formation and Piston Ring Motion for Single Cylinder Diesel Engines," SAE Paper 920089.

[63] Kim, J., et. al., 1998, "The Characteristics of Carbon Deposit Formation in Piston Top Ring Groove of Gasoline and Diesel Engine," SAE Paper 980526.

[64] Kelemen, et. al., 1998, "Fuel, Lubricant, and Additive Effects on Combustion Chamber Deposits," SAE Paper 982715.

[65] Wang, J., 1994, "Deposit Formation Tendency of Lubricants at High Temperatures," *STLE Journal*, Vol. 51, No. 5, p. 413.

[66] Covitch, M., et. al., 1988, "Microstructure of Carbonaceous Diesel Engine Piston Deposits," *STLE Journal*, Vol. 44, No. 2, p. 128.

[67] Papke, B., 1991, "High Temperature Diesel Piston Deposit Formation: Wetting and Adhesion Phenomenon [sic]," *STLE Journal*, Vol. 48, No. 3, p. 209.

[68] Zhang, Y., et. al, 1991, "A New Method to Evaluate Deposit-Forming Tendency of Liquid Lubricants by Differential Scanning Calorimetry," *STLE Journal*, Vol. 48, No. 3, p. 189.

[69] Obiols, J., 2003, "Lubricant Oxidation Monitoring Using FTIR Analysis - Application to the Development of a Laboratory Bulk Oxidation Test and to In-Service Oil Evaluation," JSAE Paper 20030124.

[70] Wang, J., Duda, J., and Klaus, E., 1994, "A Kinetic Model of Lubricant Deposit Formation Under Thin Film Conditions," *Tribology Transactions*, Vol. 37, No. 1, p. 168.

[71] Ra., Y., et. al., 2006, "Effects of Piston Crevice Flows and Lubricant Oil Vaporization on Diesel Engine Deposits," SAE Paper 2006-01-1149.

[72] Kudish, I., et. al., 2005, "Kinetics Approach to Modeling of Stress-Induced Degradation of Lubricants Formulated with Star Polymer Additives," *Tribology Transactions*, Vol. 48, p. 176.

[73] Nishiwaki, K., and Hafnan, M., 2000, "The Determination of Thermal Properties of Engine Combustion Chamber Deposits," SAE Paper 2000-01-1215.

[74] Cho, Y., and Tian, T., 2004, "Modeling Engine Oil Vaporization and Transport of the Oil Vapor in the Piston Ring Pack of Internal Combustion Engines," SAE Paper 2004-01-2912.

[75] Zerla, F., and Moore, R., 1989, "Evaluation of Diesel Engine Lubricants by Micro-Oxidation," SAE Paper 890239.

[76] Boschert, T., 2002, "The Lubricant Contribution to Future Low Emission Engine Design," Diesel Particulate and NOx Emissions Course (University of Leeds), Ann Arbor, MI.

[77] Barnes, J., and Bell, J., 1988, "Laboratory Screening of Engine Lubricants for High Temperature Performance," *Lubrication Engineering*, Vol. 45, No. 9, p. 549.

[78] Alizadeh, A., and Trimm, D., 1985, "The Formation of Deposits from Oil Under Conditions Pertinent to Diesel Engine Pistons," *J. Chem. Tech. Biotechnol.*, Vol. 35A, p. 291.

[79] Hunter, M., et. al., 1992, "Kinetic Study of Antioxidation Mechanisms," *Lubrication Engineering*, Vol. 49, No. 6, p. 492.

[80] Korcek, S., 1986, "Determination of the High-Temperature Antioxidant Capability of Lubricants and Lubricant Components," *Ind. Eng. Chem. Prod. Res. Dev.*, Vol. 25, p. 621.

[81] Klaus, E., et. al., 1991, "Thin-Film Deposition Behavior of Lubricants as a Function of Temperature: Lubricant Stability Maps," *Lubrication Engineering*, Vol. 48, No. 7, p. 599.

[82] Naidu, S., et. al., 1986, "Kinetic Model for High-Temperature Oxidation of Lubricants," *Ind. Eng. Chem. Prod. Res. Dev.*, Vol. 25, p. 596.

