Pacific
Institute
for the Mathematical Sciences
http://www.pims.math.ca
pims@pims.math.ca

Proceedings of the Fifth

# PIMS Industrial Problem Solving Workshop

## PIMS IPSW 5

Co-sponsored by:

**The Natural Science and
Engineering Research Council
of Canada**

and

**The Alberta Science and
Research Authority**

Editor: J. Macki, University of Alberta

# Proceedings of the Fifth Annual PIMS Industrial Problem Solving Workshop

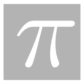
Editor: J. Macki, University of Alberta

Co-sponsored by:
The Natural Sciences and Engineering Research Council of Canada
The Alberta Science and Research Authority

NSERC
CRSNG

Alberta
INNOVATION AND SCIENCE

June, 2001



# Foreword by the PIMS Director

The Fifth Annual PIMS Industrial Problem Solving Workshop was hosted by the Department of Mathematics at the University of Washington in Seattle, June 18–22, 2001. For a full week more than 90 participants worked intensely on six problems posed by industrial companies from across North America.

The six problems came from *Microsoft Research, Firebird Semiconductors, Communications Security Establishment, Alberta Energy Company, IBM,* and *Algorithmics*.

PIMS looks forward to the next Industrial Problem Solving Workshop which will be held at the University of British Columbia in Vancouver next year.

Special thanks go to Jack Macki from the University of Alberta who edited these proceedings. I would also like to thank all the organizers and mentors.

Dr. Nassif Ghoussoub, Director
Pacific Institute for the Mathematical Sciences

# Contents

# List of Figures

# List of Tables

# Preface

The Fifth Annual PIMS Industrial Problem Solving Workshop, the IPSW-5, took place on the University of Washington campus from June 18 through 22, 2001. This was the first major PIMS event south of the border, and reflects the recent inclusion of the University of Washington as a member of PIMS. It was a great pleasure for our University to host this important workshop and many of our own students took advantage of the opportunity to participate in a very stimulating and educational program.

In all, about 100 people registered,including 58 graduate students who had taken part in the Graduate Modeling Camp at the University of Victoria the week before. Faculty from several universities around the world brought to the workshop a broad range of mathematical expertise. Most of the industrial participants were able to stay all week, and were actively involved in working with their groups. From their standpoint too it appeared to be a great success.

This was my first involvement in an IPSW or any study group of this nature, and I was very impressed by the enthusiastic manner in which all participants attacked the problems, and with the progress made. While I played the role of local organizer along with Tatiana Toro, our PIMS Site Director, I mostly worried about mundane aspects such as providing sufficient quantities of coffee and chalk (mundane but crucial). Most of the mathematical organization was carried out by an experienced crew of IPSW veterans,ably led by Marc Paulhus as the primary facilitator during the week.He and the rest of the Organizing Committee (Chris Bose, Huaxiong Huang, Ian Frigaard, and Keith Promislow) did a great job in lining up challenging industrial problems and knowledgeable mentors, insuring an interesting variety of problems for participants to choose from.

Chris Bose, who organized the Graduate Modeling Camp at UVic the week before, was also heavily involved in the logistics. Although UW and UVic are only about 150 kilometers apart as the crow flies, they are separated by both Puget Sound and an international border. Managing the transportation and diplomatic needs of 100 participants with dozens of nationalities was no small feat, but all went smoothly thanks to the efforts of Marc, Chris, and the PIMS staff.

Locally at UW, Tatiana and myself were greatly aided by Mary Sheetz and others in the Mathematics Department office. Michael O'Connell and his staff at the Mathematical Sciences Computing Center provided computer lab space and support for the participants. The computers were heavily used around the clock by the end of the week as results were computed and presentations prepared.

Finally, I would like to thank Jack Macki for taking on the substantial task of editing these proceedings. This traditionally falls to the local organizer, but with my sabbatical about to start Jack was kind enough to take pity on me and apply his expertise to insure that this excellent set of proceedings would appear in a timely manner.

Randy LeVeque
Department of Applied Mathematics
University of Washington
Seattle, Washington

## Acknowledgments

- The 5 report writers devoted a staggering amount of their own time to putting the results of the workshop into a readable form. They were:
    - Nancy Ann Neudauer: "The Disc Layout Problem"
    - C. Sean Bohun: "Modelling InSb Czochralski Growth"
    - Brian Alspach: "Adapting Search Theory to Networks"
    - Randall Pyke: "An Automated Algorithm for Decline Analysis"
    - Alan King: "Web Hosting Service Level Agreements"
    - Selly Kane, Viktoria Krupp & Jack Macki: "Monte Carol Methods in the Integrated market and Credit Risk Portfolio Model"

Jack Macki, Editor,
Department of Mathematical Sciences,
University of Alberta,
Edmonton, Alberta T6G 2G1
jmacki@gpu.srv.ualberta.ca
Fax: (780) 492-6826, Phone: 492-5725

# Pimslips

When people are working intensely, they sometimes say things that, in retrospect, are amusing or downright hilarious. Here are some from IPSW5:

- “Yes, take photographs please” (during a technology-challenged talk)
  John DeTreville

- Tim: “They shipped my stuff to Capetown”
  John: “How did they ship it?;;
  Tim: “By ship”
  Tim Myers & John Stockie

- “In 1D volume is not a problem”
  Victoria Krupp

- Marc: “Is that micro-seconds or milli-seconds?”
  Nancy: “Milliseconds.”
  Marc: “Not Microseconds?”
  Nancy: “Microsoft-seconds”
  Marc Paulhus & Nancy Ann Neudauer

- “Probability zero means it is as unlikely as you like”
  Theodore Kolkohlinkov

- “That is not constant enough”
  Ian Friggard

- John: “How long is this simulation going to take? To the end of the world?”
  Alex: “For us the end of the world is Thursday”
  John Chadam & Alex Kreinin

- “It is $x/2$ but it is really $x$”
  Nancy Ann Neudauer

- “$Ta(y)$ is an unknown known function”
  Rex Westbrook

- “So I should expect some unexpected slides”
  Nancy Ann Neudauer

- Q: “Who is your roommate?”
  A: “I paid extra for a single room... you never know who you are going to end up sleeping with”

- Sean: "I'm not sure if the surface is wetting itself"
  J.F. "I'm doing it by hand. I'm trying to shoot manually"
  Sean: "We have all done that"
  Sean Bohun & J.F. Williams (winner)
- "$\alpha$ is really small... less than $1/\epsilon$ where $\epsilon$ is really really small"
  Huaxiong Huang

# Chapter 1

# The Disk Layout Problem

Brian Corbett[1], Gregory Dresden[2], Nancy Ann Neudauer[3], Marc Paulhus[4],
Report prepared by Nancy Ann Neudauer

We can organize data on a personal computer's hard drive according to many different data strategies resulting in different performances due to disk latencies, consisting of both rotational latency and seek time. Rotational latency is a physical characteristic of the disk and motor, so we focus on the problem of storing data in a manner that optimizes the seek time of the data. The optimization of this problem will result in better performance for users.

## 1.1 Introduction

Imagine that we keep a daily log of the files that our computer reads from its hard disk. For most computer users the logs of one day compared to the next may be very similar. For example, opening up a commonly used program may require access to the same files in the same order every time that event occurs. We shall call such a sequence of files a *trace*. Our daily log is composed of a large number of different traces. However, there is good reason to belive that some traces will appear largely unchanged in our log from day-to-day and perhaps multiple times in a single daily log.

Now imagine that our disk is a random ad-hoc jumble of files in no particular order (this should not be too hard for most of us to imagine). Our computer, performing the tasks we ask of it, may have to work very hard to access the files in the order that they are required. If files that are adjacent in a common trace are stored far apart on the disk then we should expect that our disk performance will be poor. On the other hand, if we rearranged our disk in such a way that those files were close together, we should expect improved performance.

This is the essence of the problem Microsoft posed to the PIMS 5th Industrial Problem Solving Workshop. Given a set of traces that are expected to be representative of common use, we must rearrange the files on the disk so that the performance is optimized.

---

[1] University of Manitoba

[2] Washington and Lee University

[3] Pacific University

[4] Pacific Institute for the Mathematical Sciences

Some immediate observations are clear. One is that all the parts of a single file should be contiguous (assuming that the computer only has uses for complete files). A second is that it can not help our disk performance to have gaps of data on our disk; gaps can only increase the distances between files.

Programs called *disk defragmenters* use these simple principles to rearrange data records on a disk so that each file is contiguous, with no holes or few holes between data records. Some more sophisticated disk defragmenters also try to place related files near each other, usually based on simple static structure rather than a dynamic analysis of the accesses. We are interested in more dynamic defragmentation procedures.

We first consider a 1D model of the disk. We then look at the results from an investigation of the 2D disk model followed by a discussion of caching strategies. Finally we list some of the complications that may need to be addressed in order to make the models more realistic.

## 1.2 1D Disk Layout Model

One way to model the disk is to imagine it as having only a single (circular) track, with blocks on that track labeled $B_0, B_1, \ldots, B_n$ where block $B_n$ is followed by block $B_0$, then block $B_1$, and so on, creating a cycle. The *files*, say $D_0, D_1, \ldots D_k$, are placed inside these blocks. The *head* sits in a fixed location and the disk spins (in one direction, for our purposes counterclockwise). The head can read the file that is directly beneath it. See Figure 1.1.

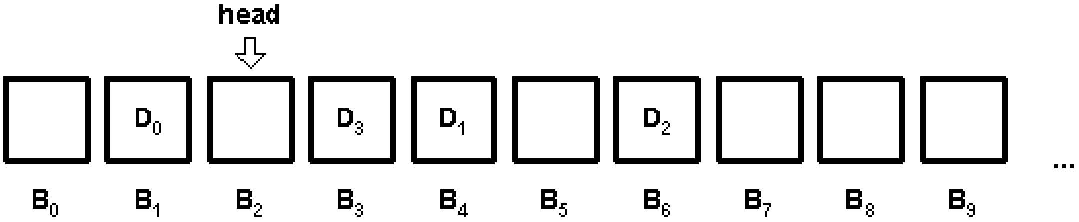


Figure 1.1: A one-dimensional array of fixed-sized blocks.

Our task is to rearrange the files that are assigned to each block to minimize the *cost* on a given trace. The *cost* is simply a count of the number of blocks which must pass under the head while it is reading the given trace.

Suppose that the trace in question is $\{D_0\ D_1\ D_2\ D_3\}$. Then from the starting position shown in Figure 1.1, the cost is 21. Is there a better layout that would reduce the cost?

In this case, of course, the solution is obvious. Since there is only one trace and each file appears exactly once, the optimal data layout is the trace itself, as shown in Figure 1.2. The cost of executing the trace is now just 3.

When there is more than one trace or when the same file appears multiple times in the same trace then the situation gets more complicated. As a model for this scenario, consider a complete directed graph where the nodes are the files and each directed edge $(F_1, F_2)$ is assigned a cost function based on the number of times that file $F_1$ is followed by file $F_2$ in the given trace.

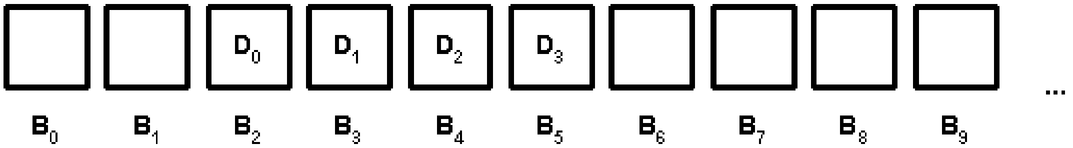


Figure 1.2: An optimal data layout for $\{D_0\ D_1\ D_2\ D_3\}$ trace

A good layout to consider is the maximal tour on this graph. Thus we can see that the problem is closely related to the famous NP-hard traveling salesman problem (see, for example, Cormen, Leiserson and Rivest, *Introduction to Algorithms*, MIT Press (1995), pp. 969).

This 1D disk model is not a particularly accurate representation as most disks consist of a number of concentric tracks, sometimes on both sides of the disk, and sometimes with a platter of disks stacked one atop the other. However, it is an unfortunate reality that most disks in use today do not reveal their precise geometry to the operating system. Instead, they reveal a 1D geometry not unlike our simple linear model. In this common case the 1D model is the only option available.

We now describe some heuristic methods we use to investigate the problem.

The following assumptions are made:

1. All files are the same size and will fit exactly in one block.
2. The disk is completely packed. That is, there are no empty blocks (this is relaxed slightly by necessity in Section 1.3).
3. The disk spins at a constant rate.
4. There is a cache of size one. That is, if file $F_1$ has just been read and the trace asks to read $F_1$ again, then there is no cost for this. More on caching will be discussed in Section 1.4.
5. Every file appears on the disk exactly once. It may seem tempting to duplicate commonly used files to improve disk performance. However, if overused this technique will quickly fill a disk. Also, the time required to update or change a file will increase.

### 1.2.1 1D Disk Layout Results

Consider a set of fixed traces, each consisting of a certain number of files. We seek a new arrangement of these blocks such that the cost function, applied to each trace, is reasonably low.

If we had world enough and time (to quote Andrew Marvell) we could look at every possible permutation of blocks, calculate the cost of each trace on each permutation, and thus find the best arrangement. This is obviously impractical, so we need to come up with a faster way to calculate the cost, and a better way to find a good arrangement of blocks.

Let us first define and discuss the adjacency matrix, which gives us a quick way of judging the worthiness of a particular configuration of the blocks. We define $A$ to be an $n \times n$ matrix, initially all entries 0, and indexed by the blocks in the trace. For each consecutive pair of blocks $i, j$ in the traces, we increment the corresponding matrix entry $A_{i,j}$ by 1. So, given the trace $T = \{d, c, b, a, d, b, c, a, a, d, c, a, d, d, d, b, a, c, b, a\}$, and with rows and columns labelled in the order $a, b, c, d$, the matrix A is

$$\begin{array}{c|cccc}
 & \text{a} & \text{b} & \text{c} & \text{d} \\
\hline
\text{a} & . & & 1 & 3 \\
\text{b} & 3 & . & 1 & \\
\text{c} & 2 & 2 & . & \\
\text{d} & & 2 & 2 & .
\end{array}$$

We replace the diagonal entries with "." both for ease of reading and to illustrate that there is no cost associated with accessing the same block twice in a row.

Clearly, the initial block configuration of $a, b, c, d$ for the trace $T$ (with a cost of 41) is far from optimal here: we see from our matrix that $a$ is never followed by $b$, nor is $c$ followed by $d$ (as $A_{a,b} = A_{c,d} = 0$). However, the pair $b, a$ occurs three times, as $A_{b,a} = 3$. We seek a block configuration that gives an adjacency matrix with large numbers on the upper diagonal and small numbers on the lower diagonal, thus indicating that commonly-occuring (respectively rarely-occuring) pairs of blocks in the trace $T$ will actually be adjacent (respectively, far apart) in the new block configuration. In this case, a better configuration might be $d, c, b, a$ with adjacency matrix:

$$\begin{array}{c|cccc}
 & \text{a} & \text{b} & \text{c} & \text{d} \\
\hline
\text{a} & . & 3 & 2 & \\
\text{b} & & . & 2 & 2 \\
\text{c} & 1 & 1 & . & 2 \\
\text{d} & 3 & & & .
\end{array}$$

The cost is easily calculated to be 23, a nice improvement.

We notice that one advantage of the adjacency matrix is that it allows us to quickly calculate the cost of a particular configuration of blocks. The explicit formula is

$$\text{cost} = \sum_{i=0}^{n-1} \sum_{j=1}^{n} i \cdot A_{j,1+(i+j-1 \mod n)}$$

Now that we can measure the effectiveness of a particular permutation of blocks, let us discuss how to find a configuration that reduces the cost. First, we employ a greedy algorithm that searches for the pair of blocks that occur together most often (say, $x$ and $y$) and places them together in locations 1 and 2. Then, we find which block follows $y$ most often, and we place it in block 3, and so on. This is an extremely fast and efficient method, and in practice this can reduce cost by as much as 25%, depending on the initial conditions. Second, we use the method of simulated annealing, in which we randomly permute pairs of blocks, re-calculate the cost, and decide whether or not to keep the new configuration. If the cost is lower, then we keep the new layout; if the cost is higher, we evaluate $e^{-d/t}$, with $d =$ the difference in cost and $t =$ the current *temperature,* a value which initially is quite large but decreases at each step. If the $e^{-d/t}$ is greater than a random number between 0 and 1, we admit the new, higher-cost configuration, but if not, we retain the original layout. Early in the algorithm, the temperature $t$ is set to be quite *hot,* and so a fair amount of randomness is tolerated; as the temperature is lowered and the algorithm *cools down,* the layout settles on a nice configuration of low cost. This process is repeated – the temperature is again raised, then cooled down, and a configuration of low cost is found.

Together, these methods are an efficient way to find a cost-effective ordering of disk blocks that, we hope, will speed up access time for the user. As an illustration, we ran a simulation with $n = 100$ blocks, and five traces of length 500 each. The traces were mostly random, except that in an attempt to simulate a typical log of disk access activity there was a one-in-three chance that a particular number, $k$, would be followed by $2k + 1 \bmod n$. Thus, the simulation represented about 2500 different visits to the (10000 total) pairs of blocks on the disk, meaning that almost every pair $i, j$ occurs no more than three times (and most pairs happen once or not at all). The cost for the initial disk layout was 121505. Application of the greedy algorithm brought the cost down to 109027, and simulated annealing brought it down further to 90929, for a total savings of about 25%.

Realizing that the above might not be the best model for disk access, we constructed another simulation. Again, we considered $n = 100$ blocks, but this time we randomly selected 200 pairs of blocks, and had each pair appear in our trace (of disk activity) a random number of times, up to 50. Thus, in this simulation we were modelling about 5000 different visits, twice as many as above, but not nearly as broadly dispersed. In this case, our starting cost was 245684, which was brought down to 190414 by the greedy algorithm and then to 103253 by simulated annealing, a savings of almost 60%.

We see that the effectiveness of our procedure depends heavily on the type of data; if the disk activity consists of visiting a large number of disparate blocks, without much repetition, then the procedure outlined above is not particularly good at finding a good configuration. (Indeed, in such a scenario it is hard to imagine how any procedure could do very well.) Fortunately, most disk activity involves repeated visits to the same sequence of blocks, and in this case our algorithm can offer significant savings.

## 1.3 2D Disk Layout

In reality a collection of stacked disks comprise a hard drive, not a 1D array of blocks. Each disk consists of a series of blocks laid-out on concentric tracks on a circular disk similar to Figure 1.3. As a disk spins, the read-head moves back and forth along a fixed radial line. Note that the number of blocks along the outside of the disk is greater than the number of blocks along the inside of the disk.

For computational simplicity we assume that the number of blocks in a given row (or track) is independent of the distance from the center of the disk. Also, rather than having the disk spin, we take the equivalent view that the head is moving on the disk in a single direction. From

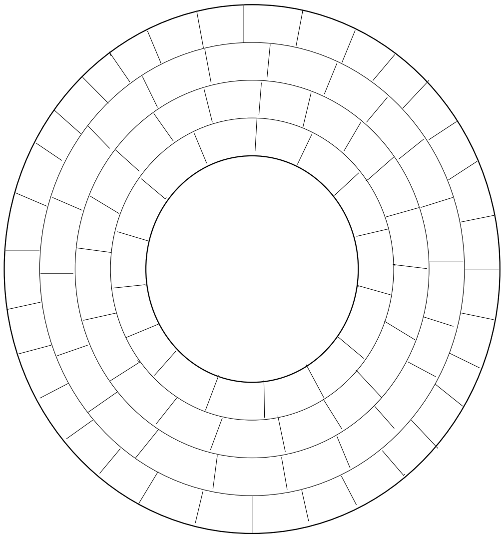


Figure 1.3: A 2D-Layout of blocks on a disk

any given cell the head can move to any adjacent cell in the next column (as in Figure 1.4).

Real disks have about twice as many rows as columns (unlike our diagrams). Finally, we restrict ourselves to considering a single disk rather than a stack of disks.

An immediate observation one can make is that the performance of even a random layout should be greatly improved in the 2D model over the 1D model. Simply put there are more files close together in the 2D model. For example, from a given file in the 2D model there are three files which can be accessed with a cost of 1, whereas in the 1D model there is only one!

A good 1D layout can be transformed into a 2D model simply by “wrapping” the files around the disk, starting in the outside and ending in the inside. See Figure 1.5.

However, given a random 2D layout we can improve on the performance of the disk by applying simulated annealing directly to the 2D geometry. Table 1.1 summarizes our results. The trace we used was extracted from some actual disk logs kindly provided by John DeTreville of Microsoft Research.

We see that our heuristic optimization techniques appear to perform better when applied

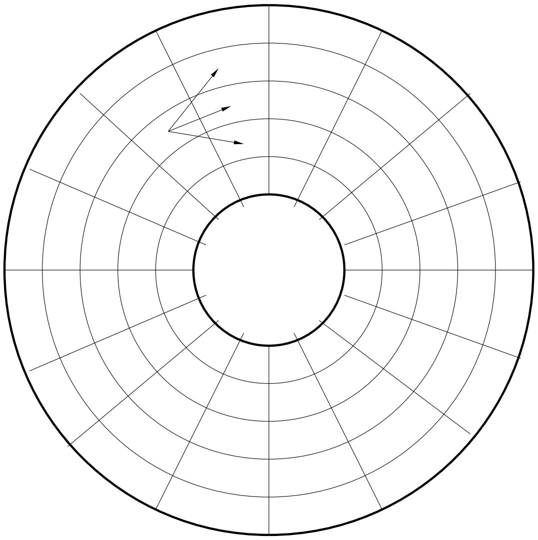


Figure 1.4: Our 2D disk model

directly to the 2D geometry than when applied them to the 1D geometry and then transformed to 2D. This suggests that when performance is critical it is better to optimize the 2D geometry directly. The main problem is, however, that modern hardware only provides access to the 1D geometry of the disk. Our results suggest that disk performance can be improved if 2D (or possibly even 3D) information were available.

## 1.4 Caching Strategies

If the same data records are frequently read from disk, it can be advantageous to keep copies of these records in RAM. This is called the *cache*. One strategy for deciding which records should be in the cache is to retain the $k$ most recently used data records, avoiding the need to reread them. There may be disk layouts that interact particularly well with such a dynamic caching policy.

Our model for the RAM cache is simple. We assume that the cache consists of $k$ block-sized

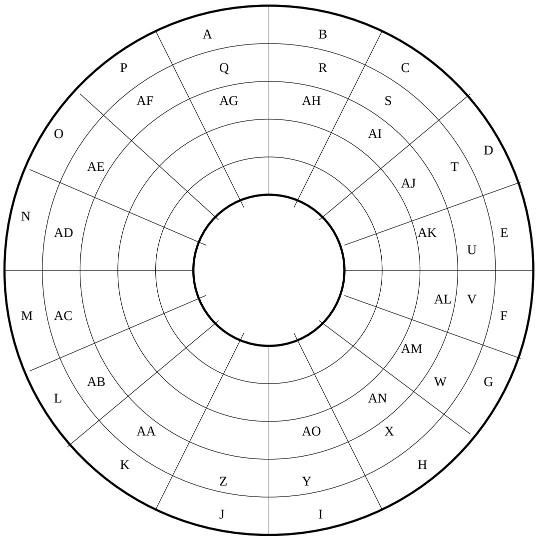


Figure 1.5: The 1D layout $\{A, B, C, ...AM, AN, AO\}$ is transformed into a 2D layout by "wrapping" the files around the disk.

memory chunks in a queue. The difference between cache memory and disk memory is that cache memory is *free* in the sense that it takes no access time to read the files in the cache. Every time a file is requested in our trace, we check to see if it is in the cache queue. If it is, then we consider the file as read and we move on to the next file in the trace. Files read from the disk will be placed in the queue. Of course, since the queue is finite, we have to decide which file to remove from the cache when we add one. There are a number of different strategies for managing the cache. We investigated four:

- A random strategy: a random file in the cache is removed to make room for the new file.
- A FIFO (first-in-first-out) strategy: the file which has been in the queue the longest is deleted to make room for the new file.
- A LRU (least-recently-used) strategy: the file in the queue which was accessed least recently is deleted to make room for the new file (note that this differs from the FIFO

| Optimization | Cost |
|---|---|
| Random Layout | 18314 |
| Best 1D Layout transformed | 16851 |
| Simulated Annealing on 2D model | 10900 |

Table 1.1: The average performance of a given thread under various optimization strategies.

strategy because a file that is accessed in the queue will be moved to the front). This strategy is the industry standard.

- A conditional strategy: the file in the queue which is least likely to be accessed next is removed to make room for the new file. Note that this strategy involves maintaining a probability-transition-matrix to keep track of which files are most likely to be accessed next. This adds significant overhead to the cache management strategy.

| Strategy | Average Cost |
|---|---|
| Random | 165000 |
| FIFO | 148500 |
| LRU | 144000 |
| Conditional | 143000 |

Table 1.2: Results of different Caching Strategies.

The results of applying these four strategies can be seen in Table 1.2. We applied the strategies to 200 random layouts of the 1D disk models. From the table we can see that the conditional strategy was the best; however it was only marginally better than the industry-standard LRU strategy. Given the additional overhead required to apply the conditional strategy, we conclude the LRU strategy is the best of those we considered.

No attempt was made to optimize the disk layout for given caching strategies. Indeed, in the results reported for 1D and 2D disk models we assume that there is a simple cache of size one, the cached file is always discarded when a new file is read.

## 1.5 Added layers of complexity to the model

### 1.5.1 Multiple outstanding requests

In our model so far we have assumed that disk accesses must be performed according to some total ordering. We might relax this to a partial ordering. For example, we might say that at any moment there can be multiple disk accesses outstanding which may be executed in any convenient order. If multiple independent programs on the computer wish to access the disk, the order in which these accesses are executed might not be important, and some orders might perform better than others. Similarly, if we wish to read a file in its entirety, the order in which its data records are read might not matter. A known good dynamic heuristic, for a given disk layout, is to reorder outstanding access requests so that the disk head seldom changes its direction of travel. It might be possible to choose a disk layout that interacts especially well with this heuristic.

It can be useful to guess what future disk reads may occur and to perform the reads before they are requested. For example, if we read the first data record of a file, we might expect that the second record will soon be read. Reading it now can obviously make sense if the disk is otherwise idle, or if the incremental cost of doing so is very small. Again, it may be possible to choose a disk layout that interacts especially well with dynamic read-ahead. Moreover, the same predictive information that is used to establish the disk layout might be used to direct read-ahead.

### 1.5.2 Exact 2D Geometries

Our 2D model assumed that each disk track had the same number of blocks. This is not true and the actual geometry of the disk adds a non-trivial complication to the model. Moreover, since the industry standard is not to report the details of the disk geometry to the operating system, only limited optimization may be possible.

### 1.5.3 Disk and Head Speed

In our model the disk was spinning at a constant rate. Indeed, this is not quite true. Disks stop, speed up, and slow down. The head accelerates and decelerates when it has to scan the surface of the disk. These factors could be substantial.

### 1.5.4 Similarity of Use

The assumption that traces that appear in one disk log are likely to appear again in future, or even the less strict assumption that current disk use is a good indicator of future disk use, is very strong. Before a great deal of effort is invested into disk layout optimization, some investigation of the validity of these assumptions should be made.

## 1.6 Acknowledgements

The authors would like to thank the Pacific Institute for Mathematical Sciences for hosting the workshop that led to this paper, and we also express our gratitude to John DeTreville of Microsoft Research for posing the disk layout problem and for providing us with invaluable assistance throughout the workshop.

