Proceedings of the Ninth

PIMS Industrial Problem Solving Workshop

PIMS IPSW 9

Co-sponsored by:

Informatics Circle of Research Excellence,

Alberta Innovation and Science,

The University of Calgary,

and

Natural Sciences and Engineering Research Council of Canada

Editor: C. Sean Bohun, Pennsylvania State University

Proceedings of the Ninth Annual
PIMS
Industrial Problem Solving Workshop

Editor: C. Sean Bohun, Pennsylvania State University

Co-sponsored by:
Informatics Circle of Research Excellence,
Alberta Innovation and Science,
The University of Calgary,
Natural Sciences and Engineering Research Council of Canada

CORE

Alberta
INNOVATION AND SCIENCE

MO SHUILE TOGAM SUAS
UNIVERSITY OF
CALGARY

NSERC
CRSNG

May, 2005

Foreword by the PIMS Director

The PIMS Industrial Problem Solving Workshops have been held annually since 1997. The 9th IPSW was held at the University of Calgary from May 15–19, 2005, immediately following the 8th GIMMC (Graduate Industrial Mathematics Modelling Camp) which was held at the University of Lethbridge. We are very grateful to the universities for hosting these events.

Approximately 55 participants worked intensely on five problems posed by industrial companies from across North America. All these problems highlight the need for correct mathematical modelling, and the industrial payback from a good solution. We are grateful to Donald Mackenzie (University of Calgary), Gerald K. Cole (Biomechanigg Research Inc.), Pierre Lemire & Rob Pinnegar (Calgary Scientific Inc.), Brad Bondy (Genus Capital Management), and Brian Russell (Hampson Russell Software) for providing the problems and guiding the participants.

Special thanks go to Sean Bohun from Penn State University who carefully edited these proceedings, and Elena Braverman and Gary Margrave from the University of Calgary for the difficult task of providing the industrial contacts and organizing the meeting.

Dr. Ivar Ekeland, Director
Pacific Institute for the Mathematical Sciences

Contents

List of Figures

Preface

The University of Calgary was the site of the ninth annual PIMS Industrial Problem Solving Workshop (IPSW). Hosted from May 15 through May 19, 2005, the workshop was co-sponsored by PIMS, Alberta Innovation and Science, iCORE and the University of Calgary. This workshop saw the bringing together of participants from across North America.

Many first time participants of these workshops are surprised by the intensity of the work. Based on the Oxford Study Group Model, the morning of the first day consists of the initial problem presentations and the beginning of focused discussions amongst the self-selected groups. Input from the academic experts was found to be especially vital during the initial phases the work, especially in providing guidance to the new participants and graduate students. Work continued throughout the next few days culminating in a summary presentation of results at the end of the fifth day. These summary presentations form the basis of these proceedings. Problems were chosen from a diverse set of topics which challenged the participants and utilized many of their individual talents. I would like to take this opportunity to thank the individual authors for their dedication and prompt response so that these proceedings were possible. These individuals were (in order of their submission):

- Tony Ware: Adaptive Statistical Evolution Tools for Equity Ranking Models
- Greg Lewis: Force-Control for the Automated Footwear Testing System
- Michael Lamoureux: Seismic Image Analysis Using Local Spectra
- Yaling Yin: Seismic Prediction of Reservoir Parameters
- C. Sean Bohun: Mathematical Model of the Mechanics and Dynamics of Tails in Dinosaurs

To ensure the smooth and efficient operation of the workshop, many individuals are needed behind the scenes. I would like to thank the organizing committee consisting of Elena Braverman and Gary Margrave both from the University of Calgary. Without their dedication and resourcefulness, this event would not have been possible.

Two other essential ingredients for the success of the workshop were the industrial representatives and the industrial experts. While the representatives are certainly an asset and provide a basis for each of the problems, it is the academic experts that are responsible for each of the groups moving along productive lines. For this year the academic experts were:

- C. Sean Bohun, Pennsylvania State University
- Len Bos, University of Calgary
- Elena Braverman, University of Calgary
- Gemai Chen, University of Calgary
- Lou Fishman, MDF International, University of Calgary
- Michael Lamoureux, University of Calgary
- Greg Lewis, University of Ontario Institute of Technology

- Qiao Sun, University of Calgary
- Tony Ware, University of Calgary
- Rex Westbrook, University of Calgary

A special thanks goes to the industrial contributors and their representatives who include:

- Donald M. Henderson, University of Calgary in collaboration with the Royal Tryell Museum of Palaeontology
- Gerald K. Cole, University of Calgary in collaboration with Biomechanigg Research Inc.
- Rob Pinnegar, Calgary Scientific Inc.
- Pierre Lemire, Calgary Scientific Inc.
- Brad Bondy, Genus Capital Management
- Brian Russell, Hampson-Russell Software

At the end of this monograph there is a listing of the workshop participants and the various institutions where they can be found and I would like to take this opportunity to apologize for any mistakes or omissions therein.

In closing, I would like to thank Heather Jenkins at the PIMS central office for her dedication, efficiency, and patience in dealing with the production of these proceedings.