[83] Egharevba, F., and Maduako, A., 2002, "Assessment of Oxidation in Automotive Crankcase Lube Oil: Effects of Metal and Water Activity," *Ind. Eng. Chem. Res.*, Vol. 41, No. 14, p. 3473.

[84] Lee, C., et. al., 1992, "Evaluation of Deposit Forming Tendency of Mineral and Synthetic Base Oils Using the Penn State Microoxidation Test," Lubrication Engineering, Vol. 49, No. 6, p. 441.

Appendix A: Lubricant Properties for Typical Multi- and Single-Grade Oils

In the simulation developed in this work, oil viscosity's temperature dependence is roughly accounted for using the methodology presented in Section 3.6.1. The figure below displays typical constants for a variety of engine oils.

Oil Grade	k (cSt)	θ_1 (°C)	θ_2 (°C)	μ_∞ / μ_0	c_1	c_2 (°C^{-1})
0W40	0.01341	1986.4	189.7	0.67	2.5	0.026
5W20	0.04576	1224	134.1	0.94	2.5	0.029
5W40	0.15	1018.74	125.91	0.8	2.3	0.0225
10W30	0.1403	869.72	104.4	0.76	2.3	0.0225
10W50A	0.0352	1658.88	163.54	0.49	2.43	0.0218
10W50B	0.0507	1362.4	129.8	0.52	2.28	0.0269
15W40A	0.1223	933.46	103.89	0.9	2.3	0.0225
15W40B	0.03435	1424.3	137.2	0.79	2.5	0.026
20W50	0.0639	1255.46	117.7	0.84	2.3	0.0225
SAE10	0.0258	1345.42	144.58	1	2.3	0.0225
SAE30	0.0246	1432.29	132.94	1	2.3	0.0225
SAE50	0.0384	1349.94	115.16	1	2.3	0.0225

Parameters for calculating lubricant properties using the Vogel and Cross equations.

The variables c_1 and c_2 correspond to a and b in the expression for the critical shear rate, equation (42). One should note that k is in centistokes, which is a unit of kinematic viscosity. The Vogel equation, (43), uses dynamic viscosity. The TLOMM makes this conversion internally; it assumes that the user got the value of k from this table, and multiplies k by 850 (this table was prepared assuming all of the oils have a density of $850 \, \text{kg/m}^3$) as well as a unit conversion factor.

Appendix B: Required Gas Velocity Data File Format

The TLOMM simulation imports gas velocity data detailing the temporally and spatially resolved flow field in the top land crevice. The format of the input files is not arbitrary.

Three files (one for each velocity component) must be provided, all having the format shown in the table below.

text	text	text	text			
text	text	text	text			
text	text	text	text			
text	text	text	text			
any number	any number	any #	x coordinate of node 1	x coordinate of node 2	⋯	x coordinate of node n
any number	any number	any #	y coordinate of node 1	y coordinate of node 2	⋯	y coordinate of node n
any number	any number	any #	z coordinate of node 1	z coordinate of node 2	⋯	z coordinate of node n
crank angle @ step 1	time at step 1	any #	velocity at node 1	velocity at node 2	⋯	velocity at node n
crank angle @ step 2	time at step 2	any #	velocity at node 1	velocity at node 2	⋯	velocity at node n
. . .	. . .	. . .	. . .	. . .	. . .	. . .
crank angle @ step N	time at step N	any #	velocity at node 1	velocity at node 2	⋯	velocity at node n

Input data file format for gas flow field.

The data should be delimited by white spaces (commas might work too). A few rows of descriptive text may be placed at the top of the file; these are ignored. The first three lines of numeric values should present the node coordinates of the mesh (in meters). All subsequent rows should contain the velocity data (in m/s); each new row is for a new time step. Velocities are expressed relative to the liner. Column three can be any collection of numbers; the TLOMM ignores it. It only exists because the data files supplied in this project included a column reporting the number of computational iterations made by the CFD simulation, which the oil film simulation obviously does not need. Crank angle should be reported in degrees ($0-720$ scale), and time in milliseconds.