# Chapter 2

# Modelling InSb Czochralski Growth

Tark Bouhennache[1], Leslie Fairbairn[2], Ian Frigaard[3], Joe Ho[4], Alex Hodge[5],
Huaxiong Huang[6], Mahtab Kamali[7], Mehdi H. K. Kharrazi[3], Namyong Lee[8], Randy LeVeque[4],
Margaret Liang[3], Shuqing Liang[6], Tatiana Marquez-Lago[2], Allan Majdanac[2],
W. F. Micklethwaite[9], Matthias Mück[10], Tim Myers[11], Ali Rasekh[3], James Rossmanith[4],
Ali Sanaie-Fard[3], John Stockie[12], Rex Westbrook[13], JF Williams[14], Jill Zarestky[15],

Report prepared by C. Sean Bohun[16].

## 2.1 Introduction

The dominant technique for producing large defect free crystals is known as the Czochralski method. Developed in 1916 by Jan Czochralski as a method of producing crystals of rare metals, this method is now used to produce most of the semiconductor wafers in the electronics industry.

The method begins with a crucible loaded with starting material (polycrystalline indium antimonide) and a seed crystal on which the growth of a single crystalline ingot is initiated. Once the starting material is melted to the correct consistency, a seed crystal is lowered on

---

[1] UCLA
[2] Simon Fraser University
[3] University of British Columbia
[4] University of Washington
[5] University of Victoria
[6] York University
[7] Concordia University
[8] Minnesota State University
[9] Firebird Semiconductors
[10] University of Toronto
[11] University of Cape Town
[12] University of New Brunswick
[13] University of Calgary
[14] University of Bath
[15] University of Texas, Austin
[16] Pennsylvania State University

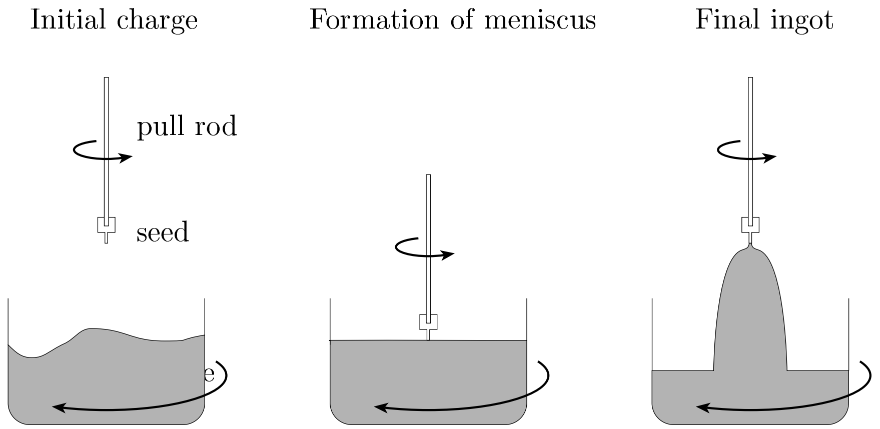


Figure 2.1: The Czochralski crystal pulling technique.

a pull rod until the tip of the seed crystal just penetrates the molten surface. At this point, the seed crystal and the crucible containing the molten starting material are counter-rotated and the temperature is adjusted until a meniscus is supported. As the pull rod is rotated, the seed crystal is slowly withdrawn from the melt developing a single crystal. By carefully controlling the temperatures and rotation rates of the crucible and the rod, a precise diameter of the resulting crystal can be maintained. This process is illustrated in Figure 2.1.

A common problem of using the Czochralski technique is that defects begin to appear in the crystal once the diameter of the crystal exceeds some critical value. The main objective of this study is to attempt to understand this phenomena by modelling the process mathematically. Hopefully, the model can also be used to design growth procedures that produce crystals without defects even when the diameters are greater than the critical values observed under current pull conditions. As indium antimonide (InSb) is used as an infrared detector, being able to manufacture large diameter crystals would have an immediate impact in industry.

The whole growing assembly is maintained in an envelope that permits the control of the ambient gas and enables the crystal to be observed visually. In the case of InSb, the ambient gas is hydrogen to ensure the reduction of any InOx compounds that may be produced. This addition of hydrogen necessitates additional complications to the growth procedure. Namely, i) the high heat losses due to the fluidity of the hydrogen and ii) the avoidance of any oxygen to avoid explosions!

Many aspects of this problem have been investigated to gain a greater insight of the physical processes involved. We begin with the heat problem first as a one dimensional model in Section 2.4 and then extending to a second dimension in Section 2.5. This analysis indicates that the temperature of the gas surrounding the crystal has a major impact on both the thermal stress experienced by the crystal and the shape of the crystal/melt interface. In contrast, variations in the heat flux from the melt have much less of an effect. For completeness the temperature profile of the crucible is also determined in Section 2.7 by neglecting the convection of the liquid InSb.

Having investigated the temperature profiles, the analysis focuses on the behaviour of the fluid in Section 2.8. Scaling arguments are used to estimate the thickness of the various boundary layers and explain the main flow patterns that are experimentally observed.

In Section 2.9 the shape of the meniscus is determined for various rotation rates. The height of the meniscus above the surface of the fluid is about 0.3 mm irrespective of the rotation rate. However, at a rotation rate of 10 rpm, the height of the triple point drops about 0.15 mm from its stationary value. This analysis shows that the shape of the meniscus is relatively invariant at least at low rotation rates yet the actual vertical position of the meniscus changes readily with the rate of rotation.

After analyzing the fluid flow patterns, a model is developed in Section 2.10 for the height of the melt as a function of time. This indicates that for a crystal of constant radius the proportion of the effective pull rate due to the falling fluid level remains essentially constant over the complete growing time of the crystal. This no longer remains true if the radius of the crystal is allowed to increase at a constant rate.

## 2.2 Mathematical Model: Heat Flow

We begin by describing in some detail the mathematical model of the heat flow in the crystal, melt and gas assuming axial symmetry. This model will later be simplified but for now we suppose that the material, in both the solid and liquid states, cools by radiation. In the Czochralski process, the liquid is drawn up, cools to the solidification temperature, and solidifies. As a result the governing equation is

$$\frac{\partial T}{\partial t} + \nabla \cdot (\vec{v}\,T) = \frac{1}{\rho c} \nabla \cdot (k \nabla T) \tag{2.1}$$

where $T$ denotes temperature, $\vec{v}$ velocity, $\rho$ density, $c$ specific heat, and $k$ thermal conductivity. This model assumes that the fluid shear does not dissipate enough energy to heat up the liquid significantly. By fixing the oordinate system to the surface of the liquid, the velocity in the solid phase, $v_p$, is the sum of the crystal pull rate and the rate at which the fluid level drops in the crucible. In the melt, the fluid is assumed to be incompressible and as such the fluid velocity, $v_l$, satisfies $\nabla \cdot \vec{v}_f = 0$.

Let the melt/gas and crystal/gas interfaces be denoted by the surfaces $z = f_l(r,t)$ and $z = f_s(r,t)$ respectively. The normal component of the heat flux must be continuous at these surfaces. Therefore, assuming that the heat is lost through convection and radiation, this gives the boundary condition

$$-k \frac{\partial T}{\partial \mathbf{n}} = h(T - T_g) + \epsilon \sigma (T^4 - T_a^4). \tag{2.2}$$

For this expression $\mathbf{n}$ denotes the outward normal of the interface, $h$ the heat transfer coefficient, $\epsilon$ the emittance, $\sigma$ the Stefan-Boltzmann constant, $T_g$ is the gas temperature, and $T_a$ the ambient temperature.

The crystal/melt interface, $z = S(r,t)$, is a free boundary. At this interface

$$T = T_F \qquad \text{on} \qquad z = S(r,t) \tag{2.3}$$

where $T_F$ is the freezing temperature and

$$\rho_s L \left( \frac{\partial S}{\partial t} - v_p \right) = \left[ -k \frac{\partial T}{\partial \mathbf{n}} \right]_s^l = k_s \left( \frac{\partial T_s}{\partial z} - \frac{\partial T_s}{\partial r}\frac{\partial S}{\partial r} \right) - k_l \left( \frac{\partial T_l}{\partial z} - \frac{\partial T_l}{\partial r}\frac{\partial S}{\partial r} \right). \tag{2.4}$$

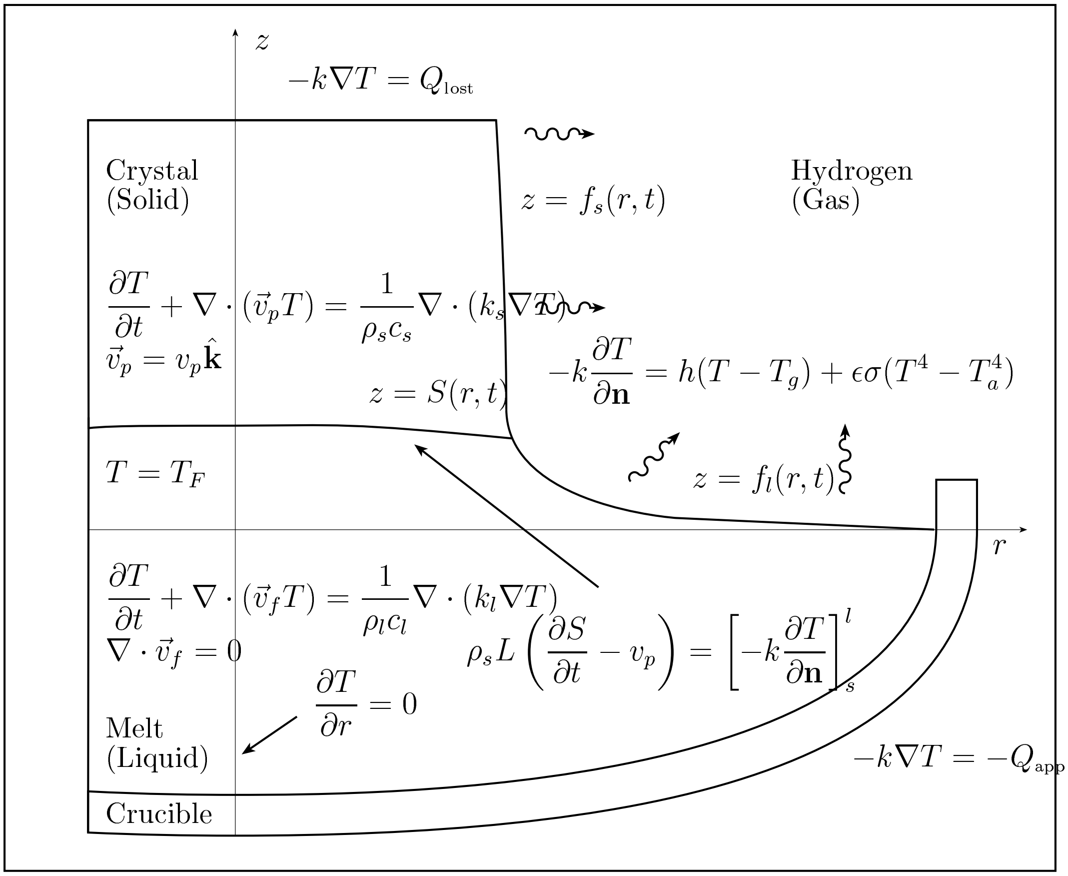


Figure 2.2: Summary of the equations, geometry and boundary conditions. The $z$ direction is greatly exaggerated for clarity in that the interface $z = S(r,t)$ is shown in Section 2.9 to lie very close to the line $z = 0$. See Section 2.7 for an analysis of the heat in the crucible region.

This latter condition equates the heat lost in the phase transition from liquid to solid ($L$ per unit mass) to the net heat flux accumulating at the interface. Since InSb expands on freezing there is either a net flow of InSb away from $z = S$ or the surface of the crystal must rise. Other boundary conditions include a regularity condition at $r = 0$, an applied heat flux of $Q_{\text{app}}$ in the crucible and a heat flux $Q_{\text{lost}}$ lost out the top of the crystal. Figure 2.2 illustrates the geometry and summarizes the equations and boundary conditions in the crystal, melt and crucible. These problems are specifically dealt with in Sections 2.4, 2.5 and 2.7.

## 2.3 Nondimensionalization: Heat Flow

To identify the dimensionless parameters in the heat problem and to determine the relative importance of the various terms we set

$$r^* = r/l_r, \qquad S^* = S/l_r, \qquad z^* = z/l_z, \qquad t^* = t/\tau,$$
$$v_p^* = v_p/v_o, \qquad T^* = \frac{T - T_a}{T_F - T_a}$$

where $l_r$, $l_z$ are the characteristic lengths, $\tau$ and $v_o$ are the time and velocity scales, and $T_F - T_a$ is the representative temperature scale. In terms of these variables equation (2.1) in the crystal becomes

$$\frac{\rho_s c_s l_z^2}{k_s \tau}\left(\frac{\partial T^*}{\partial t^*} + \frac{v_o \tau}{l_z} v_p^* \frac{\partial T^*}{\partial z^*}\right) = \frac{\partial^2 T^*}{\partial {z^*}^2} + \frac{l_z^2}{l_r^2}\frac{1}{r^*}\frac{\partial}{\partial r^*}\left(r^* \frac{\partial T^*}{\partial r^*}\right)$$

while the Stefan equation yields

$$\frac{\rho_s L l_z l_r}{k_s (T_F - T_a)\tau}\left(\frac{\partial S^*}{\partial t^*} - \frac{v_o \tau}{l_r} v_p^*\right) = \left(\frac{\partial T_s^*}{\partial z^*} - \frac{l_z}{l_r}\frac{\partial T_s^*}{\partial r^*}\frac{\partial S^*}{\partial r^*}\right) - \frac{k_l}{k_s}\left(\frac{\partial T_l^*}{\partial z^*} - \frac{l_z}{l_r}\frac{\partial T_l^*}{\partial r^*}\frac{\partial S^*}{\partial r^*}\right).$$

Denoting $\delta = l_r/l_z$, $\tau = l_z/v_o$, $\mathrm{Pe} = v_o l_z \rho_s c_s / k_s$, the Péclét number based on the length in the $z$ direction, and dropping the asterisks results in the expression

$$\mathrm{Pe}\left(\frac{\partial T}{\partial t} + v_p \frac{\partial T}{\partial z}\right) = \frac{\partial^2 T}{\partial z^2} + \frac{1}{\delta^2}\frac{1}{r}\frac{\partial}{\partial r}\left(r\frac{\partial T}{\partial r}\right) \tag{2.5}$$

and the Stefan condition becomes

$$\delta \frac{\partial S}{\partial t} = v_p + \frac{k_s (T_F - T_a)}{\rho_s L v_o l_z}\left[\left(\frac{\partial T_s}{\partial z} - \frac{k_l}{k_s}\frac{\partial T_l}{\partial z}\right) - \frac{1}{\delta}\left(\frac{\partial T_s}{\partial r}\frac{\partial S}{\partial r} - \frac{k_l}{k_s}\frac{\partial T_l}{\partial r}\frac{\partial S}{\partial r}\right)\right]. \tag{2.6}$$

Ignoring the effects of radiation, the boundary conditions at $r = 0$ and $r = 1$ are given by

$$\frac{\partial T}{\partial r}(0, z) = 0, \qquad \frac{\partial T}{\partial r}(1, z) = -\gamma[T - T_g(1)] \tag{2.7}$$

where $\gamma = h l_r / k_s$ from expression (2.2), $T_g(1)$ is the nondimensional gas temperature near the crystal surface, and for simplicity we have neglected the heat loss due to radiation.

As typical growth parameters for InSb we take $\rho_s L = 1.3 \times 10^9$ J m$^{-3}$, $T_F = 798.4$ K, $T_a \simeq 300$ K, $k_l = 9.23$ J m$^{-1}$s$^{-1}$K$^{-1}$, $k_s = 4.57$ J m$^{-1}$s$^{-1}$K$^{-1}$, $\rho_l c_l = 1.7 \times 10^6$ J m$^{-3}$K$^{-1}$, $\rho_s c_s = 1.5 \times 10^6$ J m$^{-3}$K$^{-1}$, $\rho_l = 6.47 \times 10^3$ kg m$^{-3}$, $\rho_s = 5.64 \times 10^3$ kg m$^{-3}$, $l_r = 0.03$ m, $h = 10$ J m$^{-2}$s$^{-1}$K$^{-1}$. With this choice of parameters

$$v_o l_z = 1.75 \times 10^{-6}, \qquad \mathrm{Pe} = 9850 \delta v_o, \qquad \gamma = 6.56 \times 10^{-2}$$

where the first parameter is determined by setting the coefficient in the Stefan equation to one. This condition connects the aspect ratio and the pull rate through $\delta = 1.71 \times 10^4 v_o$. Typical pull rates range from 0.1-100 mm hr$^{-1}$ or about $10^{-8}$-$10^{-5}$ m s$^{-1}$. Consequently $\mathrm{Pe} \leq 0.02$ and the left hand side of (2.5) may be neglected.

For the numerical simulations, the temperature of the gas, $T_g(z)$, was given an exponential behaviour. In non dimensionalized form

$$T_g(z) = T_{\min} + (T_{\max} - T_{\min})e^{-\lambda z}, \qquad \lambda = 0.15, T_{\min} = 0.5, T_{\max} = 0.9. \tag{2.8}$$

A crude estimate for the fluid heat flux $k_l \partial T_l/\partial z \simeq k_l \Delta T_l/\Delta z$ where $\Delta z$ is the width of the fluid boundary layer and $\Delta T = T_{\text{crucible}} - T_{\text{melt}}$. Details on how $\Delta z$ is determined can be found in Section 2.8.2. In the case of InSb this gives $k_l \partial T_l/\partial z \simeq -50k_l \simeq -450$ W m$^{-2}$.

Converting from the non dimensionalized values back into their dimensional versions is straightforward. Taking the non dimensionalized uniform pull rate, $v_p^* = 1$ yields

$$v_o = \frac{k_s(T_F - T_a)}{\rho_s L l_z}, \qquad \frac{\partial S}{\partial t} - v_p = v_o\left(\frac{\partial T*}{\partial z^*} - \frac{k_l}{k_s}\frac{\partial T*}{\partial r^*}\right)$$

and, $T = T_a + (T_F - T_a)T^*$. The fixed uniform pull rate is an artifact of choosing the coefficient in expression (2.6) to be unity and could be changed with the addition of another parameter. Finally, since the system is encapsulated, the ambient temperature is probably much higher than $T_a = 300$ K. Increasing $T_a$ will result in a corresponding drop in the value of $v_p$.

## 2.4 First Steps: A 1D Temperature Model

For any fixed height $z$ the average of the temperature across the crystal radius is given by

$$\overline{T}(z) = 2\int_0^1 T(r,z)\, r\, dr$$

where we have used the non dimensionalized coordinates. Applying this averaging technique to equations (2.3), (2.5) and (2.7) we obtain the second order linear nonhomgeneous boundary value problem

$$\frac{d^2\overline{T}}{dz^2} = -\frac{2\gamma}{\delta^2}[\overline{T} - T_g(z)], \qquad \overline{T}(0) = 1,\ \frac{d\overline{T}}{dz}(1) = -\frac{\gamma}{\delta}[\overline{T}(1) - T_g(1)] \tag{2.9}$$

where $T_g(z)$ is given by (2.8) and $\delta = l_r/l_z = 1/3$. The growth of the crystal/melt interface is governed by the Stefan condition (2.6) and by assuming that the slope of the interface is small, $|\partial S/\partial r| \ll 1$, one obtains

$$\delta\frac{\partial S}{\partial t} = v_p + \frac{\partial T_s}{\partial z} - \frac{k_l}{k_s}\frac{\partial T_l}{\partial z}. \tag{2.10}$$

With this averaging method, $T_s = \overline{T}(0)$ while the value for $k_l \partial T_l/\partial z \simeq -450$ W m$^{-2}$.

Expression (2.9) was solved using a shooting method starting at $z = 1$ and shooting towards $z = 0$. The Robin condition, $d\overline{T}/dz(1) = -(\gamma/\delta)[\overline{T}(1) - T_g(1)]$ precluded starting at $z = 0$. In detail, the temperature $\overline{T}(1)$ was assumed and $d\overline{T}/dz(1)$ is given by the Robin condition. The next choice for $\overline{T}(1)$ depends on the value of $\overline{T}(0)$, the method converging once $\overline{T}(0) = 1$. Solving (2.9) for $\overline{T}(z)$ gives the decreasing temperature profile shown on the left of Figure 2.3. The right side of the illustration is the temperature dependence of the gas, $T_g(z)$. In this case $T_F - T_g(0) = 80$ K in dimensionalized units and the interface velocity from uniform, $v_p = 70$ mm hr$^{-1}$, is $\partial S/\partial t - v_p = -29.6$ mm hr$^{-1}$. Figure 2.4 illustrates the relative velocity as $T_F - T_g(0)$ varies from 80 K to 400 K. As expected, increasing $T_F - T_g(0)$ increases the speed of the interface.

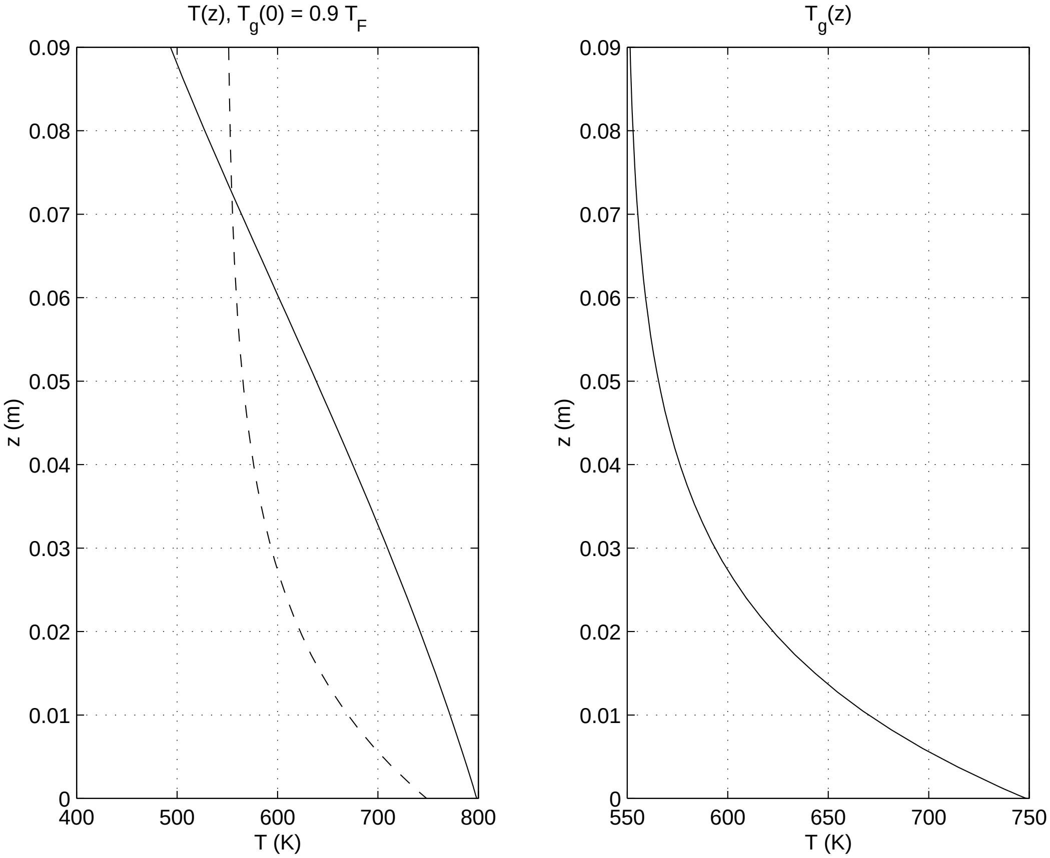


Figure 2.3: The left graph shows the average temperature $\overline{T}(z)$ over the length of the crystal with the temperature of the gas $T_g(z)$ overlaid for comparison. On the right is just the temperature of the gas. The uniform interface velocity is $v_p = 70$ mm hr$^{-1}$ and the deviation from uniform, $\partial S/\partial t - v_p = -29.6$ mm hr$^{-1}$.

## 2.5 2D Temperature Distribution of the Crystal

For the two dimensional problem we return to expression (2.5) and make the standard ansatz

$$T(r,z) = T_0 + \delta T_1 + \delta^2 T_2 + \cdots .$$

This implies that $T_0$ satisfies

$$\frac{1}{r}\frac{\partial}{\partial r}\left(r\frac{\partial T_0}{\partial r}\right) = 0, \qquad \frac{\partial T_0}{\partial r}\bigg|_{r=0} = 0, \qquad \frac{\partial T_0}{\partial r}\bigg|_{r=1} = -\gamma[T_0 - T_g(z)]$$

giving $T_0 = T_g(z)$. Continuing in this fashion we find to $O(\delta^2)$ that

$$T(r,z) = T_g(z) + \delta^2\left(1 - r^2 + \frac{2}{\gamma}\right)\frac{T_g''(z)}{4}. \tag{2.11}$$

A difficulty arises as $z \to 0$ where in the non dimensionalized variables we have the condition $T = 1$. It is unlikely that $T(r,0) = 1 = T_g(0)$ so that a boundary layer correction is required. For

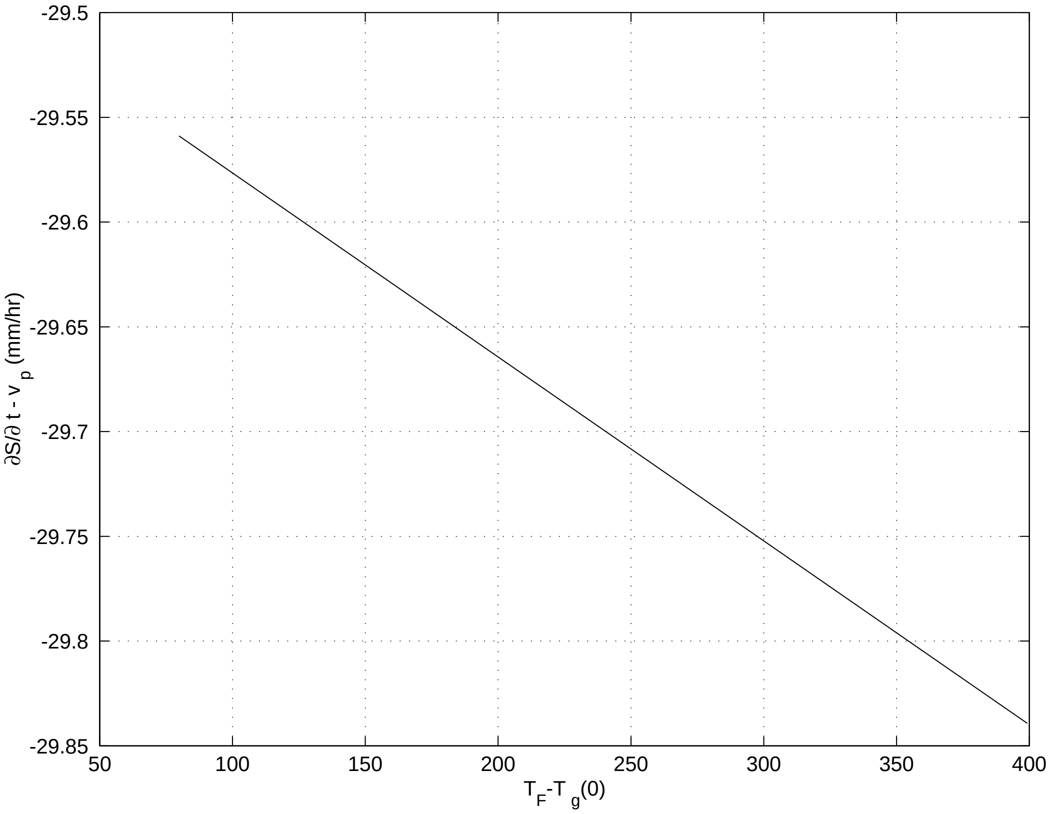


Figure 2.4: The deviation from uniform interface velocity, $\partial S/\partial t - v_p$, as a function of $T_F - T_g(0)$.

the boundary layer solution, $T_{\mathrm{bl}}$, we rescale the $z$ in expression (2.5) by $\delta$ and denote $\tilde{z} = z/\delta$. When the equations are scaled in this way $T_{\mathrm{bl}}$ satisfies

$$\frac{\partial^2 T_{\mathrm{bl}}}{\partial \tilde{z}^2} + \frac{1}{r}\frac{\partial}{\partial r}\left( r \frac{\partial T_{\mathrm{bl}}}{\partial r} \right) = 0 \tag{2.12}$$

with the boundary conditions

$$\frac{\partial T_{\mathrm{bl}}}{\partial r}(0, \tilde{z}) = 0, \qquad \frac{\partial T_{\mathrm{bl}}}{\partial r}(1, \tilde{z}) = -\gamma (T - T_g), \qquad T_{\mathrm{bl}}(r, 0) = 1 - T_g(0), \qquad \lim_{\tilde{z} \to \infty} T_{\mathrm{bl}}(r, \tilde{z}) = 0. \tag{2.13}$$

At $\tilde{z} = 0$ the condition $1 - T_g(0)$ corrects for the $T_g(0)$ from expression (2.11). Solving (2.12)-(2.13) gives to leading order in $\delta$

$$T(r, z) = T_g(z) + T_{\mathrm{bl}}(r, z) = T_g(z) + [1 - T_g(0)] \sum_{n=0}^{\infty} \frac{2\gamma}{\gamma^2 + \zeta_n^2} \frac{J_0(\zeta_n r)}{J_0(\zeta_n)} e^{-\zeta_n z/\delta} \tag{2.14}$$

where $J_0$ is the zeroth order Bessel function of the first kind and the $\zeta_n$ are the zeros of $\zeta_n J_0'(\zeta_n) = -\gamma J_0(\zeta_n)$.

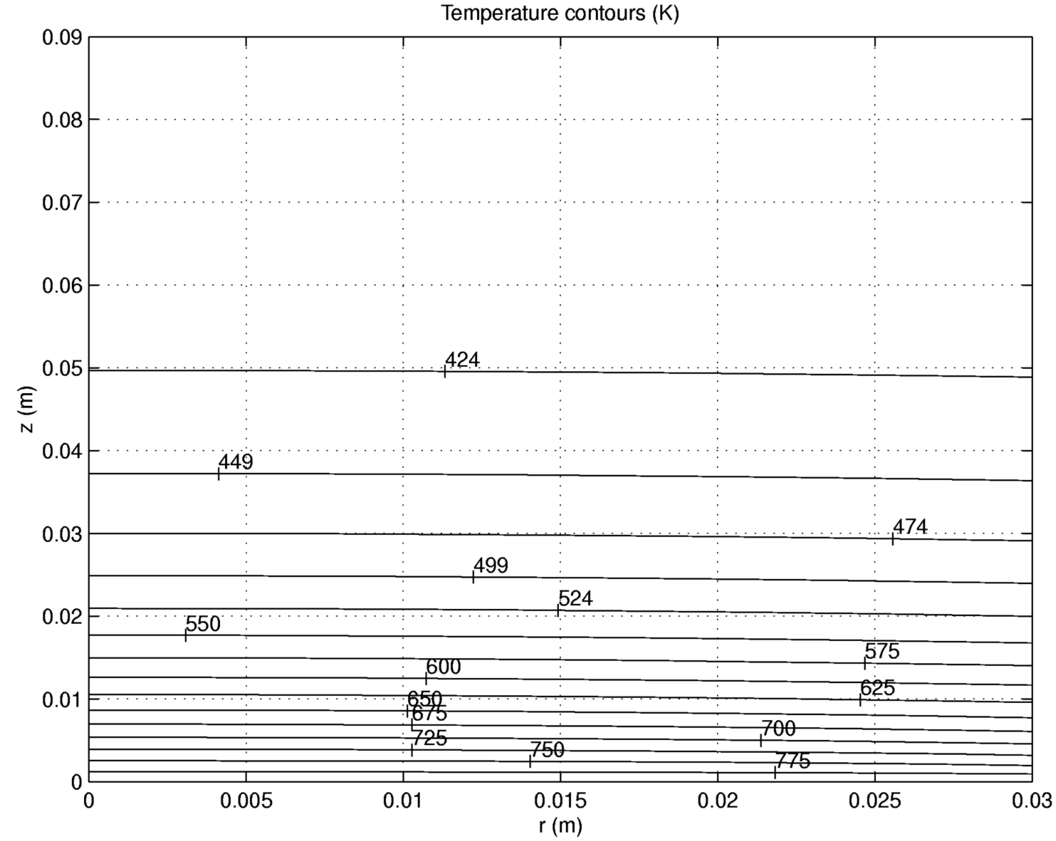


Figure 2.5: Temperature profile $T(r,z)$ in the crystal with $\delta = 1/3$.

As with the one dimensional case, the growth of the crystal/melt interface is governed by the Stefan condition (2.10) where $\partial T_s/\partial z$ now varies with $r$ according to expression (2.14).

For the numerical simulations, $T_g(z)$ was specified by equation (2.8) and $k_l \partial T_l/\partial z$ was varied linearly over the radial coordinate by 15% with an average value of -450 W m$^{-2}$ as in the one dimensional case so that $k_l \partial T_l/\partial z \simeq$ -480 W m$^{-2}$ at $r = 0$ and $k_l \partial T_l/\partial z \simeq$ -420 W m$^{-2}$ at $r = 1$. Choosing $\delta = 1/3$ gives a uniform pull rate of $v_p = 70$ mm hr$^{-1}$. The corresponding two dimensional temperature profile is illustrated in Figure 2.5 and should be compared with Figure 2.3, the profile for the one dimensional case. Since the isotherms in the two dimensional situation are quite flat one would expect considerable agreement with the temperature in the one dimensional case. However, the temperature decreases with $z$ much faster in the two dimensional case. As a result, the speed of the interface, illustrated in Figure 2.6, is about three times that predicted with the one dimensional model. The model accurately predicts that the growth rate is larger near the periphery of the crystal so that the interface is concave down. This asymmetry in the growth rate across the interface increases as $T_F - T_g(0)$ increases. At the other extreme, $T_g(0) > T_F$ the gas melts the crystal and the shape of the crystal/melt interface becomes concave up. Clearly, controlling the temperature of the surrounding gas is critical in reducing the thermal stress within the crystal.

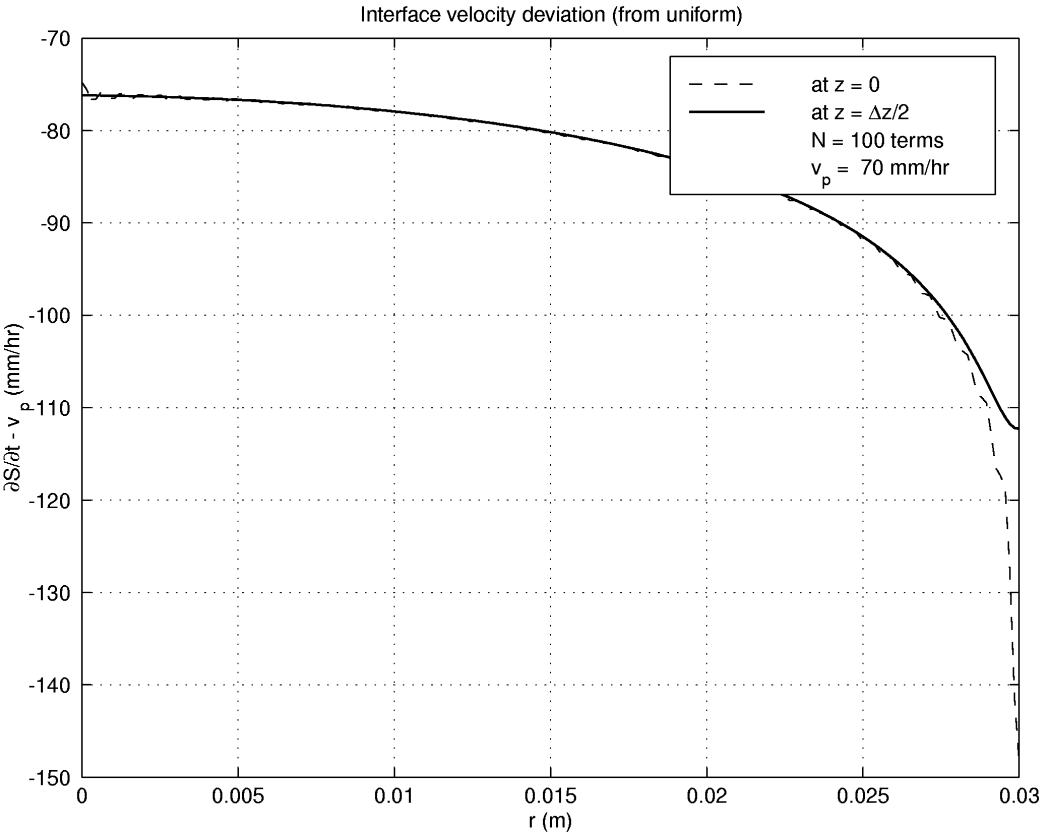


Figure 2.6: Radial dependence of the relative speed of the interface $\partial S/\partial z - v_p$ with $\delta = 1/3$. The dashed curve is the speed at $z = 0$ while the solid curve is the speed just inside the interface at $z = \Delta z/2$. Negative values indicate that the interface is growing downwards. Finally, the $N = 100$ indicates that the Bessel series solution was truncated at 100 terms.

## 2.6 The Thermal Stress Problem

The temperature distribution induces a thermal stress field in the crystal due to the inhomogeneities in the thermal contraction. Some analytical insight as to the source of the stress can be gained by supposing that we have a thin body, $l_r/l_z \ll 1$, and looking at the outer region where the scaling $r/l_r$ and $z/l_z$ is appropriate. The radial and axial displacements $u$ and $w$ are scaled in a similar fashion $u/l_r$ and $w/l_z$. The thermal stresses are scaled by $\alpha T_F E$ where $\alpha$ is the thermal expansion coefficient, $T_F$ is the melting temperature and $E$ the Young's modulus. Under this scaling the strains are $O(1)$.

In terms of scaled variables and using the result $T = T_g(z)$ from Section 2.5 yields

$$\begin{aligned}
\epsilon_r &= T_g(z) + [\sigma_r - \nu(\sigma_\theta + \sigma_z)] = \frac{\partial u}{\partial r} \\
\epsilon_\theta &= T_g(z) + [\sigma_\theta - \nu(\sigma_r + \sigma_z)] = \frac{u}{r} \\
\epsilon_z &= T_g(z) + [\sigma_z - \nu(\sigma_r + \sigma_\theta)] = \frac{\partial w}{\partial z} \\
\epsilon_{rz} &= (1+\nu)\sigma_{rz} = \frac{1}{2}\left(\delta \frac{\partial u}{\partial z} + \frac{1}{\delta}\frac{\partial w}{\partial r}\right).
\end{aligned}$$

with $\nu$ the Poisson ratio. The scaled equilibrium equations are

$$\begin{aligned}
\frac{\partial}{\partial r}\sigma_r + \frac{1}{r}(\sigma_r - \sigma_\theta) + \delta \frac{\partial}{\partial z}\sigma_{rz} = 0 \\
\frac{\partial}{\partial r}\sigma_{rz} + \frac{1}{r}\sigma_{rz} + \delta \frac{\partial}{\partial z}\sigma_z = 0.
\end{aligned}$$

As for boundary conditions, because of the axisymmetry we have $u = 0$ and $\partial w/\partial r = 0$ at $r = 0$ while the boundary at $r = 1$ is unstressed so that $\sigma_r = \sigma_{rz} = 0$ at $r = 1$.

Making the standard ansatz $u = u^0 + \delta u^1 + \cdots$, $w = w^0 + \delta w^1 + \cdots$ and using the expression for $\epsilon_{rz}$ one has

$$2(1+\nu)\sigma_{rz} = \frac{1}{\delta}\frac{\partial w^0}{\partial r} + \frac{\partial w^1}{\partial r} + \delta \frac{\partial u^0}{\partial z} + O(\delta^2).$$

Since $\epsilon_{rz}$ is $O(1)$, $w^0 = W(z)$ and therefore $\sigma^0_{rz} = 0$. In addition, the second equilibrium equation implies that

$$\frac{\partial}{\partial r}(r\sigma^1_{rz}) = -r\frac{\partial}{\partial z}\sigma^0_z$$

and by applying the boundary condition at $r = 1$ we have $\sigma^1_{rz} = 0$ and $\partial\sigma^0_z/\partial z = 0$.

The relationship for $u^0$ comes from the first equilibrium equation which reduces to

$$\frac{\partial^2 u^0}{\partial r^2} + \frac{1}{r}\frac{\partial u^0}{\partial r} - \frac{1}{r^2}u^0 = 0$$

with solution $u^0 = A(z)r$. Thus we obtain

$$\sigma^0_r = \frac{A(z) + \nu W'(z)}{(1+\nu)(1-2\nu)} - \frac{T_g(z)}{(1-2\nu)}.$$

Using the boundary condition at $r = 1$ once again gives $\sigma^0_r = 0$ and hence $A(z) = -\nu W'(z) + (1+\nu)T_g(z)$. In a similar fashion we obtain $\sigma^0_\theta = 0$ and $\sigma^0_z = W'(z) - T_g(z) = C$, a constant. If we consider the exact solution for the whole cylinder when the base of the crystal is stress free and simple equilibrium considerations give

$$\int_0^1 \sigma_z r\, dr = 0$$

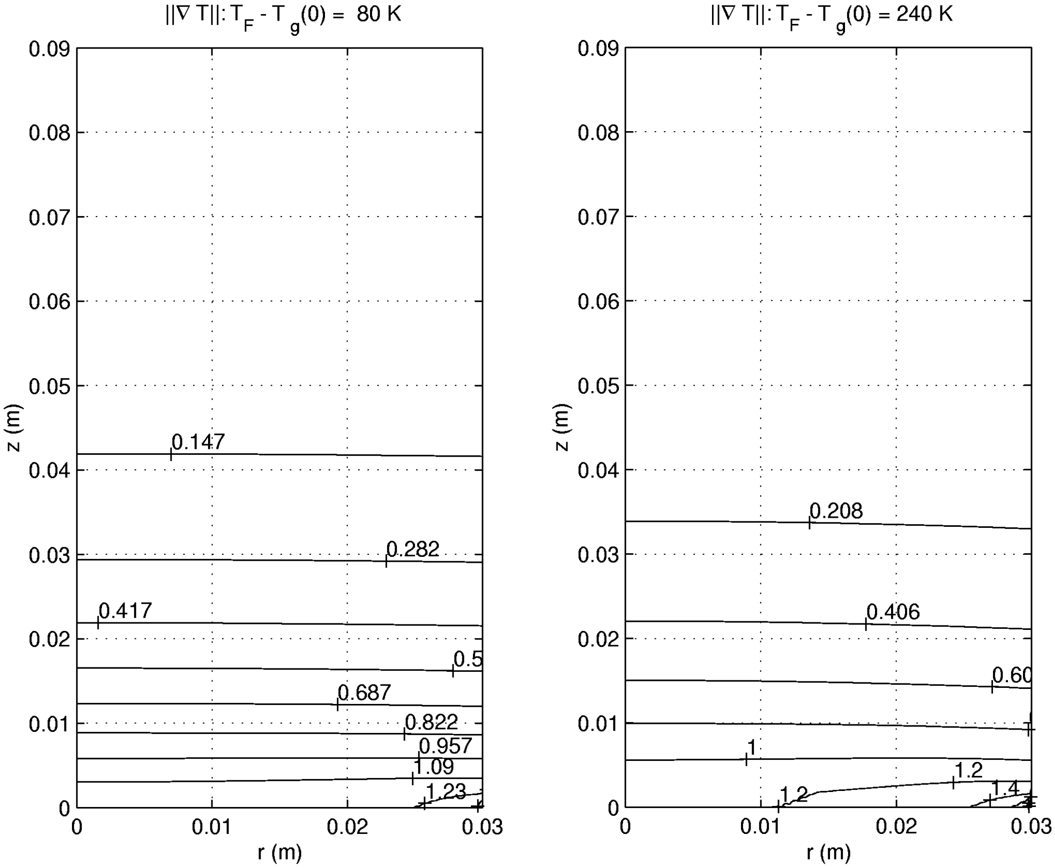


Figure 2.7: Norm of the gradient of the temperature as $T_g(0)$ varies. The figure on the left has $T_g(0) = 720$ K and the figure on the right has $T_g(0) = 560$ K.

at any value of $z$, thus we may conclude that $\sigma_z^0 = 0$ and $W'(z) = T_g(z)$.

Thermal stress will be restricted to a region within a distance $l_r$ from the growing surface. Since these stresses, in the nondimensional case, will depend on the scaled temperature difference $1 - T_g(0)$ we expect them to be of magnitude $\alpha E[T_F - T_g(0)]$ and they will be determined by a solution of the full axisymmetric equations; a problem which appears to be analytically intractable. However it is clear that the magnitude of the stresses can be controlled by making $T_F - T_g(0)$ as small as possible. As numerical evidence of these observations Figure 2.7 displays contours for the norm of the temperature gradient as an indicator of the total stress. Figure 2.8 shows the von Mises stress produced by the temperature distribution obtained in Section 2.5. The von Mises stress is defined as

$$\sigma_e = \left[\frac{(\sigma_1 - \sigma_2)^2 + (\sigma_1 - \sigma_3)^2 + (\sigma_2 - \sigma_3)^2}{2}\right]^{1/2}$$

where $\sigma_1$, $\sigma_2$ and $\sigma_3$ are the principle stresses at a given point within the crystal.

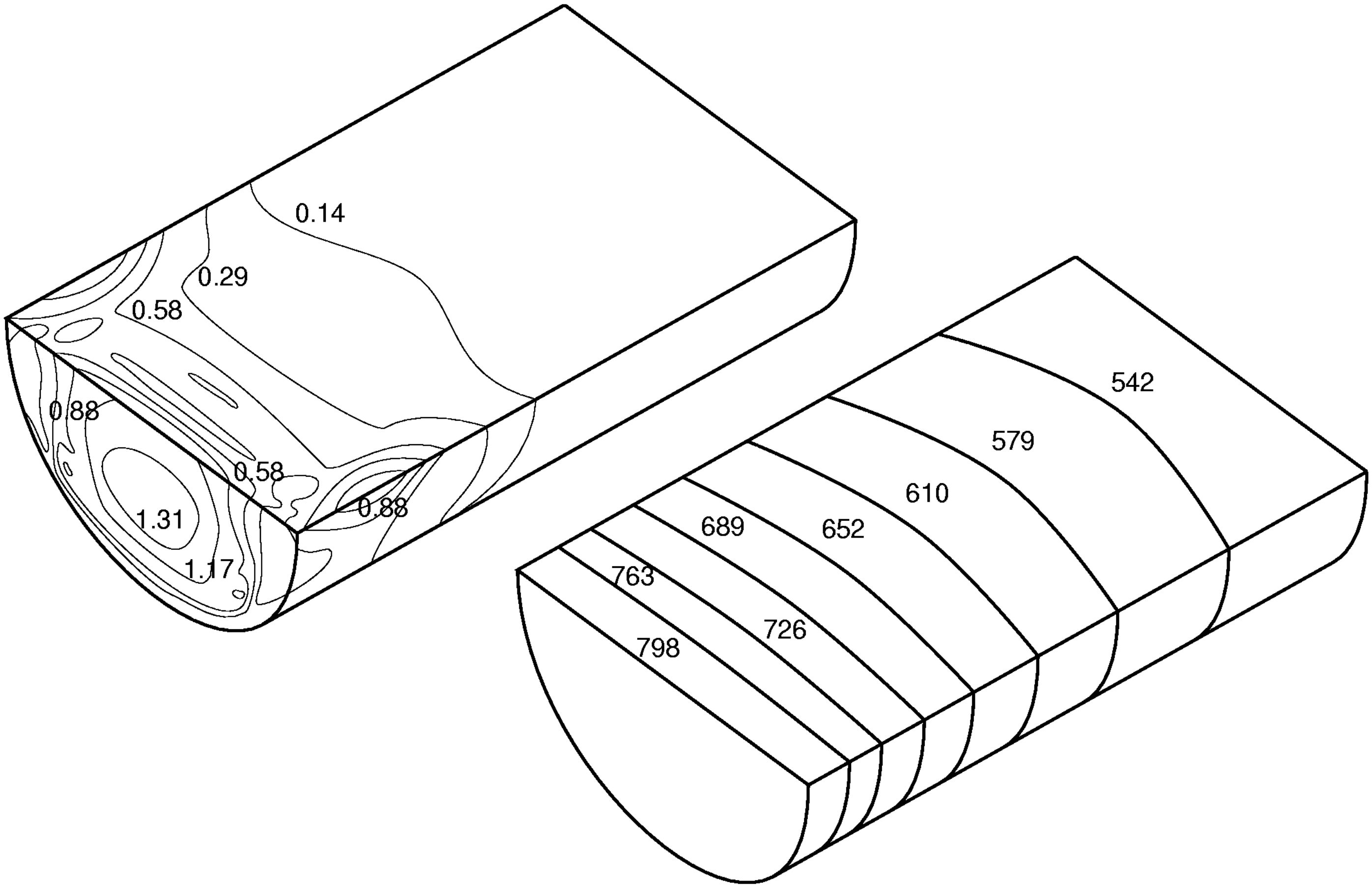


Figure 2.8: von Mises stress of an InSb crystal together with the corresponding temperature distribution.

## 2.7 Distribution of Heat in the Crucible

For completeness we now determine the temperature profile in the crucible and the holder assuming no motion of the fluid. The isotherms will be modified by any convective flow in the crucible but as we will see in Section 2.8 this flow is practically inviscid so that the temperature will for the most part remain stratified. Figure 2.9 illustrates the domain and summarizes the boundary conditions. For the interior region we have liquid InSb with a thermal conductivity of $k_l = 9.23$ W m$^{-1}$K$^{-1}$. Outside of this is a thin layer of quartz, 3 mm, with a conductivity of approximately $k_q = 1.5$ W m$^{-1}$K$^{-1}$ and finally surrounded by a layer of graphite with $k_g = 120$ W m$^{-1}$K$^{-1}$. It should be noted that for simplicity we have taken the thermal conductivity of each of these materials to be constant however they are actually functions of the temperature. For example, $k_g$ varies from 150 W m$^{-1}$K$^{-1}$ to 100 W m$^{-1}$K$^{-1}$ as the temperature increases from 300 K to 900 K. This problem is complicated by the involved boundary conditions. There is a regularity condition at $r = 0$ and a heat inflow at $r = 0.1$ m with an applied heat flux of about $Q_{\text{app}} = 1200$ W. At $z = -0.16$ m there is heat lost due to convection with a heat transfer coefficient $h = 10$ W m$^{-2}$K$^{-1}$ to the surrounding hydrogen gas at a temperature $T_{g1} = 600$ K. At the top of the melt, $z = 0$, there are two conditions. At the crystal/melt interface the

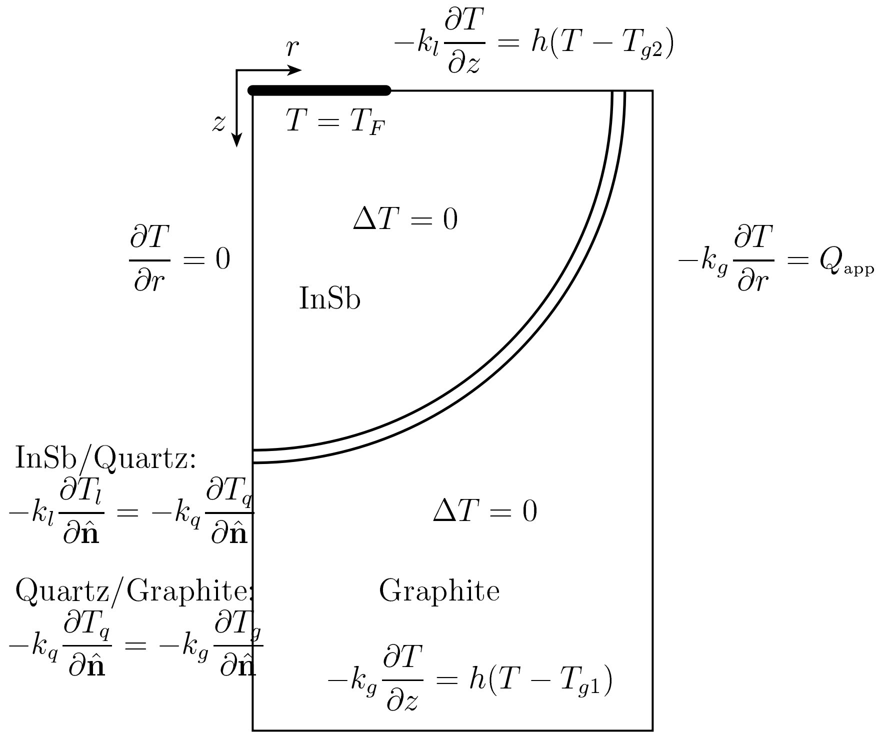


Figure 2.9: Shown here is the geometry and boundary conditions for solving the steady state heat equation in the crucible and the holder. Summarizing the parameters: $k_l = 9.23$ W m$^{-1}$K$^{-1}$, $k_q = 1.5$ W m$^{-1}$K$^{-1}$, $k_g = 120$ W

temperature of the melt is the solidification temperature of the crystal. Therefore, $T = T_F = 798.4$ K for $z = 0$ and $0 \leq r \leq l_r$ with $l_r = 0.03$ m. The remainder of this boundary suffers heat loss due to convection again with a heat transfer coefficient of $h = 10$ W m$^{-2}$K$^{-1}$ but in this case the surrounding gas is taken to have a temperature of about $T_{g2} = 700$ K. Two final conditions are that the temperature flux must be continuous at the graphite/quartz and the quartz/InSb boundaries. Figure 2.10 shows the isotherms and the interesting artifact of a cold spot at the bottom of the holder at $r = 0$.

## 2.8 Mathematical Model: Fluid Flow

We now turn our attention to the behaviour of the fluid. The fundamental equations of the fluid motion are governed by the incompressible Navier-Stokes equations within a rotating crucible. We assume that the flow is independent of the azimuthal angle and that the variations in the fluid density can be ignored except insofar as their effect on the gravitation forces. This latter assumption is known as the Boussinesq approximation.

Consider for a moment the force on the fluid due to gravity

$$\vec{F}_g = \rho_l \vec{g} = -\rho \nabla \phi$$

where $\phi = gz$ is the gravitational potential and $\rho_l$ is the density of the fluid. By expressing the

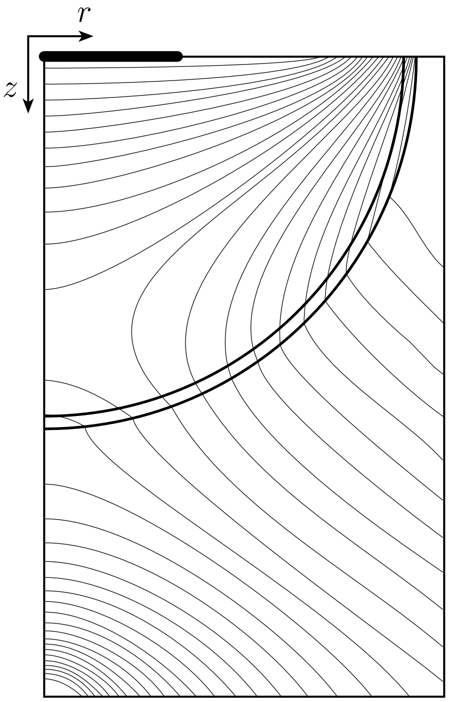


Figure 2.10: Illustrated is the temperature profile of the crucible and the holder. Note the cold spot at the base of the holder at $r = 0$. This pattern is expected to persist in the presence of the convective flow of the melt since in Section 2.8 it is shown that the fluid flow is essentially inviscid.

density as a constant $\rho_o$ and a small variation $\rho_\epsilon$ we have $\rho_l = \rho_o + \rho_\epsilon$ with $\nabla \rho_o = 0$ and

$$\vec{F}_g = -\nabla(\rho_o \phi) + \rho_\epsilon \vec{g}.$$

Redefining the pressure as $P' = P + \rho_o \phi$ gives the expression

$$-\nabla P + \vec{F}_g = -\nabla P' + \rho_\epsilon \vec{g}. \tag{2.15}$$

Since the change in density, $\rho_\epsilon$, is for the most part a result of heating the fluid, we linearize this change in density so that $\rho_\epsilon \simeq \beta(T - T_F)$ where $\beta$ is the thermal coefficient of expansion.

The fact that the crucible is rotating introduces a coriolis force and a reaction force due to the centripetal acceleration of the fluid particles. This second force can be written as a potential and combined with the nonrotating gravitational potential to give

$$\phi = gz - \frac{1}{2}\omega_1^2 r^2 \tag{2.16}$$

where $-\nabla\phi$ is the measured gravitational force in the accelerated frame and we have taken the rotation rate $\vec{\omega} = -\omega_1 \hat{\mathbf{k}}$.

Combining (2.15), (2.16) and the azimuthal symmetry of the flow yields the following pseudo-

| Data | Symbol | Value |
|---|---|---|
| Growing Properties | | |
| Crystal Radius | $l_r$ | 0.03 m |
| Crucible Radius | $R_c$ | 0.08 m |
| Liquid Properties | | |
| Melting Temperature | $T_F$ | 798.4 K |
| Density | $\rho_l$ | $6.47 \times 10^4$ kg m$^{-3}$ |
| Thermal Conductivity | $k_l$ | 9.23 W m$^{-1}$K$^{-1}$ |
| Heat Capacity | $\rho_l c_l$ | $1.7 \times 10^6$ J m$^{-3}$K$^{-1}$ |
| Thermal Diffusivity | $\alpha$ | $5.4 \times 10^{-6}$ m$^2$s$^{-1}$ |
| Dynamic Viscosity | $\nu$ | $3.3 \times 10^{-7}$ m$^2$s$^{-1}$ |
| Coefficient of Expansion | $\beta$ | $1 \times 10^{-4}$ K$^{-1}$ |

Table 2.1: A summary of the physical parameters of liquid InSb.

steady incompressible Navier-Stokes equations for the fluid velocity $\vec{v}_l = \langle u_r, u_\theta, u_z \rangle$

$$\begin{aligned} u_r \frac{\partial u_r}{\partial r} + u_z \frac{\partial u_r}{\partial z} &= -\frac{1}{\rho_o}\frac{\partial P'}{\partial r} - 2\omega_1 u_\theta + \nu \Delta u_r && (2.17) \\ u_r \frac{\partial u_\theta}{\partial r} + u_z \frac{\partial u_\theta}{\partial z} &= 2\omega_1 u_r + \nu \Delta u_\theta && (2.18) \\ u_r \frac{\partial u_z}{\partial r} + u_z \frac{\partial u_z}{\partial z} &= -\frac{1}{\rho_o}\frac{\partial P'}{\partial z} + \nu \Delta u_z - \beta g (T - T_F). && (2.19) \end{aligned}$$

Although it does not appear in these expressions, the angular velocity of the crystal is taken to be $\omega_2 \hat{\mathbf{k}}$ which is in the opposite direction to that of the crucible. In addition to these three equations, the fluid is incompressible and the temperature satisfies expression (2.1). Thus in component form we have

$$\begin{aligned} \frac{1}{r}\frac{\partial}{\partial r}(r u_r) + \frac{\partial}{\partial z} u_z &= 0 && (2.20) \\ u_r \frac{\partial T}{\partial r} + u_z \frac{\partial T}{\partial z} &= \frac{k_l}{\rho_o c_l} \Delta T. && (2.21) \end{aligned}$$

Even without specifying any boundary conditions, the complexity of these five expressions precluded any detailed simulation of the flow. However, it is known by observing the melt that there exist three distinct regions of flow as depicted in Figure 2.11. Cell I is a buoyancy driven cell from expression (2.19). Cell II results from Ekman pumping and is a consequence of (2.17) and (2.18). Cell III is a complex spiral that is expected to exist at higher rotation rates.

Over the next couple of subsections each of these regions is analysed using the material parameters of the liquid InSb and in preparation for this, these parameters are collected in Table 2.1.

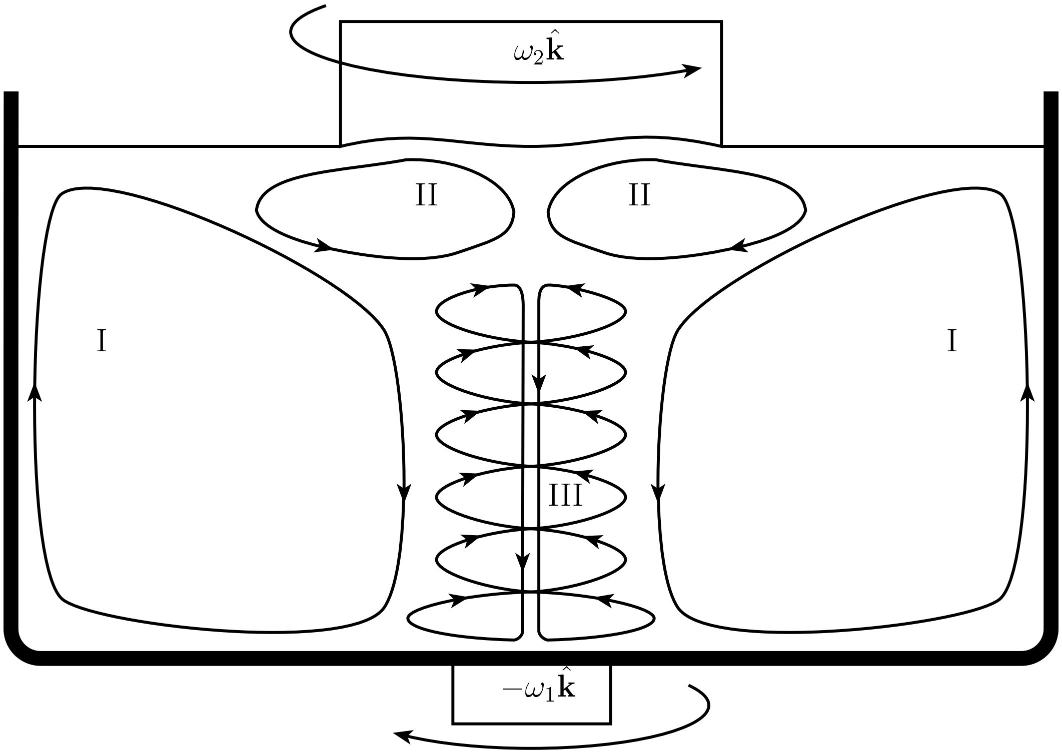


Figure 2.11: Experimentally observed flow pattern of the liquid InSb. The three major features are I: a buoyancy drive cell; II: a cell driven by Ekman pumping; III: a transient spiral.

### 2.8.1 Cell I

This cell is a buoyancy driven cell resulting from the upwelling of heated InSb at the outside wall of the crucible and the subsequent radial inflow as the fluid cools. By comparing the relative strengths of the inertial, buoyancy and viscosity forces on a packet of fluid the width and flow rate of this viscous boundary layer can be estimated. Let the viscous boundary layer have thickness $\delta_I$ and an upward velocity of $u_I$ at the crucible wall. The subscript refers to the cell under consideration. For the length scale, we choose the height of the crucible which is approximately $R_c$. Balancing the three forces yields the expression

$$\frac{u_I^2}{R_c} \simeq \beta g(T - T_F) \simeq \frac{\nu u_I}{\delta_I^2}$$

and a little rearranging gives

$$\mathrm{Re}_I = \frac{u_I R_c}{\nu} = \mathrm{Gr}_I^{1/2}, \qquad \delta_I = \mathrm{Gr}_I^{-1/4} R_c$$

where $\mathrm{Re}_I$ is the Reynolds number and $\mathrm{Gr}_I = \beta g(T - T_F)R_c^3/\nu^2$ is the Grashof number. As with liquid metals, the Prandtl number $\mathrm{Pr}_I = \nu/\alpha \simeq 0.061 \ll 1$ which implies that there is a very thin viscous boundary as compared to the thermal boundary layer so that the heat flow is driven by the thermal diffusivity.

To determine whether or not there is a convective flow we compute the Rayleigh number, $\mathrm{Ra} = \mathrm{Gr}\mathrm{Pr}$. If Ra exceeds a critical value (about 1100 for a free surface) then a convective flow is expected. In our case $T - T_F \simeq 30$ K so that $\mathrm{Ra}_I \simeq 2.8 \times 10^4$ and indeed we predict that there will be a buoyancy cell. This buoyancy cell is practically unavoidable in that one requires $T - T_F < 10^{-3}K$ to prevent it. Having established that there is a convective flow, the speed of the upwelling InSb is given by the relationship $v_{o,I}\delta_I \simeq \alpha$ or $v_{o,I} \simeq \alpha \mathrm{Gr}^{1/4}/R_c$. The flow rate around the cell is $Q_I = 2\pi R_c \delta_I v_{o,I} = 2\pi\alpha R_c$. Finally, in the core region the speed of the falling fluid satisfies $\pi l_r^2 v_{i,I} = 2\pi\alpha R_c$ which implies that $v_{i,I} = 2\alpha R_c/l_r^2$. Setting $T - T_F \simeq 30$ K gives $\mathrm{Gr}_I = 1.4 \times 10^8$, $\mathrm{Re}_I = 1.2 \times 10^4$, $\delta_I = 0.7$ mm, $v_{o,I} = 7.4$ mm s$^{-1}$, $v_{i,I} = 0.97$ mm s$^{-1}$ and $Q_I = 2.7$ ml s$^{-1}$.

### 2.8.2 Cells II and III

The steady velocity of the rotating crystal at $z = 0$ produces a thin boundary layer at the surface. By assuming a horizontal flow at the surface, expressions (2.18) and (2.19) reduce to

$$-2\omega u_\theta + \nu\frac{\partial^2 u_r}{\partial z^2} = 0$$
$$2\omega u_r + \nu\frac{\partial^2 u_\theta}{\partial z^2} = 0$$

where $\omega = |\omega_1 - \omega_2|$ by taking into account the combined rotation of the crystal and the crucible. Letting $\vec{v}_l(z = 0) = \langle 0, v_o, 0\rangle$ and choosing $\lim_{z\to-\infty} \vec{v}_l(z) = 0$ in the geometry of Figure 2.2 we have the solution

$$\vec{v}_l(z) = v_o e^{z/\delta_{II}} \langle \sin(z/\delta_{II}), \cos(z/\delta_{II}), 0\rangle.$$

The thickness of the boundary layer $\delta_{II} = \pi(\nu/|\omega_1 - \omega_2|)^{1/2}$ and is chosen to be the depth at which the velocity is opposite to that at the surface. This $\delta_{II}$ width is used to estimate the fluid heat flux back in Section 2.3. Because the fluid does not rotate as a rigid body with respect to the crystal, we approximate the radial velocity of the fluid to be a fixed proportion of its rigid value so that $v \simeq \gamma r|\omega_1 - \omega_2|$ with $\gamma \simeq 0.05$. To obtain the velocity entering the Ekman layer we take $v$ to be the radial speed of the fluid at a radius of twice the depth of the Ekman layer so that $r \simeq 2\delta_{II}$. This gives $v_{o,II} \simeq 2\pi\gamma(\nu|\omega_1 - \omega_2|)^{1/2}$. By the structure of the Ekman layer, the core velocity, $v_{i,II}$ at $z = -\delta_{II}$ is the same as $v_{o,II}$ except in the opposite direction. As for the flux, this is simply $Q_{II} = \pi l_r^2 v_{o,II} \simeq 2\pi^2\gamma l_r^2(\nu|\omega_1 - \omega_2|)^{1/2}$. For the typical rotation rates, 1-10 rpm, one finds that $v_{i,II} = v_{o,II} \simeq 0.2$ mm s$^{-1}$ and $Q_{II} = 0.65$ ml s$^{-1}$.

This leaves the transient spiral structure. It is expected that this is a result of the fluid entering the Ekman layer with a velocity that far exceeds the speed at the core region of the buoyancy driven cell. Comparing these two velocities gives the expression

$$\frac{\alpha^2}{\pi^2\gamma^2\nu} \ll |\omega_1 - \omega_2|\frac{l_r^4}{R_c^2}$$

which indicates that this structure should appear at large rates of rotation. For the values indicated in Table 2.1 one would require $|\omega_1 - \omega_2| \gg 28$ Hz.

## 2.9 Shape of the Meniscus

The shape of the melt/gas interface, $f_l(r,t)$, is determined by the Laplace-Young equation which describes the equilibrium configuration of a curved liquid surface under the effect of a gravitational field. For cylindrical growth of a crystal the radius of the crystal, $l_r$, changes according to the expression

$$\frac{dl_r}{dt} = \left(v_p - \frac{dh_r}{dt}\right)\tan(\theta - \theta_o) \tag{2.22}$$

where $\theta_o$ is the equilibrium contact angle of the surface with the vertical tangent at the triple point, $\theta$ is the current contact angle, $v_p$ is the pull rate and $dh_r/dt$ is rate of change of the crystal height at the outer edge of the crystal. Since the crucible is rotating, the shape of the meniscus and therefore the height of the triple point above the surface $z = 0$ will be affected by this rotation.

Suppose that the fluid velocity is zero so that there are no coriolis effects and the steady state pressure satisfies

$$-\frac{1}{\rho}\nabla P = \nabla\left(gz - \frac{1}{2}\omega^2 r^2\right)$$

where $\rho \simeq \rho_l$ is difference in density between the liquid and gas phases and where we have taken a rotation rate of $\omega = \omega\hat{\mathbf{k}}$. In addition, the pressure drop across the melt surface, $z = f_l$, is determined by the surface tension, $\sigma_l$ by

$$P = P_o - \sigma\kappa = P_o - \sigma_l\nabla\cdot\left[\frac{\nabla f_l}{(1 + |\nabla f_l|^2)^{1/2}}\right]$$

where $\kappa$ is the curvature of the free surface.

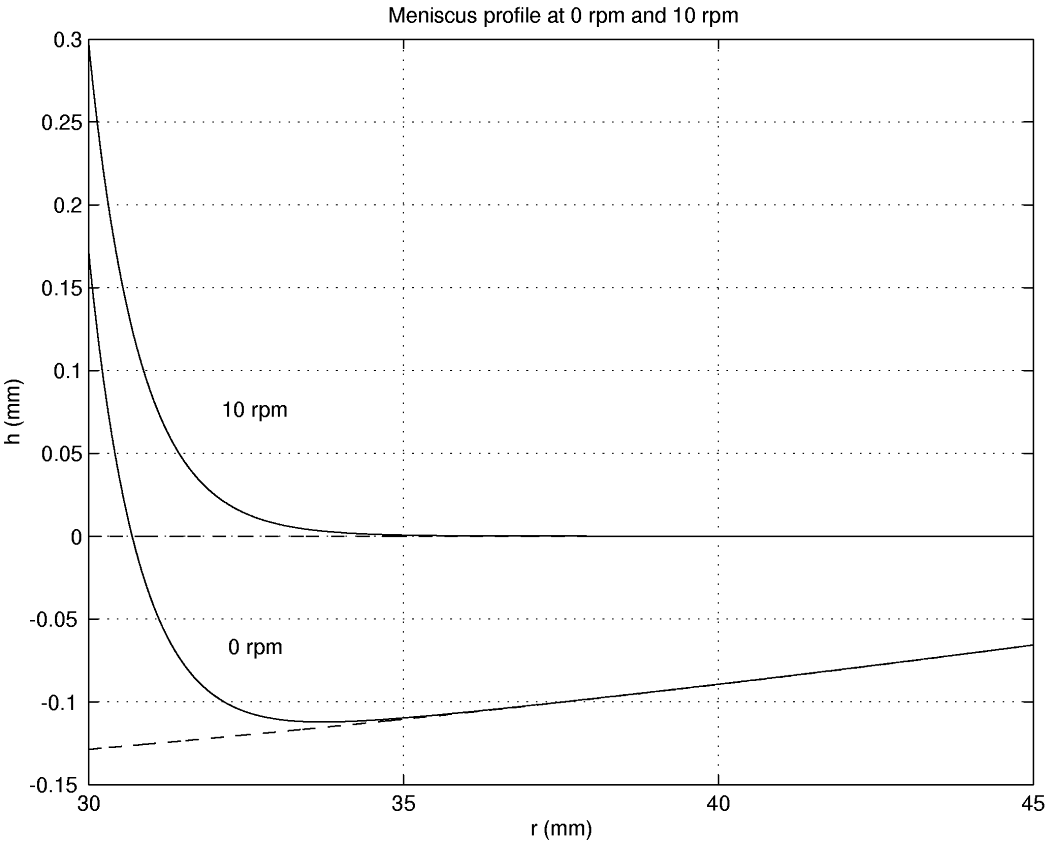


Figure 2.12: Meniscus profile for the melt/gas interface, $f_l(r,t)$ for no rotation and at 10 rpm.

Setting $f_l = h(r)$, combining these two expressions, and denoting derivatives with respect to $r$ with dots one obtains

$$-\frac{1}{\rho}\nabla P = g\dot{h} - \omega^2 r - \frac{\sigma}{\rho}\frac{d}{dr}\left[\frac{1}{r}\frac{d}{dr}\left(\frac{r\dot{h}}{(1+|\dot{h}|^2)^{1/2}}\right)\right] = 0.$$

Letting $r = ar^*$, $h = ah^*$ with $a^2 = \sigma_l/\rho_l g$ and then dropping the stars gives the nonlinear second order ODE

$$\ddot{h} + \frac{\dot{h}}{r}(1+\dot{h}^2) - \left[h - \frac{a\omega^2}{4g}(2r^2 - R_c^2)\right] = 0, \qquad l_r/a \leq r \leq R_c/a$$

where $\dot{h}(l_r/a) = -\cot(\theta_o)$ and $h(l_r/a)$ is chosen so that

$$\lim_{r\to\infty}\left[h(r) - \frac{a\omega^2}{4g}(2r^2 - R_c^2)\right] = 0$$

and at large radii $h(r)$ approaches the parabolic surface due to the rotation of the crucible.

For InSb, $\sigma_l = 0.434$ J m$^{-2}$, $\rho_l = 6.47 \times 10^4$ kg m$^{-3}$ and $\theta_o = 69^o$. Figure 2.12 illustrates the meniscus profile for two cases: no rotation and for a rotation rate of 10 rpm. In both of these cases the crystal radius, $l_r = 3$ cm and $R_c = 8$ cm. Increasing the rotation rate drops the height of the triple point.

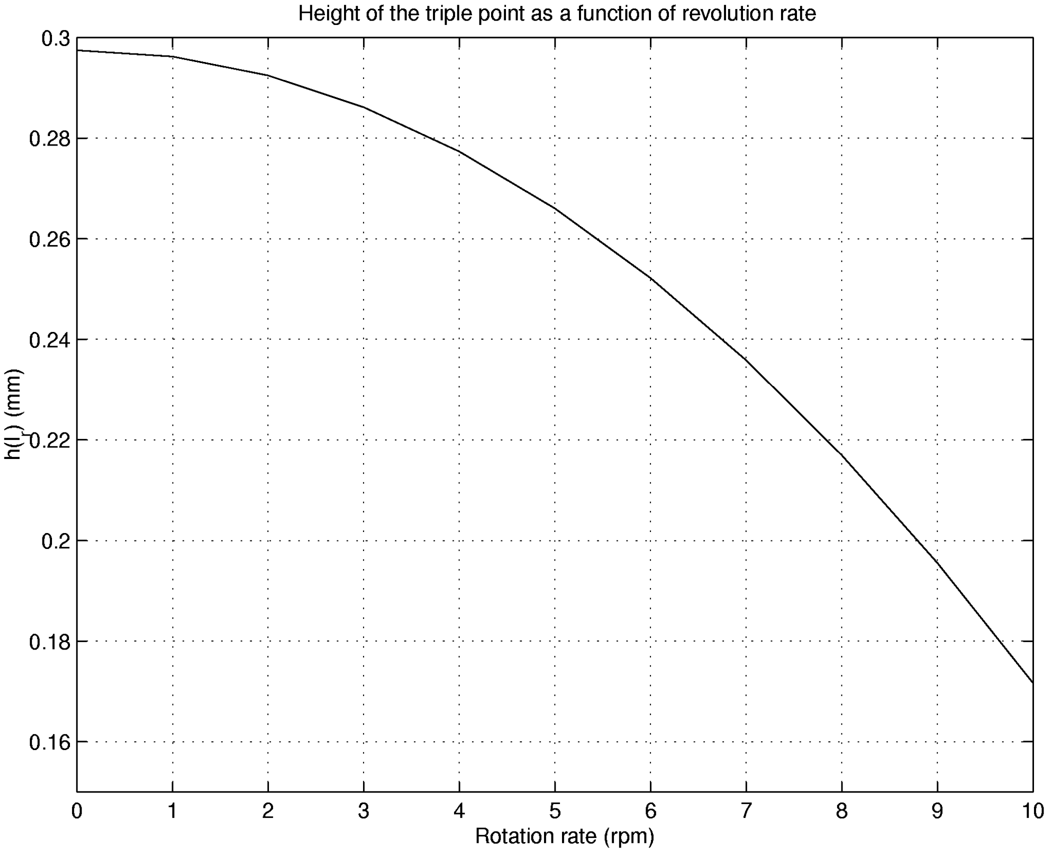


Figure 2.13: Position of the triple point as a function of the rotation rate.

## 2.10 A Model for the Melt Height

Up to this point we have taken the coordinate system to be fixed at the crystal/melt interface so that the pulling speed $v_p$ is the sum of the crystal pull rate and the rate at which the fluid level drops in the crucible. In this section we will determine the proportion of effective pulling rate that is due to the dropping level of the fluid.

At any time $t$ the mass of the fluid that leaves the crucible must equal the mass that is incorporated into the crystal. That is,

$$\rho_l \frac{\partial V_l}{\partial t} = 2\pi \rho_s \int_0^{R(t)} \left( \frac{\partial S}{\partial t} - v_p \right) r\, dr \tag{2.23}$$

where $R(t)$ is the radius of the crystal at time $t$ and $S$ is the location of the crystal/melt interface. For $V_l$ we assume that the crucible is a hemisphere of radius $R_c$ so that

$$V_l = \pi \left( \frac{2}{3} R_c^3 + S R_c^2 - \frac{1}{3} S^3 \right) \tag{2.24}$$

where $-R_c \leq S(t) \leq 0$. By assuming that the interface is essentially flat $\partial S/\partial r \simeq 0$, expressions

(2.23) and (2.24) combine to give

$$\frac{\partial S}{\partial t} = \frac{v_p}{1 - \dfrac{\rho_l}{\rho_s}\dfrac{(R_c^2 - S^2)}{R^2(t)}} \tag{2.25}$$

with $S(0) = 0$ if one starts with an initially full crucible. Expression (2.25) provides an exact solution for the height of the melt surface and can be used to accurately determine the appropriate rate at which to move the crucible. Based on the geometry $t \leq t_c$ where $t_c$ is the time at which the crystal comes in contact with the crucible, $S^2(t_c) + R^2(t_c) = R_c^2$. Consequently, the slope in expression (2.25), $\partial S/\partial t \geq v_p/(1 - \rho_l/\rho_s)$.

When the crystal radius is constant, (2.25) can be integrated to give a cubic equation for $S$ but in general we take $R(t) = l_r + v_p t \tan\varphi$ where $2\varphi \simeq 4^o$ is the growth angle. Figure 2.14 illustrates the height of the surface and the proportion of the effective pull rate due to the falling liquid state for $\varphi = 0$ (constant radius) and $\varphi = 8^o$. For the constant radius case the rate at which the fluid falls is essentially constant until the height of the fluid reaches about $-0.75R_c$. Over this region about 20% of the effective pull rate is due to the falling fluid. As the level drops further, the rate of the falling fluid becomes the dominant effect. When $\varphi = 2^o$ the growing time is reduced since the crystal reaches the sides of the crucible much earlier. However the same behaviour is observed except that the fluid accounts for about 30% of the effective pull rate and this linear behaviour extends for a shorter time period.

## 2.11 Conclusion

The main purpose of this work was to understand the growing process of InSb with the ultimate hope of growing large radius crystals. Analysing the temperature distribution within the crystal allowed us to estimate the growing rate by solving the Stefan problem. However, this relied on a very crude estimate for the heat flux from the melt. Despite this drawback, it was noticed that the growth rate of the crystal/melt interface is larger at the periphery of the crystal and that the temperature gradients are largest near the triple point. The temperature distribution was used to calculate the von Mises stress. Calculation of the stress is essentially a post processing analysis but could in principle be incorporated into a feedback control system used to produce the crystal. One question that has not been addressed is whether or not there exist temperature distributions that produce less von Mises stress. Moreover, if such temperature profiles exist, what changes in the geometry of the growing environment are required?

Another interesting problem is that of the fluid flows. Some heuristic analysis was performed but this appears to be a finely balanced system between the Ekman pumping and the buoyancy flows. Further understanding of this system would be very worthwhile yet complicated by the rotation of the crucible.

The rate which the radius of the crystal grows depends on the effective pull rate and the angle the fluid makes with the extracted crystal. Computing the shape of the meniscus at various rotation rates illustrates that increasing the rotation rate to 10 rpm drops the location of the triple point about one half the height of the nonrotating meniscus. Since the shape of the meniscus determines the location of the triple point and it is near this triple point that much of

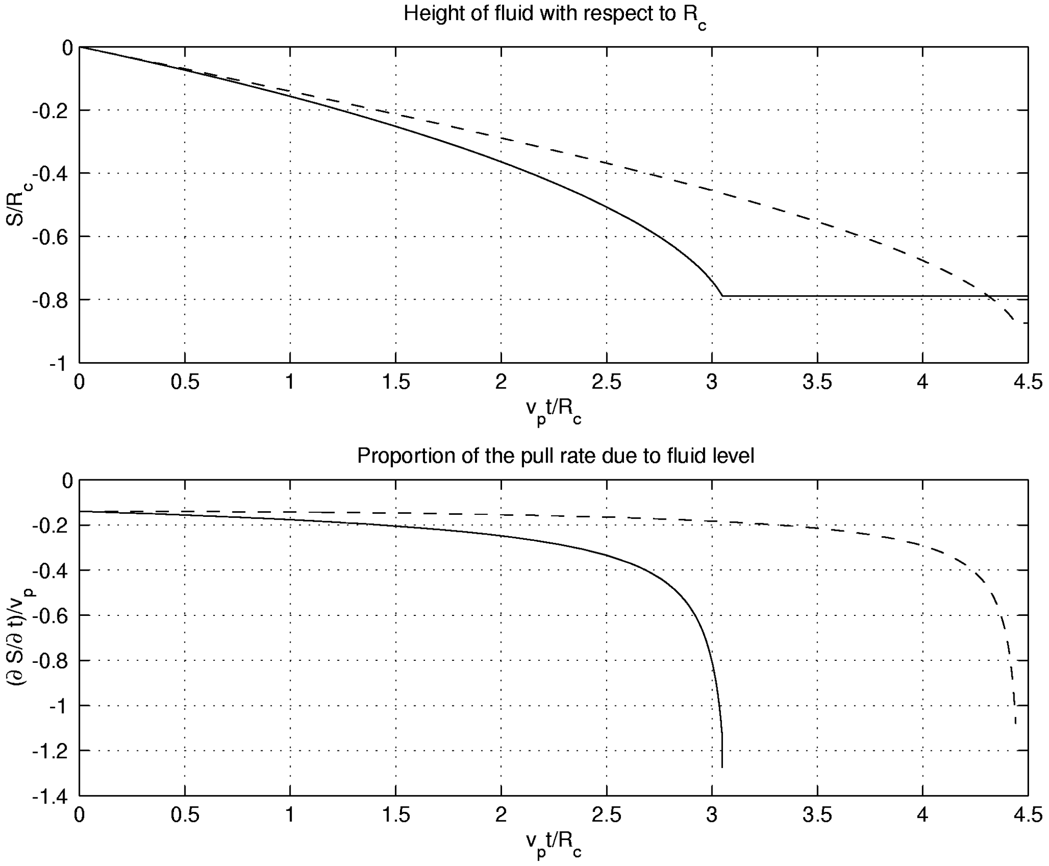


Figure 2.14: The height of the fluid and the proportion of the effective pull rate due to the falling fluid as a function of the non dimensionalized time. The solid line corresponds to $\varphi = 2^o$ while the dashed line is the case of a constant radius, $\varphi = 0$.

the thermal stress is generated, inclusion of this effect may be quite important in determining the overall shape of the crystal/melt interface.

Many aspects of the problem of growing InSb crystals were investigated in the hopes of understanding the growing process. Growing larger crystals seems to depend for the most part on controlling the temperature of the surrounding hydrogen gas. Other elements of the growing method were investigated and it is hoped that further work, perhaps on a model that incorporates most of these factors, will yield advances in this method.

# Bibliography

[1] Brummell, N., Hart, J.E. & Lopez, J.M., *On the flow induced by centrifugal buoyancy in a differentially-heated rotating cylinder*, Theoretical and Computational Fluid Dynamics, **14** (2000), 39-54.

[2] Cherepanov, G.P., *Two-dimensional convective heat/mass transfer for low Prandtl and any Peclet numbers*, SIAM Journal of Applied Mathematics, **58**(3), (1998), 942-960.

[3] Cook, R.D., Finite Element Modeling for Stress Analysis, John Wiley & Son Inc., 1995.

[4] Crowley, A.B., *Mathematical modelling of heat flow in Czochralski crystal pulling*, Journal of Applied Mathematics, **30** (1983), 173-189.

[5] Fowkes, N.D. & Hood, M.J., (1997). *Surface tension effects in a wedge*, The Quarterly Journal of Mechanics and Applied Mathematics, **51**(4), (1997), 553-561.

[6] Fowler, A.C., Mathematical Models in the Applied Sciences, Cambridge Texts in Applied Mathematics. Cambridge University Press, 1997.

[7] Hoffmann, K.A. & Chiang, S.T., Computational Fluid Dynamics for Engineers, Volume 1. www.EESbooks.com, 2000.

[8] Jordan, A.S., Caruso, R. & von Neida, A.R., *A thermoelastic analysis of dislocation generation in pulled GaAs crystals*, The Bell System Technical Journal, **59**(4), (1979), 593-637.

[9] Kusaka, Y. & Tani, A., *On the classical solvability of the Stefan problem in a viscous incompressible fluid flow*, SIAM Journal of Mathematics, **30**(3), (1999), 584-602.

[10] Lienhard, H.L., A Heat Transfer Textbook, (Second Ed.). Prentice-Hall Inc., 1987.

[11] Singh, A.K., Pardeshi, R. & Basu, B., *Modelling of convection during solidification of metal and alloys*, Sādhanā, **26**(1-2), (2001), 139-162.

[12] Stewart, R.H., Introduction to Physical Oceanography, Online textbook, 2002.

[13] Talmage, G., Shyru, S.-H., Lopez, J.M. & Walker, J.S., *Inertial effects in the rotationally driven melt motion during the Czochralski growth of silicon crystals with a strong axial magnetic field*, Zeitschrift für angewandte Mathematik und Physik, **51** (2000), 267-289.

[14] Wang, Y., Regel, L.L. & Wilcox, W.R., *Steady state detached solidification of water at zero gravity*, Journal of Crystal Growth, **226** (2001), 430-435.

[15] Zhang, Z.-C., Yu, S.T.J., Chang, S.-C. & Jorgenson, P.C.E., *Calculations of low-Mach-number viscous flows without preconditioning by the space-time CE/SE method*, Computational Fluid Dynamics 2000, Springer-Verlag, 2001, pp. 127-132.

# Chapter 3

# Adapting Search Theory to Networks

Brian Alspach[1], Kristen Jaskie[2], Theodore Kolokolnikov[3], Denis Kosygin[4], Nathan Krislock[5],
Aihna Li[5], Jacky Li[6], Eva-Marie Nosal[3], Moshe Rosenfeld[2], Sarah Sumner[7], Ling Zhao[8]
Report prepared by Brian Alspach

## 3.1 Introduction

The problem under study here was brought by Dr. Allan Douglas of the Communication Security Establishment. The Communication Security Establishment (hereafter referred to as the CSE) includes a team composed of Defence Scientists assigned from the Operational Research Division, National Defence Headquarters, Department of National Defence, Ottawa.

The CSE is interested in the general problem of locating objects in networks. Their interest in this type of problem arises because of the emerging concern regarding security issues for information operations.

The concept of transmitting code from one computer to another has been around for more than forty years. As a programming practice, it has evolved from an occasional concern of systems programmers working at the deepest levels of operating systems to a common and widespread practice. It is now utterly unremarkable for a web browser to bring in a web page containing a JAVA applet which then executes.

The next stage, in which objects move under conditions of their own determination, already is upon us. Development environments are being created to facilitate this. One of these, for example, is for the creation and use of "Aglets" which are examples of mobile network agents. Mobile network agents are programs that can be dispatched from one computer and transported to a remote computer for execution. At the remote computer they present their credentials in

---

[1] University of Regina
[2] University of Washington
[3] University of British Columbia
[4] Princeton University
[5] Loyola University
[6] Dalhousie University
[7] University of Ottawa
[8] University of Alberta

order to gain access to local services and data. It is apparent that mobile network agents are going to undergo considerable development and become extensively used. The networked world is going to see many of these objects.

Prudence dictates the expectation that some of the mobile objects will not be benign. Defensive information operations will have to deal with mobile attackers. The problems of determining whether or not attackers are present or likely to be present, and determining their present locations become of considerable interest.

The CSE people were familiar with the literature involving search theory, where the searches are carried out over two- and three-dimensional regions. The techniques for these kinds of searches typically involve partitioning the region into cells and considering the problem of getting the searcher (or one of several searchers) and the target in the same cell.

They also were familiar with an old paper by G. Pólya [34] in which he considered random searches on the $n$-dimensional grid with a single searcher and a single target. They were not aware of any other work done on searching in graphs.

Because of their exposure to search theory, the problem they brought to the workshop was phrased in terms of adapting search theory to networks. Thus, the first step was the introduction of an already existing healthy literature on searching graphs.

T. D. Parsons, who was then at Pennsylvania State University, was approached in 1977 by some local spelunkers who asked his aid in optimizing a search for someone lost in a cave in Pennsylvania. Parsons quickly formulated the problem as a search problem in a graph. After his paper [31] appeared, many subsequent papers appeared. Subsequent papers led to two divergent problems. One problem dealt with searching under assumptions of fairly extensive information, while the other problem dealt with searching under assumptions of essentially zero information. These two topics are developed in the next two sections.

## 3.2 Complete Information

There is a variety of models we may use for attempting to find an evader or evaders in a graph. The notion of *searching* a graph involves the evaders and searchers being located at vertices. Evaders and searchers may stay at their current locations, or move along an edge of the graph to a neighboring vertex. The simplest clock is one which ticks at regular intervals and all moves take place when the clock ticks. In one version, any subset of searchers and evaders may move at each tick. In a second version, subsets of searchers move on odd numbered ticks and subsets of evaders move on even numbered ticks — the point being that evaders and searchers alternate moves. In the most general version, the clock is continuous and participants may move at any time. In all of these models, capture takes place whenever an evader and a searcher occupy the same vertex at the same time.

Another decision which must be made for searching is whether or not capture takes place if an evader and a searcher "pass" each other along the same edge. For example, if at a tick of the clock, an evader moves along an edge from vertex $u$ to $v$ and a searcher moves along the same edge from vertex $v$ to vertex $u$, does capture take place?

The notion of *sweeping* a graph involves evaders and searchers being able to be located at vertices or along an edge. In this case, we view the graph as being embedded in euclidean space.

The movement of an evader or searcher corresponds to a continuous function from the interval $[0, \infty)$ to the embedded graph. If there are $k$ searchers, let $f_1, f_2, \ldots, f_k$ denote the $k$ functions describing the searchers' movements in the graph. If there are $\ell$ evaders, let $e_1, e_2, \ldots, e_\ell$ denote the $\ell$ functions describing the evaders' movements in the graph. Capture takes place whenever there exist $t, i, j$ such that $f_i(t) = e_j(t)$. In other words, a searcher and an evader occupy the same place at the same time.

Parsons' first paper on this topic [31] considered sweeping a graph. He made the following definition.

**Definition 3.2.1** Let $G$ be a finite graph. The *sweep number* of $G$, denoted $\mathrm{SW}(G)$, is the smallest integer $k$ such that $k$ searchers can sweep $G$ and capture a single evader.

He observed that $\mathrm{SW}(G)$ always exists because $|V(G)| + 1$ searchers can always capture an evader. This is done by placing a searcher at each vertex and then using an additional searcher to move along every edge of the graph. If the graph is not connected, the additional searcher may sweep each component separately.

He proved the following theorem as well.

**Theorem 3.2.2** *If $T$ is a tree and $k \geqq 1$, then $\mathrm{SW}(T) \geqq k + 1$ if and only if $T$ has a vertex $v$ at which there are at least three branches $T_1, T_2, T_3$ satisfying $\mathrm{SW}(T_j) \geqq k$ for $j = 1, 2, 3$.*

He was able to use the theorem to recursively characterize all trees with a given sweep number. At the end of [31], Parsons suggested many other variations of the problem for investigation.

A. LaPaugh [22] first proved that a graph may be optimally swept without going over any edge twice. D. Bienstock and P. Seymour gave a new proof in [5].

Other people took up the problem obtaining many results. There also was a shift towards considering searching rather than sweeping, and considering a single evader. The usual model was alternating moves and complete information, that is, everybody knows everybody's location.

**Definition 3.2.3** Let $G$ be a finite graph. The *search number* of $G$, denoted $\mathrm{SE}(G)$, is the smallest integer $k$ such that $k$ searchers can search $G$ and capture a single evader using alternate moves and with complete information.

One direction taken by various researchers was to consider classes of graphs. The class of Cayley graphs is an interesting class because of their use as models for network architectures. P. Frankl [17] was the first person to consider Cayley graphs. He proved the following result.

**Theorem 3.2.4** *If $X = X(G; S)$ is a Cayley graph on the finite abelian group $G$, then $\mathrm{SE}(X) \leqq (|S| + 1)/2$.*

Since the $n$-dimensional cube is a Cayley graph on a finite abelian group, the preceding result immediately yields the following corollary.

**Corollary 3.2.5** *If $Q_n$ is the $n$-dimensional cube, then $\mathrm{SE}(Q_n) \leqq (n + 1)/2$.*

M. Aigner and M. Fromme [1] characterized the class $\mathcal{C}$ of graphs for which $\mathrm{SE}(X) = 1$ whenever $X \in \mathcal{C}$. In particular, they showed that $\mathcal{C}$ contains all finite trees.

M. Maamoun and H. Meyniel [26] generalized the Aigner-Fromme result about trees in the following way.

**Theorem 3.2.6** *If $X$ is a cartesian product of $n$ finite trees, each of which has at least two vertices, then* $\mathrm{SE}(X) = \lceil (n+1)/2 \rceil$.

Aigner and Fromme [1] also proved the following interesting result for another important family of graphs.

**Theorem 3.2.7** *If $X$ is a planar graph, then* $\mathrm{SE}(X) \leqq 3$.

T. Andreae [3] generalized the preceding result in the direction of classes of graphs not containing a fixed graph as a minor. A. Quilliot [35] extended the result in the following direction.

**Theorem 3.2.8** *If $X$ is a graph of genus $k$, then* $\mathrm{SE}(X) \leqq 3 + 2k$.

The preceding results determine either upper bounds or exact values for the search number of graphs. P. Frankl [18] determined a lower bound in terms of girth and minimum degree.

**Theorem 3.2.9** *If $X$ is a graph with girth at least $8t-3$ and minimum degree larger than $d$, then* $\mathrm{SE}(X) > d^t$.

A negative result about lower bounds was established by T. Andreae [2]. He proved that for any positive integers $m$ and $d$, there exists a regular graph of degree $d$ for which the search number is bigger than $m$.

In [4] it is shown that for each fixed $m$, there is a polynomial time algorithm determining whether or not a fixed graph $X$ satisfies $\mathrm{SE}(X) \leqq m$. Complexity also was studied by A. Goldstein and E. Reingold in [19].

R. Nowakowski and P. Winkler [30] established a structural result by characterizing the graphs for which one searcher can always capture one intruder under the complete information model.

## 3.3 Zero Information

Zero information means that nobody knows anything about the location of anyone else. In fact, the searchers do not know if there even is an intruder in the graph they are searching. The problem is to devise a search mechanism whereby $m$ searchers are guaranteed of finding an intruder or establishing that the graph is free of any intruders.

Almost all the work which has been done on this version is the study of collision properties of two random walks in graphs. The random search assumptions are that if someone is located at a vertex of valency $d$, then at the next tick of the clock, he moves to a neighboring vertex with probability $1/d$ for each of the $d$ neighbors. Under this assumption, the following theorem holds.

**Theorem 3.3.1** *Let $u$ and $v$ be any two distinct vertices of a graph $X$, let* $\mathrm{dist}(u,v)$ *denote the distance between $u$ and $v$ in $X$, let $\Delta(X)$ denote the maximum valency of $X$, let $T(u,v)$ denote the first time two random walks beginning at $u$ and $v$ occupy the same vertex, and let* $\mathrm{ET}(u,v)$ *denote the expected value of $T(u,v)$. For any two distinct vertices $u$ and $v$ of $X$,*

- $\mathrm{Prob}(T(u,v) < \infty) = 1$,
- $\mathrm{ET}(u,v) < 5\Delta(X)\mathrm{dist}(u,v)|V(X)|$, *and*
- *for every* $\epsilon > 0$, *there exists* $C(\epsilon) > 0$ *such that* $\mathrm{Prob}(T(u,v) > C(\epsilon)|V(X)|) \geqq 1 - \epsilon$.

Increasing the number of searchers does materially affect the preceding result. For a good discussion of this material see [23].

## 3.4 Workshop

On the afternoon of the first day, the group had its first meeting. Several participants were aware of some of the existing literature on searching and sweeping graphs. Our first course of action was to launch an extensive literature search, visit the excellent mathematics library on campus, and make copies of those papers which seemed most relevant.

By the end of the afternoon, most of the references given in the bibliography had been discovered and copies of approximately ten of the papers were distributed to the group. The object was to be able to begin to discuss their contents Tuesday morning.

Tuesday morning was spent discussing the contents of the papers, and exploring the potential usefulness with regard to the problem Dr. Douglas had in mind. Dr. Douglas himself found some of the results of considerable interest, but was mostly impressed by the fact these kinds of problems had been examined by a variety of researchers.

It became clear that different subgroups were becoming interested in pursuing different aspects of the problem. The rest of Tuesday was spent trying to work out clear objectives for different subgroups. Simultaneously, we were being presented with a nice description of random walks in graphs by Dr. Kosygin. By the time Tuesday evening rolled around, it was clear we still were not well organized.

Wednesday morning was spent completing the development of objectives for several subgroups. The subgroups then went their separate ways agreeing to meet late in the afternoon to sum up what they had accomplished during the day.

One subgroup was working on various problems dealing with trees. Their first foray into this was an attempt to independently work out proofs of several of the results on trees that we found in the literature. After coming to grips with the proofs, they then began to think about possible algorithms for computing search numbers for trees.

One approach is based on reducing a tree by removing all the leaves in a single stage. After this stage, the smaller tree then has its leaves removed. This is a layered approach and is recursive. It is shown that one less searcher is needed at each stage.

A divide and conquer approach for trees is based on deleting a central vertex, leaving a searcher at the deleted vertex to prevent the target from moving from one subtree into another subtree, and then searching each subtree separately. It is clear that the minimum number of searchers needed to search the entire tree is one more than the maximum number needed to search any of the subtrees created by the deletion of the vertex.

The last activity undertaken was an attempt to modify some of the known results on trees to other search models. They made a presentation to the group on Thursday.

A second subgroup wanted to look at the behavior of random searches. They decided to concentrate on the $n$-dimensional cube, with some consideration of complete graphs.

Let's look at what was done for the 10-dimensional cube first. The following table contains information on how long it took six searchers, using random search, to capture one target in the 10-dimensional cube, and the following table one searcher.

The column headed *Frequency* gives the number of random searches using a number of steps in the range shown in the column headed *Time*.

| Frequency | Time |
|---|---|
| 1 | 1 – 71 |
| 71 | 72 – 142 |
| 57 | 143 – 213 |
| 38 | 214 – 285 |
| 29 | 286 – 356 |
| 18 | 357 – 427 |
| 16 | 428 – 499 |
| 9 | 500 – 570 |
| 7 | 571 – 641 |
| 2 | 642 — 713 |
| 3 | 714 – 784 |
| 3 | 785 – |

SIX SEARCHERS, ONE TARGET

| Frequency | Time |
|---|---|
| 1 | 11 – 366 |
| 72 | 367 – 722 |
| 50 | 723 – 1,080 |
| 38 | 1,081 – 1,435 |
| 24 | 1,436 – 1,792 |
| 24 | 1,793 – 2,148 |
| 18 | 2,149 – 2,505 |
| 5 | 2,506 – 2,861 |
| 8 | 2,862 – 3,217 |
| 5 | 3,218 — 3,574 |
| 3 | 3,575 – 3,930 |
| 1 | 3,931 – 4,287 |
| 3 | 4,288 – 4,643 |
| 5 | 4,644 – |

ONE SEARCHER, ONE TARGET

The next table provides some data correlating the number of searchers and the average number of steps required to capture a single target, using random search, in the 10-dimensional cube. The data indicates a definite trend, but many more trials are necessary to increase the accuracy. The column headed *Time* is the average number of steps required using the number of searchers under the column headed *Number*.

| Number | Time |
|---|---|
| 1 | 1,060 |
| 2 | 567 |
| 3 | 412 |
| 4 | 220 |
| 5 | 20 |
| 6 | 195 |
| 7 | 170 |
| 8 | 140 |
| 9 | 130 |
| 10 | 120 |
| 11 | 95 |
| 12 | 103 |
| 13 | 82 |
| 14 | 94 |
| 15 | 72 |
| 16 | 70 |
| 17 | 62 |
| 18 | 60 |
| 19 | 55 |
| 20 | 51 |

Number of searchers and time

For one searcher and one target, the next table relates steps to capture and the dimension of the cube.

| Dimension | Time |
|---|---|
| 1 | 0 |
| 2 | 2 |
| 3 | 10 |
| 4 | 25 |
| 5 | 40 |
| 6 | 85 |
| 7 | 130 |
| 8 | 320 |
| 9 | 590 |
| 10 | 1100 |

Dimension and time

We ran random searches, using one searcher and one target, on complete graphs whose orders were multiples of 100. We started with 100 vertices and went through 2,000 vertices. The number of trials again needs to be increased considerably in order to introduce more accuracy. Still, the average time behaved reasonably except for 1,300 and 1,400 vertices. From 100 through 1,800 vertices, the average number of steps required was approximately equal to the number of vertices. For 1,900 and 2,000 there were steep increases. There is a lot of room for studying this question more.

The subgroup working on generating random searches also made a presentation to the group on Thursday.

The rest of Thursday was used for amalgamating our efforts into a report to be presented Friday.

## 3.5 Summary

The following summarizes what we discovered during the week.

- Search results are greatly affected by the paradigm used.
- Searches on well defined families of graphs already present challenging problems.
- Simulations on random graphs and the $n$-dimensional cube agreed with computed expectations.
- Random searches usually perform as well as structured searches.
- The surface of the problem has been only scratched.

## 3.6 Acknowledgements

The team wishes to thank the University of Washington and the organizers for providing us with wonderful facilities, well organized structure, and excellent responses to our requests, even those that might have been somewhat unreasonable.

Most of all we wish to thank Dr. Allan Douglas of CSE for his patience, humor, detailed explanations in response to our questions, willingness to spend the entire week at the session, and the stimulating question he brought to the Workshop.

I, personally, want to thank Allan for his encouragement to go ahead and develop a MITACS proposal based on the Workshop experience. A proposal has been submitted and at the time of this report, we are awaiting a final decision.

# Bibliography

[1] Aigner, M. and Fromme, M., *A game of cops and robbers*, Discrete Appl. Math., **8** (1984), 1–12.

[2] Andreae, T., *Note on a pursuit game played on graphs*, Discrete Appl. Math., **9** (1984), 111–115.

[3] Andreae, T., *On a pursuit game played on graphs for which a minor is excluded*, J. Combin. Theory Ser. B, **41** (1986), 37–47.

[4] Berarducci, A. and Intrigila, B., *On the cop number of a graph*, Adv. in Appl. Math., **14** (1993), 389–403.

[5] Bienstock, D. and Seymour, P., *Monotonicity in graph searching*, J. Algorithms, **12** (1991), 239–245.

[6] Brightwell, G. and Winkler, P., *Maximum hitting time for random walks on graphs*, J. Random Structures and Algorithms, **1** (1990), 263–276.

[7] Clarke, N.E., The Effects of Replacing Cops and Searchers with Technology, M.Sc. Thesis, Dalhousie University, 1999.

[8] Clarke, N.E. and Nowakowski, R.J., *Cops, robber and photoradar*, Ars Combinatoria, **LVI**, 97–103.

[9] Clarke, N.E. and Nowakowski, R.J., *Cops, robber and traps*, Utilitas Math., (to appear).

[10] Coppersmith, D., Tetali, P. and Winkler, P., *Collisions among random walks on a graph*, SIAM J. Discrete Math., **6**, (1993), 363–374.

[11] Ellis, J.A., Sudborough, I.H. and Turner, J.S., *The vertex separation and search number of a graph*, Inform. and Comput. **113** (1994), 50–79.

[12] Fitzpatrick, S.L., *The isometric path number of the Cartesian product of paths*, Proc. Thirtieth Southeastern Conf. Combinatorics, Graph Theory, and Computing, Congr. Numer. **137** (1999), 109–119.

[13] Fitzpatrick, S.L. and Nowakowski, R.J., *Copnumber of graphs with strong isometric dimension two*, Ars Combin. **59** (2001), 65–73.

[14] Fitzpatrick, S.L. and Nowakowski, R.J., *The strong isometric dimension of finite reflexive graphs*, Discuss. Math. Graph Theory, **20**, 23–38.

[15] Fomin, F.V., *Note on a helicopter search problem on graphs*, Proc. Conference Optimal Discrete Structures and Algorithms—ODSA '97, Discrete Appl. Math., **95** (1999), 241–249.

[16] Fomin, F.V., *Helicopter search problems, bandwidth and pathwidth*, Discrete Appl. Math., **85** (1998), 59–70.

[17] Frankl, P., *On a pursuit game on Cayley graphs*, Combinatorica, **7** (1987), 67–70.

[18] Frankl, P., *Cops and robbers in graphs with large girth and Cayley graphs*, Discrete Appl. Math., **17** (1987), 301–305.

[19] Goldstein, A.S. and Reingold, E.M., *The complexity of pursuit on a graph*, Theoret. Comput. Sci., **143** (1995), 93–112.

[20] Hahn, G. and Tardif, C., *Graph homomorphisms: structure and symmetry*, Graph Symmetry, Kluwer Acad. Publ., Dordrecht, 1997, 107–166.

[21] Hamidoune, Y.O., *On a pursuit game on Cayley digraphs*, European J. Combin., **8** (1987), 289–295.

[22] LaPaugh, A., *Recontamination does not help to search a graph*, Technical Report No. 335, Department of Electrical Engineering and Computer Science, Princeton University, 1982.

[23] Lovász, L., *Random Walks on graphs: a survey*, Combinatorics, Paul Erdös is Eighty, Vol. 2, Keszthely, 1993, 353–397.

[24] Lovász, L. and Winkler, P., *A note on the last vertex visited by a random walk*, J. Graph Theory, **17** (1993), 593–596.

[25] Lovász, L. and Winkler, P., *Mixing of random walks and other diffusions on a graph*, Surveys in Combinatorics, London Math. Soc. Lecture Note Ser., **218**, Cambridge University Press, 1995, 119–154.

[26] Maamoun, M. and Meyniel, H., *On a game of policemen and robbers*, Discrete Appl. Math., **17** (1987), 307–309.

[27] Neufeld, S., The Game of Cops and Robber, M.Sc Thesis, Dalhousie University, 1990.

[28] Neufeld, S. and Nowakowski, R.J., *A game of cops and robber played on products of graphs*, Discrete Math., **186**, 253–268.

[29] Nowakowski, R.J., *Search and sweep numbers of finite directed acyclic graphs*, Discrete Applied Math., **41** (1993) 1–11.

[30] Nowakowski, R. and Winkler, P., *Vertex-to-vertex pursuit in a graph*, Discrete Math., **43** (1983), 235–240.

[31] Parsons, T.D., *Pursuit evasion in a graph*, Lecture Notes in Math., **642**, Springer-Verlag, Berlin, 1978, 426–441.

[32] Parsons, T.D., *The search number of a connected graph*, Proc. Ninth Southeastern Conf. Combinatorics, Graph Theory, and Computing, Cong. Num., **21** (1978), 549–554.

[33] Polat, N., *Retract-collapsible graphs and invariant subgraph properties*, J. Graph Theory, **19** (1995).

[34] Pólya, G. *Über eine Aufgabe der Wahrscheinlichkeitsrechnung betreffend die Irrfahrt in Strassennetz*, Math. Ann., **84** (1921), 149–160.

[35] Quilliot, A., Thèse d'Etat, Université de Paris VI, 1983.

[36] Quilliot, A., *A short note about pursuit games played on a graph with a given genus*, J. Combin. Theory Ser. B, **38** (1985), 89–92.

[37] Quilliot, A., *About the complexity of some homomorphism problems on graphs*, Topics in Combinatorics and Graph Theory, Physica, Heidelberg, 1990, 561–567.

[38] Schroeder, B.S.W., *The copnumber of a graph is bounded by* $\lfloor \frac{3}{2}\ \mathit{genus}\ (G) \rfloor + 3$, Trends Math., Birkhäuser Boston, 2001, 243–263.

[39] Tošić, R., *On cops and robber game*, Studia Sci. Math. Hungar., **23** 1988.

[40] Winkler, P., *Dependent percolation and colliding random walks*, Random Structures Algorithms, **16** (2000), 58–84.

# Chapter 4

# An Automated Algorithm for Decline Analysis

Rita Aggarwala[1], Tom Alberts[2], Chris Bose[3], Adrian Driga[1], Aude Espesset[4], John Harlim[5], Jihyoun Jeon[6], Seungwon Jeon[7], Qutaibeh Katatbeh[8], Larry Kolasa[9], Melvin Leok[10], Mufeed Mahmoud[4], Rafael Meza[6], Alberto Nettel-Aguirre[1], Christina Popescu[2], Randall Pyke[9], Mariana Carrasco Teja[6],
Report prepared by Randall Pyke

## 4.1 Introduction

Oil and gas wells are regularly monitored for their production rates. Typically, daily production rate data is available, expressed in millions of standard cubic feet per day (MMscf/d) for natural gas wells or barrels per day (Bbl/d) for liquids, i.e., oil or water. This data reflects changing physical conditions within the oil or gas reservoir, changes in equipment (eg. failure or maintenance), activity of surrounding wells, variability in outshipping methods, and changing production rates due to economic factors. As a result, typical production rate data is noisy and highly discontinuous.

Decline analysis is a process that extrapolates trends in the production rate data from oil and gas wells to forecast future production and ultimate cumulative reserve recovery. Current software often attempts a best fit approach through all the data, but the result is erroneous in the majority of cases. A human operator with an understanding of the factors that affect the

---

[1] University of Alberta
[2] University of Calgary
[3] University of Victoria
[4] University of Western Ontario
[5] University of Guelph
[6] University of Washington
[7] University of Texas at Austin
[8] Concordia University
[9] Ryerson University
[10] California Institute of Technology

behavior of oil and gas wells can do a much better job of forecasting appropriately; however, it is a time-consuming process.

The goal is to find an algorithm that can be easily interfaced with standard industrial software and that incorporates some of the criteria used in the human analysis so as to perform acceptable forecasts in the majority of cases.

## 4.2 Proposed Solution

The proposed solution consists of three main steps: (1) Segmentation of Data, (2) Curve fitting, and (3) a Decision Process. Segmentation of Data attempts to identify intervals in the data where a single trend is dominant. A curve from an appropriate family of functions is then fitted to this interval of data. The Decision Process gauges the quality of the trends identified and either formulates a final answer or, if the program cannot come to a reliable answer, 'flags' the well to be looked at by an operator.

### 4.2.1 Segmentation

The input data is assumed to consist of a time series, $\{U(i)\}_{i=1}^{N}$, where $U(i)$ represents the data point (rate of oil/gas flow from the well) at time $i$. We assume the list is contiguous, i.e., there is a data point for each time step (the length of a time step is input by the operator). In this stage of the analysis the data set is divided into segments. Each segment will be analyzed in subsequent stages.

The end points of the segments are determined by (1) discontinuities in the data, and (2) discontinuities in the slope of the mean data (changes in trend). Two methods were developed for detecting these types of discontinuities; one method for detecting type (1) discontinuities, and one method for detecting type (2) discontinuities.

Because the data is typically very noisy, the data is smoothed a number of times. This smoothing reduces the amplitude of the oscillations of the noise relative to the amplitudes of the discontinuities in the data, making the discontinuities easier to identify.

Each smoothing operation is obtained by moving averages with a window of width three. Let $\{U_{\mathrm{S}}^{l-1}(i)\}_{i=1}^{N}$, $l \geq 1$, denote the data after being smoothed $l-1$ times. Then the next smoothed version of the data, $\{U_{\mathrm{S}}^{l}(i)\}_{i=1}^{N}$, is obtained by the formula;

$$U_{\mathrm{S}}^{l}(i) \;=\; \frac{1}{3}\Big(U_{\mathrm{S}}^{l-1}(i-1) + U_{\mathrm{S}}^{l-1}(i) + U_{\mathrm{S}}^{l-1}(i+1)\Big).$$

The number of times the data is smoothed depends on the length $N$ of the data set. We found from experience that an appropriate value for the number of times $k$ the data should be smoothed is $k = \lfloor \log_2(N) \rfloor$. (From now on, "smoothed data" will mean $\{U_{\mathrm{S}}^{k}(i)\}_{i=1}^{N}$.)

After the data is smoothed a data set of differences $\{U_{\mathrm{d}}(i)\}_{i=1}^{N}$ is produced, where

$$U_{\mathrm{d}}(i) = |U(i) - U_{\mathrm{S}}^{k}(i)|.$$

That is, the differences are the (absolute value of) oscillations of the data around the smoothed data. The method for detecting type (1) discontinuities analyzes these differences.

The type (1) discontinuities in the data show up as larger peaks in the differences. However, because the data is typically very noisy, and to facilitate the identification of the locations of peaks (as described below), the differences are also smoothed a number of times, precisely, $k/2$ times (where $k$ is as above), and in the same manner as the data was smoothed, i.e., by moving averages with a window of width three. This has the effect of suppressing oscillations in the differences $\{U_{\mathrm{d}}(i)\}_{i=1}^{N}$ that are due to noise in the data, and thereby enhancing the peaks in the differences that are due to discontinuities in the data.

The discontinuities in the data are identified as the largest peaks in the (smoothed) differences. This is accomplished by locating peaks in the differences that are above some threshold. The threshold is set at 15% of the mean of the (original) data. That is, a peak in $\{U_{\mathrm{d}}(i)\}_{i=1}^{N}$ is identified as a point of discontinuity in the data if the amplitude of the peak is greater or equal to 15% of the mean value of the data.

The locations of the peaks are found by identifying local maxima in the differences that are above the threshold. Here we look for points where the (finite-difference) derivative of the differences change from positive to negative. That is, if $D(i) = U_{\mathrm{d}}(i+1) - U_{\mathrm{d}}(i)$, then if $D(i) > 0$ and $D(i+1) < 0$ we record $i$ as a point of discontinuity in the data (provided that $U_{\mathrm{d}}(i)$ is above the threshold).

Now we look for discontinuities in the slope of the (mean) data (i.e., the type (2) discontinuities). For each interval obtained above (from looking for type (1) discontinuities), a value for the derivative of the ’mean’ data at that point is obtained by computing the derivative of the best-fit parabola at that point. The parabola is fitted only to the data within 15 points on either side of the point under consideration. This gives a time series of ’mean’ derivative values of the data in the interval. This time series is fed into the program that finds type (1) discontinuities. The output is a list of points where the slope of the ’mean’ data has a discontinuity.

The locations of the two types of discontinuities are combined into one list - the list of end points of intervals for the data. This list is passed on to the next stage in the analysis.

### Wavelet analysis

Another approach to determining the segmentation of the data uses wavelets (see for example [1]). Wavelets are functions that cut up the data into different temporal and frequency components, and then study each component with a resolution matched to its scale. They are highly useful in analyzing physical situations where the signal is discontinuous. In this work, given the noisy data, wavelets are used to de-noise the data and to divide time series into segments.

The technique works in the following way. When you decompose a data set using wavelets, you use filters that act as averaging filters and others that produce details. Some of the resulting wavelet coefficients correspond to details in the data set. If the details are small, they might be omitted without substantially affecting the main features of the data set. The idea of thresholding, then, is to set to zero all coefficients that are less than a particular threshold. We then use an inverse wavelet transformation to reconstruct the data set. The de-noising is carried out without smoothing out the sharp structures. The result is a cleaned-up signal that still shows important details. In our case we used a Haar Wavelet base function to de-noise the noisy decline data and also to determine the changes in the production dynamics.

### 4.2.2 Curve Fitting and EUR

The previous stage in the analysis produced a list $\{a_1, \ldots, a_m\}$ of end points of intervals $I_1 = (1, a_1),\ I_2 = (a_2, a_2), \ldots,\ I_{m+1} = (a_m, N)$ for the data $\{U(i)\}_{i=1}^{N}$. The second stage in the analysis performs a least squares curve fitting to each of the intervals. That is, curve fitting is applied to the time series
$\{U(i)\}_{i=1}^{a_1},\ \{U(i)\}_{i=a_1}^{a_2},\ \ldots,\ \{U(i)\}_{i=a_m}^{N}$. The actual intervals of data used in the curve fitting are slightly smaller than these (to remove transient effects).

The class of curves used in the fitting belong to the family

$$q(t) \;=\; q_o(1 + nDt)^{-1/n} \tag{4.1}$$

where $q_o, n$ and $D$ are parameters. It has long been accepted within the petroleum industry that this function accurately models the uninterrupted flow rate of a well, and it can be also derived from basic physical principles.

Once we have found the best fit for the kth interval $I_k$, we compute a number of summary statistics;

- $q_{o,k},\ n_k,\ D_k$ ; least squares estimates of $q_o$, $n$ and $D$, and
- $S_k^2 \;=\;$ normalized variance of the data over the interval $= \sum_{i \in I_k} |U(i) - q_k(i)|^2 / l_k$ where $l_k$ is the length of $I_k$ and $q_k(t)$ is the curve fit for the kth interval.

The most important of these statistics is EUR$_k$, the Estimated Ultimate Recovery based on the best fit curve of the kth interval. It is determined by solving for the time at which the best fit curve passes below a minimum threshold rate, call this time $T_k$, and then computing the sum;

$$\mathrm{EUR}_k \;=\; \sum_{1}^{a_k - 1} U(i) \;\;+\;\; \sum_{i=a_k}^{T_k} q_k(i) \tag{4.2}$$

where the first sum simply represents the volume of oil or gas that has already been produced, and the second is the amount we expect to produce based on the curve fit $q_k(t)$ for the kth interval. These statistics are then passed as parameters to a weight function which decides their relevance and usefulness, and based on this we can calculate a final EUR value.

### 4.2.3 Decision Process, Final Estimated Ultimate Recovery (EUR), and Overall Rate Curve

Given the parameters $p_k = (n_k, q_{o,k}, D_k, \mathrm{EUR}_k, S_k^2, l_k)$, $\;k = 1, ..., m+1$ for the fits over the intervals $I_1, ..., I_{m+1}$, we choose a weight function $w_k = w_k(p_k)$ that indicates how 'reliable' the EUR$_k$ is. The following conditions may be considered as rules for a reliable EUR$_k$;

- $I_k$ is a long data set,
- the variance $S_k^2$ over $I_k$ is small,
- $0 \le n \le 0.5$ (a physically plausible hyperbola), and
- $I_k$ is a recent interval.

The weight reflects the importance of the hyperbolic curve fit over interval k in determining the final EUR. The precise formula incorporating the factors above was chosen to be;

$$w_k = \frac{\frac{(l_k - r)^+ h(n_k)}{S_k^2 (N - a_k)}}{\sum_{i=1}^{m+1} \frac{(l_i - r)^+ h(n_i)}{S_i^2 (N - a_i)}}. \tag{4.3}$$

Here, as above, $l_k$ is the length of interval $k$, $m+1$ is the total number of intervals obtained from the segmentation stage of the analysis, $r$ is an integer such that intervals of length less than or equal to $r$ will not be used, and $(l_k - r)^+$ denotes the positive part of $l_k - r$ (i.e., $(l_k - r)^+ = l_k - r$ if $l_k - r \geq 0$, and $(l_k - r)^+ = 0$ if $l_k - r < 0$). $n_k$ is the estimate of the model parameter $n$ for interval $k$ (cf. Equation (1.1) ), and $h(n)$ is a function which gives the parameter $n$ from the hyperbolic fits a separate weighting, for example,

$$\begin{aligned} h(n) &= 1 \text{ if } n \in [0,\ 0.5] \\ &\phantom{=}\ 2(1 - n) \text{ if } n \in (0.5,\ 1] \\ &\phantom{=}\ 0 \text{ if } n > 1. \end{aligned}$$

(Engineering knowledge indicates that low values of the model parameter $n$ indicate a well which is past its period of transient activity.) A similar function could instead be applied. $S_k^2$ is the normalized sum of squared residuals between the curve $q_k$ and the data in the interval $I_k$ (see above).

Once the weights have been determined, the EUR is calculated with the formula,

$$\widehat{\mathrm{EUR}} = \sum_{k=1}^{m+1} w_k \times \mathrm{EUR}_k \tag{4.4}$$

where $\mathrm{EUR}_k$ is the EUR which would be predicted using the curve fitted from interval $k$ (cf. Equation (1.2) ). The following provisions must also be implemented:

- If $EUR_k$ is less than the amount of oil which has already been recovered, $w_k$ is set to zero.
- If $EUR_k$ is calculated to be infinite, $w_k$ is set to zero.

Finally, we would like to be able to predict the amount of production between the end of the observed data and any future time $t$ using a single, ‘overall’ hyperbolic curve of the form given in Equation (1.1). Engineering practice allows us to use for the estimate of the parameter $n$ of this ’overall’ curve, the convex combination of the $n_k$, namely,

$$\hat{n} = \sum_{k=1}^{m+1} w_k \, n_k.$$

This, along with the following two equations, will give us the parameters of a single hyperbolic curve.

For the final time point $N$,

$$\hat{q}_0(1+\hat{n}\hat{D}N)^{-1/\hat{n}} = q_N, \tag{4.5}$$

(cf. Equation (1.1) ), the observed rate at time $N$. Alternatively, we could use the observed rate at the last acceptable time point used for curve fitting, or an average of the last few acceptable observed rates for the right hand side of the equation.

For the $\widehat{\text{EUR}}$,

$$\hat{q}_0\left(1+\frac{\hat{D}(\hat{n}-1)\widehat{EUR}}{\hat{q}_0}\right)^{-\frac{1}{\hat{n}-1}} \; = \; 1. \tag{4.6}$$

This is simply stating that at the time of $\widehat{\text{EUR}}$, we will be pumping at a rate of 1 barrel per day. Some other acceptable cut-off value could be used instead. Note that this equation is obtained by writing the rate function in terms of the cumulative production at time $t$. This transformation is easily obtained by integrating the original rate function considered, $q_t = q_0(1+nDt)^{-1/n}$ (with respect to $t$), to come up with a cumulative function, solving for time and substituting back into the original rate function again.

Equations (1.5) and (1.6) can be reduced to a single variable problem by solving Equation (1.5) for $\hat{q}_0$ and then substituting into Equation (1.6). Then, only $\hat{D}$ would need to be solved for numerically.

When all three parameters in the final hyperbolic curve are estimated, we can make forward predictions for rate and cumulative production at future time points.

## 4.3 Discussion

A more thorough testing of the algorithm presented here would include;

- More sophisticated techniques such as wavelets or neural networks (see below) could be used in the segmentation stage if the present method (of moving averages) turns out to be unreliable.

- An examination of the uncertainties in the curve fits and the final predictions made.
- Fine tuning the weighting functions $w_k$.
- Verification that the parameters for the final ('overall') hyperbolic curve are realistic.
- A more robust approach to the estimates obtained in stage 3; initial tests show that Equations (1.5) and (1.6) may be unstable.

One may consider using a neural network in the analysis (see for example [2]). Given sufficiently many data sets it may be sensible to avoid the choice of a heuristic weighting scheme such as given by Equation (1.3) in favour of a neural net approach where we would allow the algorithm to find good weighting schemes through training. Furthermore, it may be possible to invoke some training components in the segmentation algorithm as well. More specifically, the choice of the important thresholding parameters in the smoothing algorithm seems to be a likely candidate for a neural net approach.

# Bibliography

[1] Daubechies, Ingrid, *Ten Lectures on Wavelets*, CBMS-NSF Regional Conference Series in Applied Mathematics, Society for Industrial and Applied Mathematics (SIAM), Philadelphia, PA, 1992.

[2] de Wilde, Philippe, Neural Network Models, Springer-Verlag, New York, NY, 1997.

# Chapter 5

# Web Hosting Service Level Agreements

Alan King (Mentor)[1], Mehmet Begen[2], Monica Cojocaru[3], Ellen Fowler[2], Yashar Ganjali[4],
Judy Lai[5], Taejin Lee[6], Carmeliza Navasca[7], Daniel Ryan[2]

Report prepared by Alan King.

## 5.1 Introduction

This paper proposes a model for a relatively simple Web hosting provider. The model assumes the existence of a load-dispatcher and a finite number of Web-servers.

We quantify the quality of service towards the clients of this facility based on a service level agreement between the two parts: the web hosting provider and the client. We assume that the client has the knowledge and resources to quantify its needs. Based on these quantifications, which in our model become parameters, the provider can establish a service offer. In our model, this offer covers the quality of service and the price options for it.

The paper is organized as follows: in Section 2 there is a description of the parameters requested from the client and the provider's offers. In Section 3 we present the mathematical formulation of the model and its dynamics. In Section 4 we introduce an algorithm for the provider's resource allocation of Web servers in order to optimally serve potential clients, within the quality of service stated. The algorithm, consistent with the provider's offer, assures substantial profits serving clients requesting a bigger volume of data transfer. Section 5 is an outline of proposed future developments of this model.

---

[1] IBM Thomas J. Watson Research Center
[2] University of British Columbia
[3] Jeffery Hall, 229 Queen's University, Canada
[4] University of Waterloo
[5] University of Texas, Austin
[6] Kangwon National University & APCTP, Korea
[7] University of California at Davis

## 5.2 Service Level Agreement

In this section we describe a possible agreement between the provider and the clients. We assume that the provider as well as the customer are classifying the requests in classes depending on the file size. Each class will therefore have a different service time.

The customers are requested to quantify specific values for the following three parameters for the requests they experience:

- estimates, on average, of the arrival rate of file sizes
- probability distribution of file sizes
- probability distribution of service times

With these quantifications/parameters, the provider's offer includes

1. Base Service Level $L$ represents the base service, the maximum number of servers $L$ to be allocated to a specific class of requests based on the parameters offered by the client.
2. Per-Unit Bid $B$ represents the variable rate the client agrees to pay for adding servers beyond Base Service Level. This might happen if its number of requests increases beyond the estimated level covered in (1). Service up to the Base Service Level $L$ is guaranteed. Requests that exceed $L$ are satisfied if possible when the per-unit bid equals or exceeds the spot market price. Per-Unit Bid can equal 0 or the host's minimum variable charge $M$ (i.e. cost + economic profit). In other cases the bid is an explicit customer-supplied and changeable bid $B$, $B \geq M$. Whether the bid is 0, $M$, or $B$, it reflects the nature of the customer:
    - the customer wants no service beyond its base level; its implicit bid is 0
    - the customer wants service beyond its base level; its implicit bid is M
    - the customer wants requests beyond its base level to be completed; its explicit bid is B.
3. Quality of Service (level $z$, probability $\alpha$ ) is denoted by (QoS) and is defined in terms of response time $RT$ by modelling the probability that requests are completed within a specified service time $z$

$$P[\text{RT} > z] < \alpha.$$

The number $\alpha$ is a probability level.

## 5.3 Dynamics

Due to the complexity of the problem, we apply queuing approximations from [4]. Let $k = 1, \ldots, K$ index the distinct service classes, and $j = 1, \ldots, J$ index the customers. Each class will have a Service Level Agreement, i. e. the parameters identified in the first section will all be indexed by $k$. The system is composed of an incoming stream of requests with a known Poisson distribution with parameter $\lambda_k^{(j)}$. There are $i = 1, \ldots, I$ servers and service time is exponential with parameter $1/\mu_{ik}^{(j)}$. See figure below.

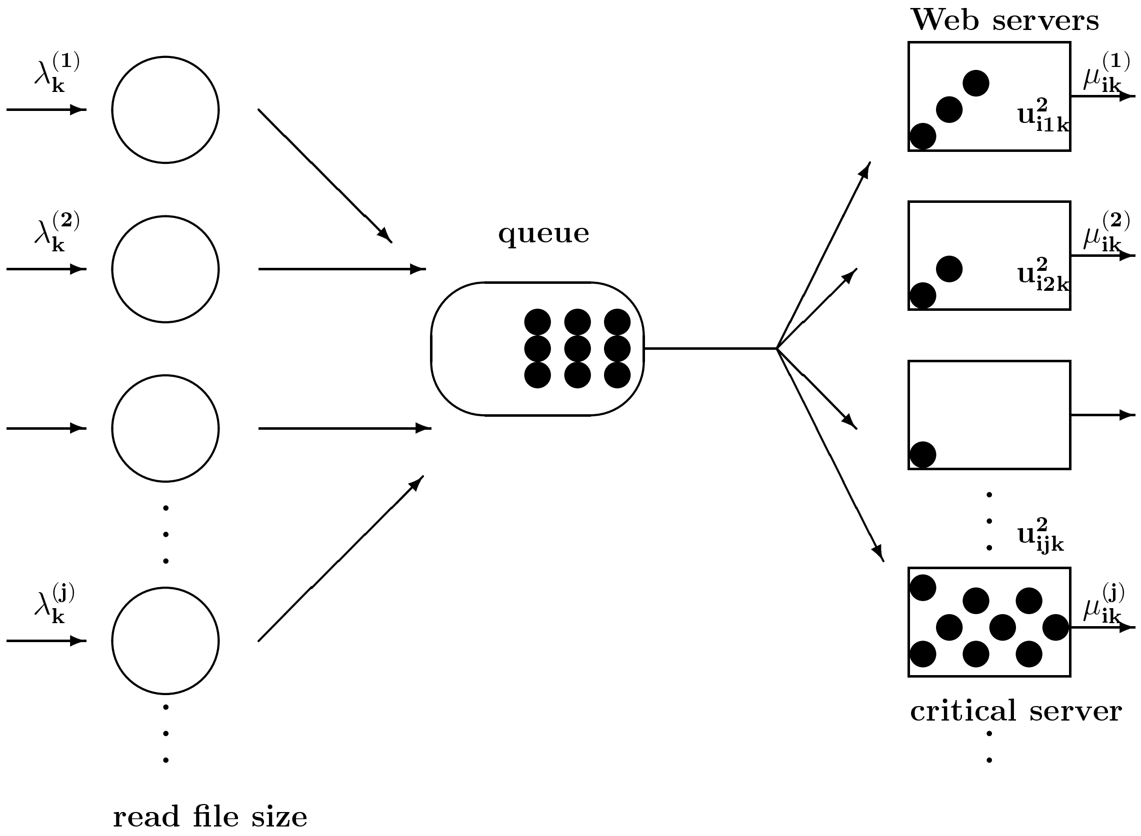


Figure 1: The figure describes the flow of requests with several web servers. All the accepted requests are completely served (i.e. no delay occurs between the moment the request is processed by the server until its final destination). The server is in critical phase if the QoS is low. The probability of the RT is above the bound $\alpha$.

The available servers are partitioned between the customers so that $\cup_{j=1}^{J} I_j = I$. There are also control actions available to allocate arriving sessions from customer $j$ class $k$ to a given machine $i$, and also to set the fraction of machine $i$ assigned to the work in progress. Let us call the first control rate $u^1_{ijk}$ and the second rate $u^2_{ijk}$. It is assumed that controls are applied randomly to the arrival streams. We have

$$\sum_{i \in I_j} u^1_{ijk} = \lambda_k^{(j)} \qquad \text{for } k = 1, \ldots, K \quad j = 1, \ldots, J$$

and

$$\sum_{k} u^2_{ijk} = \mu_{ik}^{(j)} \qquad \text{for } i \in I_j \quad j = 1, \ldots, J$$

Following the analysis in [4], one may estimate the tail probability for response time for each class $k$ customer $j$ and machine $i$ as follows:

$$P[\mathrm{RT}_{ik}^{(j)} > z_{jk}] \leq e^{-[u^2_{ijk} - u^1_{ijk}] z_{jk}} \tag{5.1}$$

To bound the response time it is sufficient to bound the estimate. For planning purposes we will assume that the QoS is violated on a given machine $i$ when the estimate is larger than $\alpha_{jk}$.

In short, we assume the following general framework

1. In each interval of time, the number of new requests considered for service is variable and distributed according to a Poisson process with parameter $\lambda_k^{(j)}$.
2. The assignment of rates occurs at the starting point of each time unit.
3. Unserved requests have no impact on the system.
4. It is possible to forecast the demands $\lambda_k^{(j)}$ and the service rates $\mu_{i,k}^{(j)}$ so that one can allocate processors in time to satisfy the QoS.

## 5.4 Server Allocation With Delay (SAWD)

At times, it will be necessary to reallocate servers to compensate for predicted abnormal situations.

In order to do this, we will have to consider the revenue implications of such a move. By a family of servers we mean a set of servers sharing the load of one Web site. When the number of requests for a Web site causes the probability of a large response time to the customer, we say the family of servers *is going red* or enters a *critical phase*. See figure above.

To complicate matters, it is not possible to reallocate a server instantaneously. This is due to security issues. To reallocate a server, we must first let its active threads die out. Only then can it be reallocated to a new customer (this usually takes about 5 minutes).

Finding an optimal solution through dynamic programming is an extremely difficult task due to the long-time horizon in this problem (24 hours), and the short intervals on which decisions are made. This leads to a problem of such large magnitude that a solution is impractical. Instead, various threshold algorithms can be used to get good solutions. We give an example of such a scheme below.

We will make our decisions based on three important values, namely

- the probability of a server family going red, $P$
- the expected cost rate incurred from going red and not meeting the SLA, denoted $C$
- the expected revenue rate for providing service beyond the customer's required level (detailed in their SLA), denoted $R$ (this value may be zero for customers not willing to bid, see Section 2).

Note that $C$ and $R$ are both non-negative values and cannot be both zero at the same time for a particular family of servers. This is because $C$ is non-zero when we have gone red as a result of not providing the resources required in the SLA, whereas $R$ is non-zero when we have gone red as a result of traffic being so high that the level of resources agreed to in the SLA is insufficient.

As mentioned above, it takes about 5 minutes for a server being moved to come on-line in its new family. However, the server does not immediately stop contributing to its original family. We have approximated that it continues to work for approximately 1/3 of a 5-minute interval (indicating the time period in which it is still handling active threads), after which it is removed from its family. So, for 2/3 of a 5-minute interval, it is not active in any family. This reflects the period in which it is shutting down and being rebooted.

We will introduce subscripts to reflect when a parameter is measured. A subscript of 1 indicates the parameter is measured 5 minutes from now, a subscript of 2 indicates 10 minutes from now, etc. We will introduce a superscript of +1 or -1 to our parameter $P$ to indicate the probability of going red given the addition or subtraction of a server from the family, respectively, i.e. $P_1^{-1}$ indicates the probability of going red 5 minutes from now given that we have removed a server from the family.

For each family of servers, we have created the following measures:

$$Need = P_1 \cdot C_1 + P_2 \cdot C_2 + (1 - P_1^{+1})R_1 + (1 - P_2^{+1})R_2 \tag{5.2}$$

Note that due to the mutually non-zero relationship of $C$ and $R$ mentioned above, either the first two terms above are zero, or the second two terms are zero. If the first two terms are zero, this indicates that a traffic level higher than agreed to in the SLA would push us into red, and if the last two terms are zero, this indicates that we might fall into a penalty situation. Thus, *Need* can reflect either a possibility to make extra revenue (if action is taken), or the possibility of paying penalties (if action is not taken), depending on which terms are zero. The higher the *Need* of a family is, the more money that can be lost or earned by adding a server to that family.

$$Availability = \tag{5.3}$$

$$= [\frac{2}{3}P_{1/3}^{-1} \cdot C_{1/3} + P_1^{-1} \cdot C_1 + P_2^{-1} \cdot C_2 + \frac{2}{3}(1 - P_{1/3})R_{1/3} +$$

$$(1 - P_1)R_1 + (1 - P_2)R_2$$

*Availabiliy* is closely related to *Need*, but there are two significant differences. The first is that the superscripts reflect that we are considering removing a computer from the family, as opposed to adding one. The second difference is that there are two extra terms. These terms reflect the fact that the server will be removed from the family after 1/3 of the first 5 minute interval. *Availability* is intended to measure the amount of penalties that will be paid, or revenue lost if we move a server from that family. Hence, the smaller the *Availability* of a family is, the less money we are likely to lose from moving a server from that family.

Note that all terms in the above equations are non-negative, and for one particular family of servers, the *Availability* value will always exceed the *Need* value.

In order to decide when to take action and move a server from one family to another, we use the following heuristic:

1. Calculate the *Need* and *Availability* for every family of servers.

2. Compare the largest *Need* value with the smallest *Availability* value. If the *Need* value exceeds the *Availability* value, one server is taken from the family corresponding to the *Availability* value and given to the family corresponding to the *Need* value.

3. If a server was told to move, go back to step 1 (note: the probabilities will change as the number of servers used to make the calculations will be different). Terminate the loop if no server was told to move in the last iteration.

The above iteration loop should be performed on a frequent basis. We suggest about every 15 seconds.

This is only one possible heuristic, and we have yet to actually compare it in simulation with an optimal solution. However, it has the obvious advantage of requiring considerably less computation than a long-time horizon dynamic program, which allows it to be performed very often. This allows us to react nearly instantaneously to a predicted critical situation. The $P$, $C$ and $R$ values are obtained from forecasts provided from the router control level.

## 5.5 Future Work

We would like to extend the single server single queue dynamics to a system which encompasses the initiation of the SAWD. Quantifying the QoS was the challenging task during the allotted time; thus we must decide which measurement of the QoS best fits the model. In consequence, expressing the measure of QoS is imperative in the formulation of the stochastic optimal control problem for maximizing the total expected revenue. Finding a fair price is another goal. In addition, we wish to perform simulations to test the heuristics and investigate the distribution of the real file size data.

# Bibliography

[1] Bertsekas, D., Dynamic Programming and Stochastic Control, New York: Academic Press, 1976.

[2] Chang, Y-C., Guo, X., Kimbrel, T. and King, A., *Optimal allocation policies for Web hosting*, IBM T.J. Watson Research Centre, P.O. Box 704, N.Y. 10598.

[3] Chatwin, R. E., *Continuous time airline overbooking with time dependent fares and refunds*, Applied Decision Analysis LLC, Transportation Science, **33**(2), (1999), 182-199.

[4] Liu, Z. et al., *On maximizing service level agreement profits*, IBM T.J. Watson Research Center, P.O.Box 704, NY 10598.

[5] Nortel and Bay Networks, *IP QoS-A bold new network*, September 1998, Nortel Marketing Publications, Dept. 4262, P.O. Box 13010, Research Triangle Park, NC 27709.

[6] Paschalidis, I. Ch. and Tsitsiklis, J.N., *Congestion-depending pricing of network services*, Technical Report, October 1998, Dept. of Manufacturing Engineering, Boston University, Boston MA 02215.

[7] Subramaanian, J., Stidham, S. and Lautenbacher, C.J., *Airline yield management with overbooking, cancellation, and no-shows*, Transportation Science **33**(2), (1999), 147-167.

# Chapter 6

# Monte Carlo Simulation in the Integrated Market and Credit Portfolio Model

John Chadam[8], Joel Hanson[2],
Yuriy Kazmerchuk[3], Selly Kane[4], Alex Kreinin[5], Viktoria Hsu[6], Eric Machorro[7], Jack Macki[4],
Hassan Masum[9], Ramin Mahammadalikhan[1], Barkha Saxena[10], LiMei Sun[11], Rafael Meza[6],
Report prepared by Selly Kane, Viktoria Krupp and Jack Macki

## 6.1 Introduction

Credit granting institutions deal with large portfolios of assets. These assets represent credit granted to obligors as well as investments in securities. A common size for such a portfolio lies from anywhere between 400 to 10,000 instruments.

The essential goal of the credit institution is to minimize their losses due to default. By default we mean any event causing an asset to stop producing income. This can be the closure of a stock as well as the inability of an obligor to pay their debt, or even an obligor's decision to pay out all his debt.

Minimizing the combined losses of a credit portfolio is not a deterministic problem with one clean solution. The large number of factors influencing each obligor, different market sectors, their interactions and trends, etc. are more commonly dealt with in terms of statistical measures.

---

[1] University of Toronto
[2] University of California Berkeley
[3] York University
[4] University of Alberta
[5] Algorithmics Inc.
[6] University of Washington
[7] Portland State University
[8] University of Pittsburgh
[9] Carleton University
[10] University of California Santa Barbara
[11] Memorial University of Newfoundland

Such include the expectation of return and the volatility of each asset associated with a given time horizon.

In this sense, we consider in the following the expected loss and risk associated with the assets in a credit portfolio over a given time horizon of (typically) 10 to 30 years. We use a Monte Carlo approach to simulate the loss of a portfolio in multiple scenarios, which leads to a distribution function for the expected loss of the portfolio over that time horizon. Second, we compare the results of the simulation to a Gaussian approximation obtained via the Lindeberg-Feller Theorem. Consistent with our expectations, the Gaussian approximation compares well with a Monte Carlo simulation in case of a portfolio of very risky assets.

Using a model which produces a distribution of expected losses allows credit institutions to estimate their maximum expected loss with a certain confidence interval. This in turn helps in making important decisions about whether to grant credit to an obligor, to exercise options or otherwise take advantage of sophisticated securities to minimize losses. Ultimately, this leads to the process of credit risk management.

## 6.2 The Problem

Estimation of the risk involved in large portfolios of securities posing various individual credit risks is a problem which can be studied using Monte Carlo methods. The main difficulties include

- the large number of different risk factors (interest rates, foreign exchange rates, ...)
- statistical dependencies between market risk factors and probabilities of default.

There are several variance reduction techniques (importance sampling, stratified sampling, ...) which are applicable to many practical problems in finance, in particular to the pricing of sophisticated securities. The problem we face is how to utilize these techniques for portfolio risk analysis.

In general, the problem can be considered in both one-time-step and multi-time-step settings.

The most interesting practical case corresponds to non-risky credit portfolios. In this case the portfolio losses depend on default events that are relatively rare. Therefore, efficient Monte Carlo simulation could be based on a transformation of the measure describing the joint evolution of market and credit risk factors.

A framework for credit risk estimation that has been used in industry is based on a joint market credit risk model described in Idcoe, Kreinin and Rosen [11].

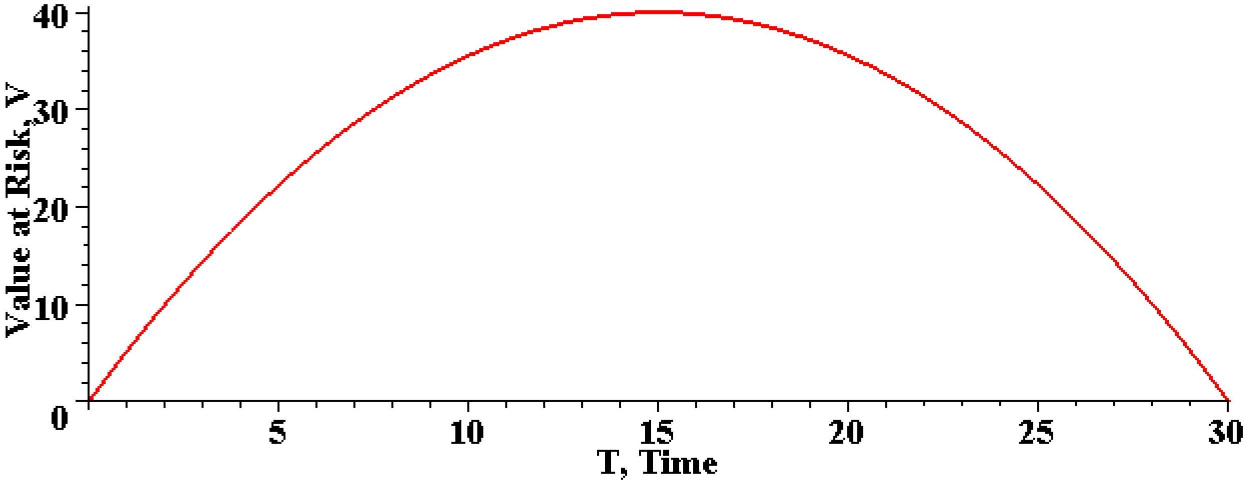
## 6.3 The Model

We consider a portfolio of individual investments, from which we swap bonds/stocks/derivatives, borrow money, or lend to obligors. Each item (counter party) in our portfolio of investments has a risk of default in form of a probability associated with it. Various individual credit risks are determined by, for example, Moody's ratings and like instruments, plus our own evaluations.

There can be anywhere from 400 to 10,000 items in a portfolio, from 8 to 200 credit classes which form a partition of the items in the portfolio, and several (independent) risk factors.

Default, again, means the event of an obligor deciding to not pay, or to completely pay out a debt, since in either case our income stops. Stocks, bonds and derivatives also represent obligors in the sense that if the value of a stock associated with an obligor drops to zero, then the obligor has decided not to pay us a return on our investment. In case of default, each item has a corresponding expected loss called the "value at risk" or "exposure", $V$, of that item.

Figure 1 shows a possible time plot of the exposure of one asset over a 30-year time interval in millions of dollars. The maximum exposure is different for each counter party (obligor/asset), and the time course shown here is typical for a bond swap, which is most common in a credit portfolio.



Unlike most currently used Portfolio Credit Risk (PCR) models, we will assume that market risk factors (interest rates, foreign exchange rates. etc.) are stochastic rather than deterministic. This level of generality is not of particular importance for portfolios of loans and floating rate instruments, but is of great importance for derivatives such as swaps and options.

Our main assumption is that conditional on a market, all defaults and rating changes are independent. The state of our model at any time is a complete specification of the relevant economic and financial credit drivers and market factors (macroeconomic, microeconomic, financial, industrial, etc.) that drive the model.

The actual loss experienced in case of default at time $t$ is described by a random variable, $L = V(t)B$, where $V(t)$ is the exposure at time $t$, and $B$ is Bernoulli distributed with mean $p$, the probability of default. The idea is to find the probability of default, $p$, and obtain the expected loss at time $t$, $E(L) = pV(t)$.

The framework of our model can be broken down into three parts: Risk Factors and Scenarios, the Joint Default Model, and the Modeling of Obligor Exposures and Recoveries within a Scenario. The Joint Default Model in turn has its own three components: the definitions of Unconditional Default Probabilities, of Credit Worthiness Indices, and the construction of a model which links each obligor's credit worthiness index to the credit drivers. We discuss these parts in detail below.

### 6.3.1 Risk Factors and Scenarios

With t measured in years, consider the single period $[t_0, t]$ where $t = t_0 + 1$. In this single time step, a scenario corresponds to one possible state of the world at time t. More precisely, a scenario is identified with a list of K systemic factors which are one possible realization of the corresponding *credit drivers*. The credit drivers in turn are the stochastic quantities which underlie each scenario, and which directly influence the credit worthiness of each obligor in the portfolio. It is common to consider anywhere from 100 to 10,000 possible scenarios.

Let $x(t)$ denote the vector of logarithms of relative risk factors at time $t$, i.e.,

$$x_k(t) = \ln\left[\frac{r_k(t)}{r_k(t_0)}\right],$$

where $r_k(t)$ denotes the value of the k$^{th}$ risk factor at time $t$. We assume that at the time horizon the $x_k(t)$ are normally distributed: $\mathbf{x}(t) \sim N(\mu, \mathbf{Q})$, where $\mu$ is a vector of mean returns and $\mathbf{Q}$ is a covariance matrix. Denote by $\mathbf{Z}(t)$ the vector after normalization after its components, $Z_k(t) = (x_k(t) - \mu_k)\,\sigma_k$.

In a single time step, each risk factor $Z_k(t) \sim N(0, 1)$. However, the risk factors are correlated according to $\mathbf{Z}(t) \sim \mathbf{N}(\mathbf{0}, \mathbf{Q})$ with a correlation matrix, $\mathbf{Q}$, which has ones on its diagonal. More precisely, for some empirically obtained number $\rho < 1$

$$\mathbf{Q} = \begin{pmatrix} 1 & \rho & \rho^2 & \cdots & \rho^K \\ \rho & 1 & \rho & \ddots & \vdots \\ \rho^2 & \rho & 1 & \ddots & \rho^2 \\ \vdots & \ddots & \ddots & \ddots & \rho \\ \rho^K & \cdots & \rho^2 & \rho & 1 \end{pmatrix}$$

This correlation structure can be created by starting with a vector of standard normal distributed elements, $\eta \sim \mathbf{N}(\mathbf{0}, \mathbf{I})$, where $\mathbf{I}$ denotes the K-dimensional identity matrix. For some matrix, $\mathbf{A}$, $\mathbf{A}\eta \sim \mathbf{N}(\mathbf{0}, \mathbf{A}\mathbf{A}^\top)$. From a Cholesky factorization for positive definite matrices we obtain $\mathbf{Q} = \mathbf{R}^\top\mathbf{R}$. Therefore, define $\mathbf{A} = \mathbf{R}^\top$ and obtain the correlated risk factors

$$\mathbf{Z}(t) = \mathbf{A}\eta \sim \mathbf{N}(\mathbf{0}, \mathbf{Q})$$

For multiple time steps, the default risk accumulates over time. Therefore we model the risk factors as starting with $\mathbf{Z}(0) = \mathbf{0}$, and evolving according to

$$\mathbf{Z}(t) = \mathbf{Z}(t-1) + \mathbf{z}$$

where $\mathbf{z}$ is determined by evaluating $\mathbf{Z}(t)$ over a single time step.

### 6.3.2 The Joint Default Model

The joint default model consists of three components: First, the definition of unconditional default probabilities; second, a multi-factor model of the credit worthiness index for each counter party based upon unconditional default probabilities and credit drivers; and third, an estimate for the conditional default probability of each counter party according to the multi-factor model.

### Unconditional Default Probabilities

Let $\tau_j$ denote the time of default by an obligor in sector (credit class) j, and let $p_j(t)$ denote this obligor's *unconditional probability of default*, i.e. the probability of default by an obligor in sector j by time t.

$$p_j(t) = Pr\{\tau_j \leq t\}.$$

We assume that all obligors in sector j have the same unconditional probability of default. We assume that the unconditional probabilities for each sector are available from an internal model or from an external agency. In particular,

$$p_j(t) = 1 - \exp(-\lambda_j t)$$

where the parameters $\lambda$ are specific to the various credit classes. We use $\lambda_{AAA}^{-1} = 150$ years, $\lambda_D^{-1} = 3$ years, and distribute the values for the remaining six classes linearly in between these values. Here, according to Moody's, AAA is the most reliable credit class, and D is the class of most risky assets.

### Credit Worthiness Index

The *credit worthiness index*, $Y_j(t)$, of obligor j, for j=1,...,N, determines the credit worthiness or financial health of counter party j at time t. By considering the value of its index, it can be determined whether an obligor is in default. We define the credit worthiness index by assuming that $Y_j(t)$, a continuous random variable, is related to the credit drivers through a linear multi-factor model as follows. Recall that the number of risk factors is K, therefore

$$Y_j(t) = \sum_{k=1}^{K} \beta_{c(j)k} Z_k(t) + \sigma_{c(j)} \epsilon_j, \tag{cwi}$$

where

$$\sigma_{c(j)} = \left[ 1 - \sum_{k=1}^{K} \beta_{c(j)k}^2 \right]^{\frac{1}{2}}.$$

is the volatility of the idiosyncratic component associated with the credit class, c(j), of obligor j, and $\epsilon_j$, $j = 1, 2, \ldots, N$ are i.i.d. standard normal variables representing random events affecting obligor j. The coefficients $\beta_{c(j)k}$ correspond to the sensitivity of the index of an obligor in credit class c(j) to the k$^{th}$ risk factor. Therefore, the first term on the right in (1) is the systemic component of the index, while the second term is the idiosyncratic component, specific to each counter party. Note that the distribution of the index is standard normal; it has zero mean and unit variance, which will later allow us to obtain our Gaussian approximation.

Since all obligors in a sector are statistically identical, obligors in a given sector share the same multi-factor model. However, while all obligors in a sector, c, share the same $\beta'_{ck}$s and $\sigma_c$, each has its own idiosyncratic uncorrelated component $\epsilon_j$.

For implementation purposes we link each credit class, c, to exactly one credit driver, $Z_{k(c)}(t)$, and obtain an index for each obligor j in credit class c:

$$Y_j(t) = \beta_c Z_{k(c)}(t) + \sigma_c \epsilon_j, \qquad \text{where } \sigma_c = \left[1 - \beta_c^2\right]^{\frac{1}{2}} \tag{6.1}$$

Values of $\sigma_c^2$ are chosen as 0.25, 0.35, 0.55, and 0.8 for the drivers corresponding to the least risky to most risky counter parties, respectively.

The notion of ”multi-factor” usually indicates that the credit worthiness index is some non-trivial combination of multiple, independent risk factors. Here, we associate each of our eight Moody type credit classes with one of four risk factors. Therefore, in our framework multi-factor is to be understood in the sense that each driver already is defined as a combination of a set of independent drivers via the correlation matrix. Therefore, it represents the influence of multiple, independent drivers, even though we explicitly include only one risk factor in each credit worthiness index.

### Conditional Default Probabilities

The *conditional probability of default* of an obligor in sector j, $p_j(t, \mathbf{Z})$, is the probability that an obligor in sector j defaults at time t, conditional on scenario $\mathbf{Z}$:

$$p_j(t, \mathbf{Z}) = Pr\left\{\tau_j \leq t \mid \mathbf{Z}\right\} \tag{6.2}$$

To estimate these conditional probabilities we will need a conditional default model which describes the functional relationship between the credit worthiness index $Y_j(t)$ (and hence the systemic factors) and the default probabilities $p_j(t, \mathbf{Z})$.

We assume that default is driven by a Merton model [14], i.e., default occurs when the assets of the firm fall below a given boundary or threshold, generally given by its liabilities. In our model the obligor defaults when its credit worthiness index, $Y_j$, falls below $\alpha_j$, the *unconditional default threshold*.

In these terms, the unconditional default probability of obligor j is given by

$$p_j = Pr\left\{\tau_j \leq t\right\} = Pr\left\{Y_j < \alpha_j\right\} = \Phi(\alpha_j), \tag{6.3}$$

where $\Phi$ denotes the normal cumulative density function (for simplicity we have dropped the time from the notation). Thus the unconditional threshold $\alpha_j$ is obtained from the inverse of equation 6.3. In particular, for a single time step:

$$\alpha_j = \Phi^{-1}(p_j) = \Phi^{-1}(1 - \exp(-\lambda_j t)). \tag{6.4}$$

The conditional probability of default on the other hand is the probability for the credit worthiness index to fall below its threshold in a given scenario. For convenience, we again drop time from the notation:

$$p_j(\mathbf{Z}) = Pr\left\{Y_j < \alpha_j \,|\, \mathbf{Z}\right\} =$$

$$= Pr\left\{\sum_{k=1}^{q^c} \beta_{jk}\mathbf{Z}_k + \sigma_j\epsilon_j < \alpha_j \,|\, \mathbf{Z}\right\} = Pr\left\{\epsilon_j < \frac{\alpha_j - \sum_{k=1}^{q^c} \beta_{jk}\mathbf{Z}_k}{\sigma_j}\right\} =$$

$$= \Phi\left(\frac{\alpha_j - \sum_{k=1}^{q^c} \beta_{jk}\mathbf{Z}_k}{\sigma_j}\right) = \Phi(\hat{\alpha}_j\left(\mathbf{Z}\right)).$$

The *conditional threshold*, $\hat{\alpha}(\mathbf{Z})$, is the threshold below which the idiosyncratic component of obligor j, $\epsilon_j$, must fall for default to occur in a given scenario, $\mathbf{Z}$.

Note that the obligor credit worthiness index correlations are uniquely determined by the default model and the multi-factor model, which links the index to the credit drivers. The correlations between individual obligor defaults are then obtained from the functional relationship between the index and the event of default, as determined by the Merton model. For example, the indices of obligors that belong to the same sector are perfectly correlated if their idiosyncratic component is zero.

### 6.3.3 Obligor Exposures and Recoveries in a Scenario

We define the exposure to an obligor j at time t as the amount that will be lost due to outstanding transactions with that obligor if default occurs, unadjusted for future recoveries, and we denote it by $V_j(t)$. An important property of PCR-SD is the assumption that exposure is deterministic, not scenario dependent, i.e. $V_j(t)$ is not a function of $\mathbf{Z}$.

The actual loss experienced in case of default of counter party j at time $t$ is described by a random variable,

$$L_j(\mathbf{Z}) = V_j \mathbf{1}$$

where $V_j$ is the exposure at time $t$, and $\mathbf{1}$ is Bernoulli distributed with mean $p_j(\mathbf{Z})$, the probability of default at time $t$. The idea is to find the probabilities of default of each counter party and obtain the expected cumulative loss at time $t$,

$$\mathcal{E}L_j(\mathbf{Z}) = \sum_{j=1}^{N} V_j(t) \cdot p_j(\mathbf{Z}).$$

With adjustment for further recovery, the economic loss for a default by obligor j is

$$L_j(\mathbf{Z}) = V_j \cdot (1 - \gamma_j), \tag{6.5}$$

where $\gamma_j$ is the recovery rate, expressed as a fraction of the obligor exposure. Recovery in the event of default is assumed deterministic. Expressing the recovery amount as a fraction of the exposure value at default does not necessarily imply instantaneous recovery of that fraction of the exposure at the time of default.

The distribution of conditional losses for each obligor is given by

$$L_j(\mathbf{Z}) = \begin{cases} V_j \cdot (1 - \gamma_j) & \text{with probability } p_j(\mathbf{Z}); \\ 0 & \text{with probability } 1 - p_j(\mathbf{Z}). \end{cases} \tag{6.6}$$

or in short,

$$L_j(\mathbf{Z}) = V_j \cdot (1 - \gamma_j) \cdot \mathbf{1}$$

Our expected loss for a given scenario is given by the sum of the expected losses of each of the obligors:

$$\mathcal{E}L(\mathbf{Z}) = \sum_{j=1}^{N} V_j \cdot (1 - \gamma_j) \cdot p_j(\mathbf{Z}). \tag{6.7}$$

## 6.4 Investigation of the Model

In the remainder of this article we will use the following index system: j=1,...,M scenarios, k=1,...,K risk factors, and i=1,...,N default probabilities:

| Scenario | Risk Factor | Default Probabilities | Exposures |
|---|---|---|---|
| 1 | $R_1^1, \ldots, R_K^1$ | $P_1^1, \quad P_2^1 \ldots, P_N^1$ | $V_1^1, \ldots, V_N^1$ |
| $\vdots$ | $\vdots$ | $\vdots$ | $\vdots$ |
| $M$ | $R_1^M, \ldots, R_K^M$ | $P_1^M,\ P_2^M \ldots, P_N^M$ | $V_1^M, \ldots, V_N^M$ |

Let the loss in scenario j be

$$\mathcal{L}_j = \sum_{i=1}^{N} V_i^j \mathbf{1}\{\tau_i^j < \hat{t}\ \}(1 - \gamma_{ji}).$$

This sum has a huge number of terms. Note that

$$\mathcal{E}(\mathbf{1}\{\tau_i^j < \hat{t}\}) = P_i^j.$$

Our key idea is to approximate the distribution.

**Phase 1.** We use one time interval, [0, $\tau$], and recovery rates are constant ($\gamma_{ij}$).

**Phase 2:** $1 - \gamma_{ij}$ is random, we still use one time step.

$$Pr\{\gamma_{ij} < t\} = \frac{\int_0^t (1-u)^{\alpha_{ij}} u^{\beta_{ij}}\, du}{\mathcal{B}(\alpha_{ij}, \beta_{ij})}.$$

**Phase 3.** Multiple time steps.

Assume that $\gamma_{ij}$ is independent of $V_{ij}$, at least in the early scenarios. Assume that we know the distribution of the $\tau_{ij}$, that is, $Pr\{\tau_{ij} < t\}$ is known for *all* scenarios and all counterparties j = 1, . . . , N, i = 1, . . . , k.

We use discrete time and a finite number of steps.

Now think of $R_i^j = R_i^j(t)$ generating $P_i^j(t)$ and $V_i^j(t)$.

How do we develop a distribution for portfolio losses = $\mathcal{L}$ ? Since you cannot default twice, we need to pull defaults out of $\mathcal{L}$ at each time step:

$$\mathcal{L}(t) = \sum_{k=1}^{t} \mathcal{L}(k).$$

We try

$$\mathcal{L}_j(t) = \sum_{i=1}^{k} V_i^j(t) \mathbf{1}(t - 1 < \tau_i^j \leq t).$$

Here we have embedded the $1 - \gamma_{ij}(t)$ into the $V_i^j(t)$. Notice that

$$Pr\{t-1 < \tau_i^j \leq t\} = P_i^j(t) - P_i^j(t-1).$$

We know the random variables $\tau_i^j$. We would like to use some form of the central limit theorem on this sum.

We assume $\mathcal{L}(t)$ takes on the values $\mathcal{L}_1(t), \dots, \mathcal{L}_N(t)$ with equal probabilities 1/N (uniform distribution).

Typically, the maximum time is in the range 30 to 50.

The problem: How to efficiently estimate $\mathcal{L}_j(t)$, j = 1, …, N. We want the distribution of these random variables…moments are not enough. We can write

$$\mathcal{L}_j(T) = \sum_{i=1}^{k} \sum_{t=1}^{T} V_i^j(t) \mathbf{1}\{t-1 < \tau_i^j \leq t\}.$$

The inner sum in t is denoted $X_i^j(T)$ and represents counterparty loss. These random variables are conditionally independent, given the $j^{th}$ scenario.

Now k is large, say 200-300, so we can use the central limit theorem for triangular arrays, commonly referred to as the Lindeberg- Feller Theorem (see [9]). In order to apply this theorem, we must first center and standardize each of the random variables. Hence we need the mean and variance of the $X_i^j(T)$.

We have:

$$X_i^j(t) = \sum_{t=1}^{T} V_i^j(t) \mathbf{1}\{t-1 < \tau_i^j \leq t\},$$

$$\mu_i^j(T) = \mathcal{E} X_i^j(T) = \sum_{i=1}^{T} V_i^j(t)[P_i^j(t) - P_i^j(t-1)].$$

Now to the variance:

$$Var[X_i^j(T) = Var\left[\sum_{t=1}^{T} V_i^j(t)\mathbf{1}\{t-1 < \tau_i^j \leq t\}\right] =$$

$$= \sum_{i=1}^{T} [V_i^j(t)]^2 Var\{\mathbf{1}\{t-1 < \tau_i^j \leq t\}\} -$$

$$- \sum_{t=1}^{T} \sum_{\substack{t'=1 \\ t \neq t'}}^{T} V_i^j(t) V_i^j(t') Covar[\mathbf{1}\{t-1 < \tau_i^j \leq t\}\mathbf{1}\{t'-1 < \tau_i^j \leq t'\}].$$

Now

$$Covar[\mathbf{1}\{t-1 < \tau_i^j \leq t\}\mathbf{1}\{t'-1 < \tau_i^j \leq t'\}] =$$

$$= \mathcal{E}[\mathbf{1}\{t-1 < \tau_i^j \leq t\}\mathbf{1}\{t'-1 < \tau_i^j \leq t'\}] - \mathcal{E}[\mathbf{1}\{t-1 < \tau_i^j \leq t\}]\mathcal{E}[\mathbf{1}\{t'-1 < \tau_i^j \leq t'\}]$$

The expected value of $[\mathbf{1}\{t-1 < \tau_i^j \leq t\}]^2$ is

$$\int [\mathbf{1}\{t-1 < \tau_i^j \leq t\}]^2 \, dP = P_i^j(t) - P_i^j(t-1).$$

while the expected value squared of $\mathbf{1}\{t-1 < \tau_i^j \leq t\}$ is

$$\left[\int \mathbf{1}\{t-1 < \tau_i^j \leq t\} \, dP\right]^2 = [P_i^j(t) - P_i^j(t-1)]^2.$$

Hence

$$Var[\mathbf{1}\{t-1 < \tau_i^j \leq t\}] = P_i^j(t) - P_i^j(t-1) - (P_i^j(t) - P_i^j(t-1))^2.$$

Next we need to calculate

$$\mathcal{E}[\mathbf{1}\{t-1 < \tau_i^j \leq t\}\mathbf{1}\{t'-1 < \tau_i^j \leq t'\}] = \int \mathbf{1}\{t-1 < \tau_i^j \leq t\}\mathbf{1}\{t'-1 < \tau_i^j \leq t'\} \, dP.$$

but $t$ and $t'$ are natural numbers, so for $t \neq t'$ we have $(t-1, t] \cap (t'-1, t'] = \emptyset$, so

$$\mathcal{E}[\mathbf{1}\{t-1 < \tau_i^j \leq t\}\mathbf{1}\{t'-1 < \tau_i^j \leq t'\}] = \int \emptyset \, dP = 0.$$

Finally, we calculate

$$\mathcal{E}[\mathbf{1}\{t-1 < \tau_i^j \leq t\}]\mathcal{E}[\mathbf{1}\{t'-1 < \tau_i^j \leq t'\}] = [P_i^j(t) - P_i^j(t-1)][P_i^j(t') - P_i^j(t'-1)].$$

Putting all of the above together, we have

$$Covar[\mathbf{1}\{t-1 < \tau_i^j \leq t\}\mathbf{1}\{t'-1 < \tau_i^j \leq t'\} = [P_i^j(t) - P_i^j(t-1)][P_i^j(t') - P_i^j(t'-1)].$$

Thus

$$[\sigma_i^j(t)]^2 = Var[X_i^j(T) = Var\left[\sum_{t=1}^{T} V_i^j(t)\mathbf{1}\{t-1 < \tau_i^j \leq t\}\right] =$$

$$= \sum_{i=1}^{T} [V_i^j(t)]^2 Var\{\mathbf{1}\{t-1 < \tau_i^j \leq t\}\} -$$

$$- \sum_{t=1}^{T} \sum_{\substack{t'=1 \\ t \neq t'}}^{T} V_i^j(t) V_i^j(t') Covar[\mathbf{1}\{t-1 < \tau_i^j \leq t\}\mathbf{1}\{t'-1 < \tau_i^j \leq t'\}] =$$

$$= \sum_{t=1}^{T} ([V_i^j(t)]^2 [P_i^j(t) - P_i^j(t-1)][1 - (P_i^j(t) - P_i^j(t-1))] -$$

$$-[P_i^j(t) - P_i^j(t-1)] \sum_{\substack{t'=1 \\ t \neq t'}}^{T} V_i^j(t) V_i^j(t') [P_i^j(t') - P_i^j(t'-1)]). \tag{6.8}$$

Let $\mu_i^j(T) = \mathcal{E}X_i^j(T) = \sum_{t=1}^{T} V_i^j(t)[P_i^j(t) - P_i^j(t-1),]$ and let $[\sigma_i^j(t)]^2$ be as above, then define

$$S_k(T) = Var\left(\sum_{i=1}^{k} X_i^j(T)\right) = \sum_{i=1}^{k}\sum_{t=1}^{T}\left[\sigma_i^j(t)\right]^2.$$

Now if we define

$$(*) \qquad Y_{i,k}^j(T) = \frac{X_i^j(T) - \mu_i^j(T)}{\sqrt{S_k(T)}},$$

then we study

$$\sum_{i=1}^{k} Y_{i,k}^j(T).$$

It is easy to see that $\sum_{i=1}^{k}\mathcal{E}\left(Y_{i,k}^j(T)\right)^2 = 1$, which is the first requirement of the Lindeberg-Feller Theorem. The second requirement is that $\forall\epsilon > 0$,

$$\lim_{k\to\infty}\sum_{i=1}^{k}\mathcal{E}\left(|Y_{i,k}^j(T)|^2\mathbf{1}(|Y_{i,k}^j(T)| > \epsilon)\right) \to 0.$$

If we assume that the $V_i^j$ are bounded, i.e. $\sup_{i,t\le T}|V_i^j(t)| \le M^j(T) < \infty$, and that $S_k \to \infty$ as $k \to \infty$, then it is obvious that given $\epsilon > 0$ we have $|Y_{i,k}^j(T)| < \epsilon \;\; \forall i \le k$ for k sufficiently large. Hence the second condition of the Lindeberg-Feller Theorem is easily satisfied, under these very reasonable assumptions. Since the $V_i^j(t)$ represent the loss from counter party i at time t, we are just assuming that our maximum loss up to time T is finite.

Similarly, if we assume the probability of default in a given time interval, $P_i^j(t) - P_i^j(t-1)$ . is strictly between 0 and 1 and that $V_i^j(t) > 0$ for $1 \le t < T$, then we can argue as follows to see that $S_k \to \infty$ as $k \to \infty$. First, note that $S_k(1) < S_k(2) < \cdots < S_k(T)$ since $\sigma_i^j(t) > 0 \;\; \forall t$. Hence it is enough to consider $S_k(1)$, which is

$$S_k(1) = \sum_{i=1}^{k}\left[\sigma_i^j(1)\right]^2 =$$

$$= \sum_{i=1}^{k}\left[V_i^j(1)\right]^2\{P_i^j(1) - P_i^j(0)\}\{1 - P_i^j(1) + P_i^j(0)\} \ge$$

$$\ge \sum_{i=1}^{k}[V^j]^2 P^j \quad \to \infty \quad \text{as} \quad k \to \infty,,$$

where $V^j = \inf_i V_i^j(1) > 0$, and $P^j = \inf_i\left[P_i^j(1)(1 - P_i^j(1))\right] > 0$ since both infs are over finite sets of time greater than zero. Note that the double sum in (8) disappears when $T = 1$, since the sum is over the empty set.

So $\mathcal{L}_j(t)$ is approximated by a sum of Gaussian random variables. It is a *mixture* of Gaussian random variables and we know the mean and the variance. So to find the quantile for $\mathcal{L}$ we can use the Central Limit Theorem conditionally, i.e. to find $Pr\left(\mathcal{L} \le \beta_p\right) = p$ we first condition on the scenario

$$Pr\left(\mathcal{L} \le \beta_p\right) = \sum_{j=1}^{N} Pr\left(\mathcal{L} \le \beta_p \mid \text{Scenario } j\right) Pr\left(\text{Scenario } j\right) =$$

$$= \sum_{j=1}^{N} Pr\left(\mathcal{L} \le \beta_p\right) \frac{1}{N} \approx \sum_{j=1}^{N} \Phi_{\mu_j, \sigma_j}(\beta_p) \frac{1}{N}.$$

Now, for a given p, we can solve numerically for $\beta_p$. Since this is an asymptotic result, the only question left is whether k is large enough. This leads to simulation exercises.

### 6.4.1 Estimating and Simulating Default Probabilities

We have

$$Y(t) = \alpha R(t) + \beta \epsilon(t), \alpha^2 + \beta^2 = 1,$$

with $\epsilon$ and $R$ independent. R is the credit driver and indicator of industrial quality, while $\epsilon$ is the individual counterpart. If $\tau$ is the time of default, then

$$Pr\{\tau = 1\} = Pr\{Y(1) \le H_1\},$$

$$Pr\{\tau = 2\} = Pr\{Y(1) > H_1, Y(2) \le H_2\},$$

and so on, where the numbers $H(k)$ represent the boundary of the nondefault region.

$$R(t) = \sum_{1}^{t} [R(j) - R(j-1)],$$

and the differences inside the sum are $N(0, \hat{\sigma})$.

Each counterparty has its own $\alpha$ and $\beta$; Y is like an *index*, so we normalize. $\alpha$ can be assumed to lie in the interval [0.25, 0.8]. Y(t) is in some sense a probability of default:

$$Pr\{Y(1) < H_1 \mid R(1)\} = Pr\{\epsilon(1) < \frac{H_1 - \alpha R(1)}{\beta}\} =$$

$$= \Phi\left(\frac{H_1 - \alpha R(1)}{\beta}\right)$$

### 6.4.2 Finding the Unconditional Default Distribution

Define $q_k = Pr\{\tau = k\}, \quad k = 1, 2, \ldots$. These we can determine. We know Y(t) is normal and $Pr\{Y(1) \le H_1\} = q_1$, so $H_1 = \Phi^{-1}(q_1)$,

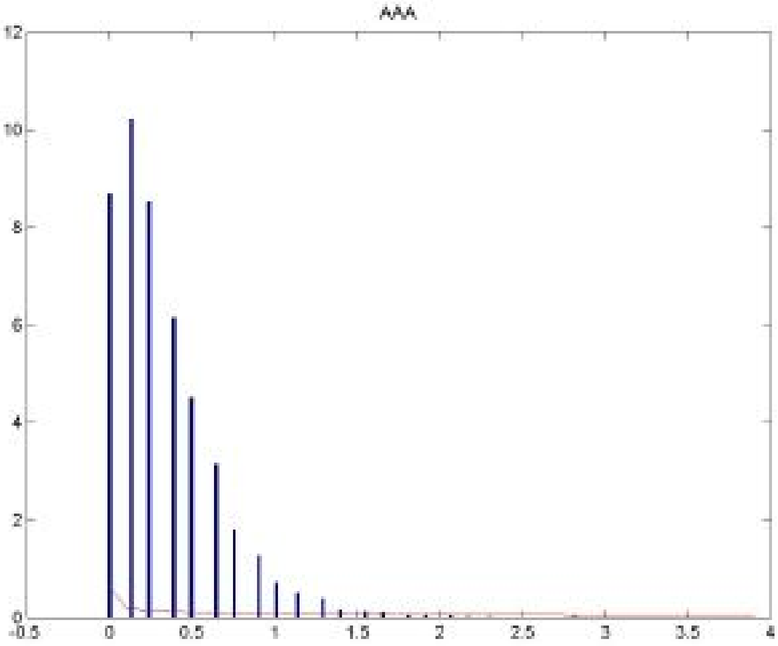
$$Pr\{Y(1) > H_1,\ Y(2) < H_2\} = q_2, \quad Y(2) = Y(1) + \Delta Y_2,$$

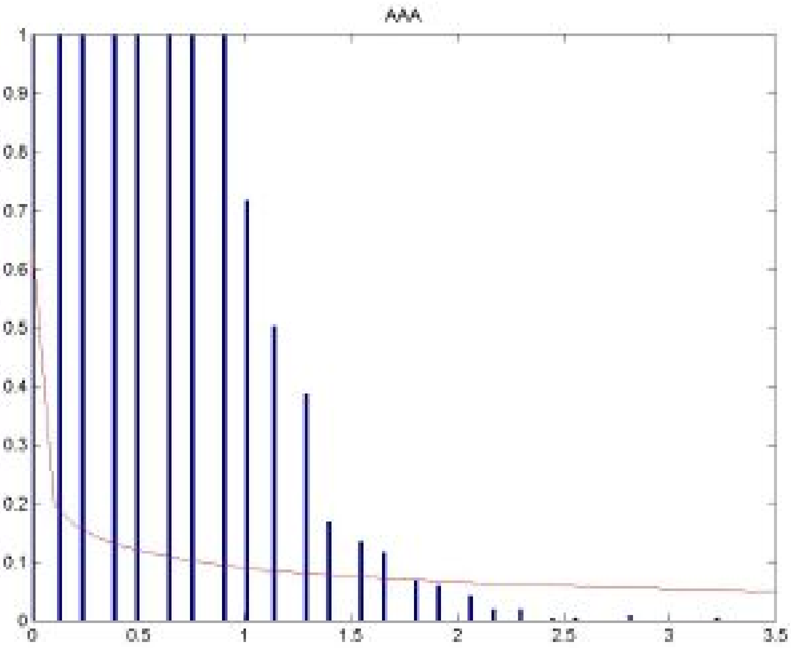
and in principle we can carry out all the necessary calculations.

## 6.5 Comparison with Simulations

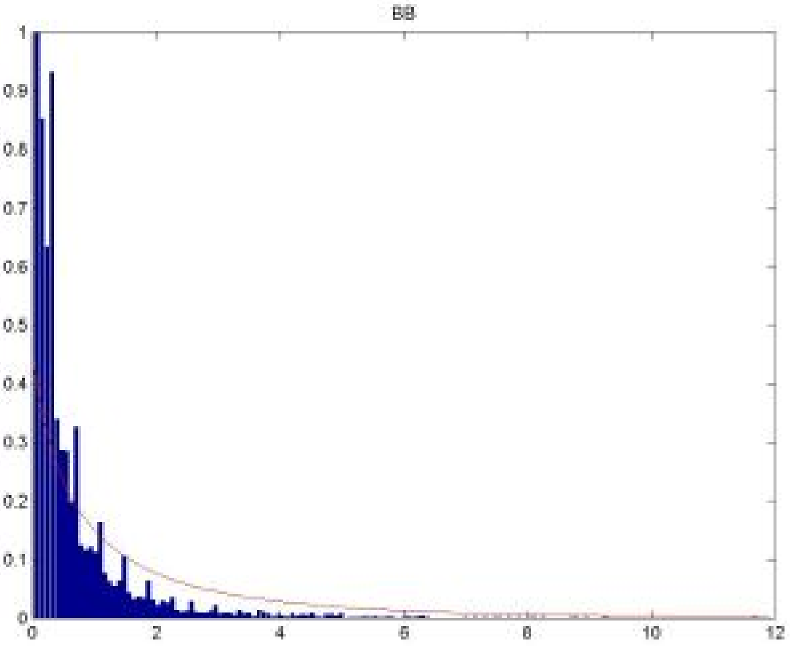
An alternative to the procedure described above is to use Monte Carlo methods, which in general are much more time-consuming. The effectiveness of our new approach can be judged by running Monte Carlo simulations and comparing with our predictions . In the short time we had, we only were able to run a few simulations, but the results were very promising.

Here are the results of some runs with AAA instruments. The result of the Monte Carlo Simulation is the vertical bars, our new method gives the curve. The figure on the right is a magnification of part of the figure on the left, notice that the best fit is in the "upper tail" :

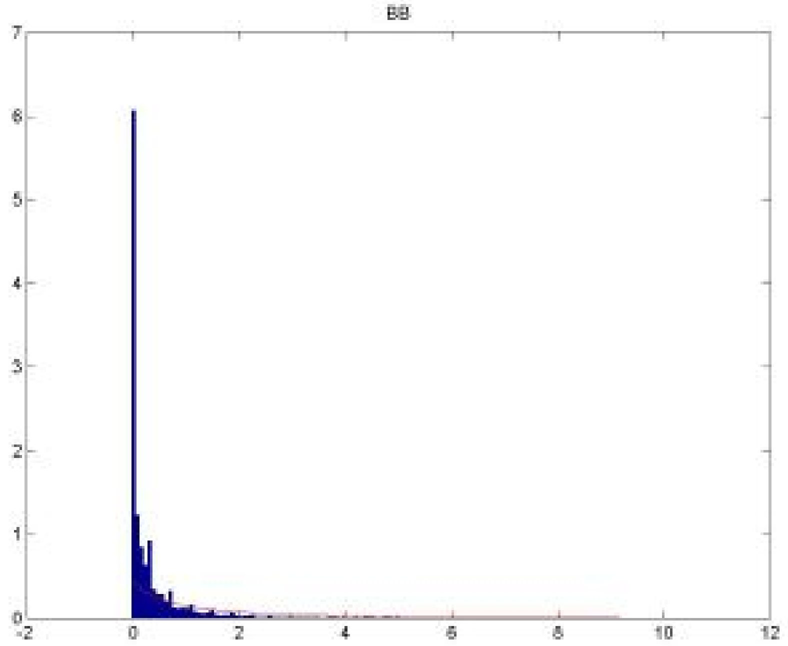




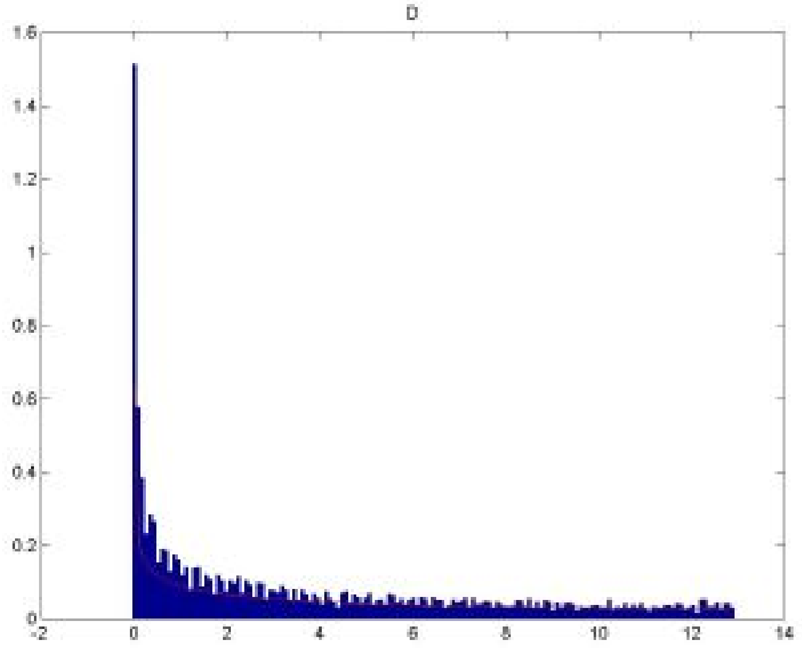
Here is the result of a simulation with mixed AAA and BB:



Our model is better with BB instruments:



and with D instruments:



## 6.6 Conclusions

By making a clever application of the Lindeberg-Feller Theorem, we were able to develop an analytic method for modelling portfolio losses. The method works well with simulations in the important upper tail of the distributions.

Future work would look at faster algorithms for huge portfolios and try to understand where the Gaussian approximation does and does not work.

# Bibliography

[1] Aziz, J. and Charupat, N., *Calculating credit exposure and credit loss: a case study*, Algo Research Quarterly, **1**(1), (1998), 31-46.

[2] Basle Committee on Banking Supervision, International Convergence of Capital Measurements and Capital Standards, July, 1998.

[3] Basle Committee on Banking Supervision, Credit Risk Modelling: Current Practices and Applications, Basle, April, 1999.

[4] Basle Committee on Banking Supervision, A New Capital Adequacy Framework, Consultative Paper, Basle, June, 1999.

[5] J.P. Morgan Inc., Credit Metrics: The Benchmark for Understanding Credit Risk, Technical Document, New York, 1997.

[6] Credite Suisse Financial Products, Creditrisk+: A Credit Risk Management Framework, New York, 1997.

[7] Crouhy, M. and Mark, R., *A comparative analysis of current credit risk models*, paper presented at the conference Credit Modeling and Regulatory Implications, London, September, 1998.

[8] Finger, C., *Conditional approaches for CreditMetrics portfolio distributions*, Credit Metrics Monitor, April 1999, 14-33.

[9] Fristedt, B. and Gray, L., A Modern Approach to Probability Theory, Chapter 17, Springer, New York, 1996.

[10] Gordy, M. A Comparative Anatomy of Credit Risk Models, Federal Reserve Board, Finance and Economics Discussion Series, No. 47, 1998.

[11] Idcoe, I., Kreinen, A. and Rosen, D., *An integrated market and credit risk portfolio model*, Algo Research Quarterly, **2**(3), pp. 21-37.

[12] Kealhofer, S., *Managing default risk in portfolios of derivatives*, in the book Derivative Credit Risk, Risk Publications, London, 1996.

[13] Koyuoglu, H.U. and Hickman, A., *Reconcilable differences*, Risk, **11**(10), (1998), 56-62.

[14] Merton, R.C., *On the pricing of corporate debt: the risk structure of interest rates*, J. Finance, **29** (1974), 449-470.

[15] Nagpal, K.M. and Bahar, R., *An analytical approach for credit risk analysis under correlated defaults*, Credit Metrics Monitor, April 1999, 51-74.

[16] Wilson, T., *Portfolio credit risk I*, Risk, **10**(9), (1997), 111-117 .

[17] Wilson, T., *Portfolio credit risk II*, Risk, **10**(10), (1997), 56-61.

# Appendix A

# List of Participants

| | | |
|---|---|---|
| Rita Aggarwala | Univ. of Calgary | rita@math.ucalgary.ca |
| Tom Alberts | Univeristy of Alberta | |
| Amal Amleh | Univ. Rhode Island | amleh@math.uri.edu |
| Peter Anderson | University of Victoria | peter@math.uvic.ca |
| Angus Argyle | University of Victoria | argyle@math.uvic.ca |
| Brian Alspach | University of Regina | alspach@math.uregina.ca |
| Mehmet Bergen | University of British Columbia | mabegen@coe.ubc.ca |
| Sean Bohun | Penn State University | csb15@psu.edu |
| Chris Bose | University of Victoria | cbose@math.uvic.ca |
| Tark Bouhennache | Univ. California, Los Angeles | bouhenna@math.ucla.edu |
| Ali Ghodsi Boushehri | University of Waterloo | aghodsib@math.uwaterloo.ca |
| Thomas Brakel | University of British Columbia | brakel@mech.ubc.ca |
| Mariana Carrasco-Teja | University of Washington | mariana@amath.washington.edu |
| Monica Cojocaru | Queen's University | monica@mast.queensu.ca |
| Brian Corbett | University of Manitoba | umcorbe1@cc.umanitoba.ca |
| Tom Cottrell | University of Calgary | cottrell@ucalgary.ca |
| Ellis Cumberbatch | Univ. of California | Ellis.Cumberbatch@cgu.edu |
| Andrea Doeschl | University of Western Ontario | abdoesch@julian.uwo.ca |
| Gregory Dresden | Univ. of Washington & Lee | dresdeng@wlu.edu |
| Adrian Driga | University of Alberta | adrian@cs.ualberta.ca |
| Aude Espesset | University of Western Ontario | aespesse@uwo.ca |
| Leslie Fairbairn | Simon Fraser University | |
| Ellen Fowler | University of British Columbia | effowler@smartt.com |
| Yashar Ganjali Gavgani | University of Waterloo | yganjali@math.uwaterloo.ca |
| Mohammad Ghufrnan | | |
| Jacquie Gregory | University of Victoria | |
| Mehdi Hadj Karim Kharazi | University of British Columbia | kharrazi@civi.ubc.ca |
| Joel Hanson | Univ. of California, Berkeley | jhanson@stat.berkeley.edu |
| John Harlim | University ofGuelph | jharlim@uoguelph.ca |
| Sheelah Heckroodt | University of British Columbia | sheelah@coe.ubc.ca |
| Joel Ho | Univ. Washington, Seattle | jho@u.washington.edu |

| | | |
|---|---|---|
| Huaxiong Huang | York University | hhuang@yorku.ca |
| Kristen Jaskie | Univ. of Washington, Seattle | kjaskie@hotmail.com |
| Jihyoun Jeon | Univ. of Washington, Seattle | jhjeon@amath.washington.edu |
| Selly Kane | University of Alberta | skane@stat.ualberta.ca |
| Qutaibeh Katatbeh | Concordia University | qutaibeh@cicma.concordia.ca |
| Mehdi Kharrazi | University of British Columbia | |
| Yuriy Kazmerchuk | York University | yuriy_kazmerch@hotmail.com |
| Andrew King | University of Victoria | |
| Lawence Kolasa | Ryerson University | lkolasa@acs.ryerson.ca |
| Theodore Kolokolnikov | University of British Columbia | tkolokol@math.ubc.ca |
| Denis Kosygin | Princeton University | kosygin@math.northwestern.edu |
| Nathan Krislock | University of British Columbia | krislock@math.ubc.ca |
| Viktoria Krupp | Univ. of Washington, Seattle | krupp@amath.washington.edu |
| Judy Lai | Univ. Texas at Austin | judylai@math.utexas.edu |
| Aihua Li | Univ. New Orleans | ali@loyno.edu |
| Lee Namyong | Minnesota State University | leen1@krypton.mnsu.edu |
| Melvin Leok | California Inst. of Technology | mleok@cds.caltech.edu |
| Randy LeVeque | Univ. Washington, Seattle | rjl@amath.washington.edu |
| Jackie Li | Dalhousie University | |
| Margarett Liang | | |
| Shuqing Liang | York University | sqliang@yorku.ca |
| Eric Machorro | Portland State University | ericmachorro@yahoo.com |
| Mufeed Mahhoud | University Western Ontario | mahmomm@engga.uwo.ca |
| Jack Macki | University of Alberta | jmacki@gpu.srv.ualberta.ca |
| Allen Majdanac | Simon Fraser University | |
| Tatiana Marquez Lago | Simon Fraser University | ttm@sfu.ca |
| Ramin Mohammadalikhani | University of Toronto | |
| Andrew Markiel | Univ. Washington, Seattle | markiel@astro.washington.edu |
| Hassam Masum | Carleton University | hmasum@ccs.carleton.ca |
| Rafael Meza | Univ. of Washington, Seattle | rafaelm@amath.washington.edu |
| Matthias Mueck | University of Toronto | mueck@math.toronto.edu |
| Carmeliza Navasca | Univ. California, Davis | navasca@math.ucdavis.edu |
| Alberto Nettel | University of Calgary | nettel@math.ucalgary.ca |
| Nancy Ann Neudauer | Pacific Lutheran Univ. | neudauna@plu.edu |
| Eva-Marie Nosal | University of British Columbia | eva-marie@math.ubc.ca |
| Asa Packer | Univ. of Washington, Seattle | packer@math.washington.edu |
| Marc Paulhus | University of Calgary | paulhusm@math.ucalgary.ca |
| Christina,Popescu | University of Alberta | popescu@vega.math.ualberta.ca |
| Randall Pyke | Ryerson University | rpyke@ryerson.ca |
| Ali Rasekh | University of British Columbia | rasekh@interchange.ubc.ca |
| Moshe Rosenfeld | Univ. Washington, Tacoma | moishe@u.washington.edu |
| Elisabeth Rosenthal | Univ. Washington, Seattle | erosen@u.washington.edu |
| Tiina Roose | Harvard Medical | tiina@steele.mgh.harvard.edu |
| James Rossmanith | Univ. of Washington, Seattle | jrossman@amath.washington.edu |

| | | |
|---|---|---|
| Daniel Ryan | University of British Columbia | dpryan@math.ubc.ca |
| Ali Sanaie-Fard | University of British Columbia | alis@civil.ubc.ca |
| Barkha Saxena | Univ. California, Santa Barbara | barkha@pstat.ucsb.edu |
| Jeon Seungwon | Univ. Texas at Austin | swjeon@math.utexas.edu |
| Jason Slemons | Univ. Washington, Seattle | slemons@math.washington.edu |
| John Stockie | University of New Brunswick | stockie@unb.ca |
| Sarah Sumner | University of Ottawa | ssumner@mathstat.uottawa.ca |
| Li Mei Sun | Memorial University | limei@math.mun.ca |
| Tataria Toro | Univ Washington, Seattle | toro@math.washington.edu |
| Eric Varley | Lehigh University | EV01@LEHIGH.EDU |
| Rex Westbrook | University of Calgary | westbroo@ucalgary.ca |
| J.F. Williams | Bath University | mapjfw@maths.bath.ac.uk |
| Sukjin Yoon | Univ. Washington, Seattle | sjyoon@u.washington.edu |
| Ling Yi | University of Alberta | |
| Jill Zarestky | Univ. Texas at Austin | jillz@ticam.utexas.edu |
| Ling Zhao | University of Alberta | zhao@cs.ualberta.ca |

# PIMS Contact Information

email: pims@pims.math.ca

http://www.pims.math.ca

Director's Office
Pacific Institute for the Mathematical Sciences
Room 200, 1933 West Mall
University of British Columbia
Vancouver BC  V6T 1Z2
Canada