C. Sean Bohun, Editor
Department of Mathematics
Pennsylvania State University

Chapter 1

Mathematical Model of the Mechanics and Dynamics of the Tails in Dinosaurs

Problem presented by: Donald Mackenzie (University of Calgary)

Mentors: C. Sean Bohun (Pennsylvania State University), Donald Henderson (University of Calgary), Bernard Monthubert (Institut de Mathématiques de Toulouse), Rex Westbrook (University of Calgary)

Student Participants: Robin Clysdale (University of Calgary), Diana David-Rus (Rutgers University), Matthew Emmett (University of Calgary), Chad Hogan (University of Calgary), Mark Hughes (University of Calgary), Enkeleida Lushi (Simon Fraser University), Peter Smith (Memorial University of Newfoundland), Naveen Vaidya (York University)

Report prepared by: C. Sean Bohun (csb15@psu.edu)

1.1 Introduction

Unlike mammals which have very reduced tails, the tails of dinosaurs represented a substantial fraction of their body lengths and masses. The left and right sides of the tail base in all dinosaurs acted as the anchor points for large, powerful muscles that attached on the rearward side of the hind limbs (Figure 1.1). These muscles pulled on the legs, causing them to rotate backwards and under the body, with the result that the animals were propelled forward. As well as pulling on the legs, these muscles would have exerted a reciprocal pull on the tail. During locomotion the left and right hind limbs would be alternately pulled, and be 180° out of phase with each other. These alternating tugs would have set up oscillations in the tail. It would seem that some sort of synchrony would have to arise between the rate at which the legs were swung back and forth and the natural frequencies of oscillations of the tail to allow efficient, stable walking and running. The extreme sizes of some dinosaurs—up to 30 tonnes in some cases—and the great range of body sizes—from a few hundred grams to many tonnes—gives dinosaurs the potential to be insightful models for the study of locomotory dynamics in terrestrial animals.

There are two possible avenues to investigate the effects of tails on locomotion:

1. Focus on just the ~ 14m tail of Diplodocus carnegii, a 24m sauropod where the tail represents approximately 26% of the total body mass detailed in Figure 1.2.

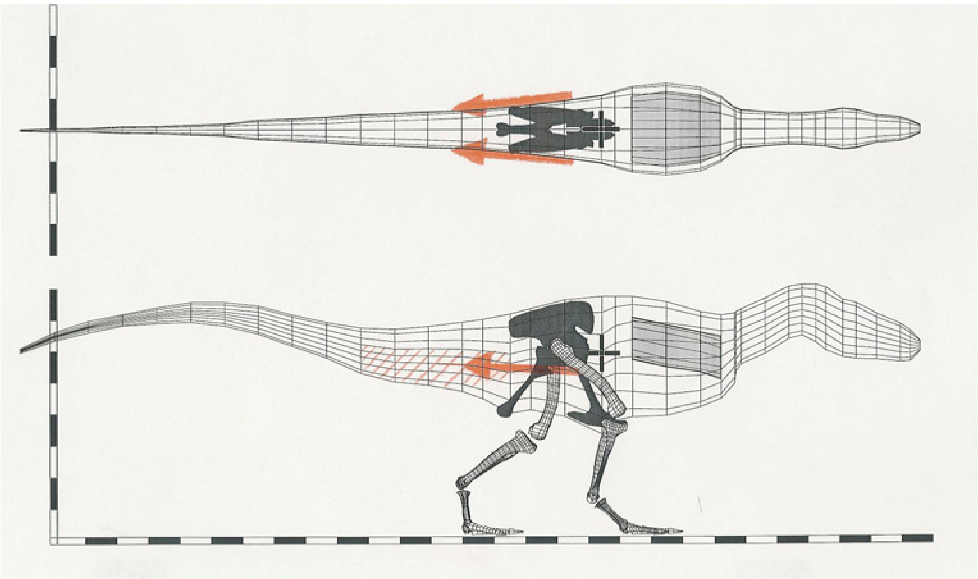


Figure 1.1: Anatomical details of the tail base structure in dinosaurs.

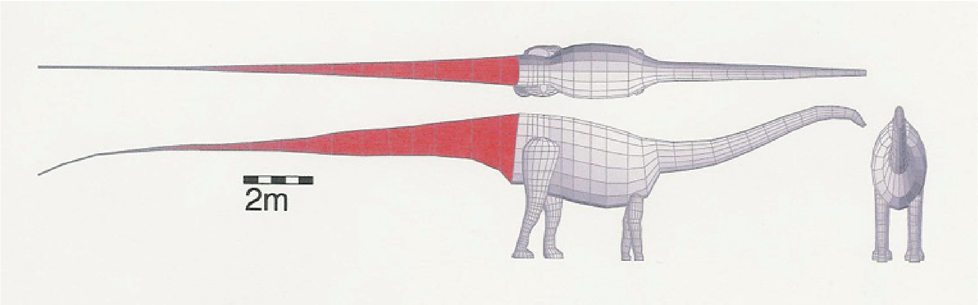


Figure 1.2: Diplodocus carnegii is a 24m sauropod with a 14m long tail.

2. Investigate the variation in tail mechanics that occurred during the evolution of theropod dinosaurs—the two-legged, carnivores. In small, early theropods such as the 30cm long Compsognathus the tail represents just over half the total body length, and is very slender and flexible. In larger, later theropods such as a 12m Tyrannosaurus the tail represents just one third of the total body length, and is proportionally deeper and much stiffer. Figure 1.3 illustrates this variation in tail structure.

Section 1.2 begins with a survey of physical data for both bipedal and quadrupedal dinosaurs. In this section the ratio of the tail to leg length is compared across many diverse species and a scaling law is developed that relates the tail length, leg length and tail radius. A continuous model for the tail is developed in Section 1.3 and by nondimensionalising a small parameter related to the thinness of the tail simplifies the resulting coupled nonlinear equations. It is shown that the resulting set of equations contain aspects of both beam dynamics and wave propagation. In Section 1.4 a discrete version of the tail is derived with the assumption that the sections of the tail are coupled with a stiff joint that allows

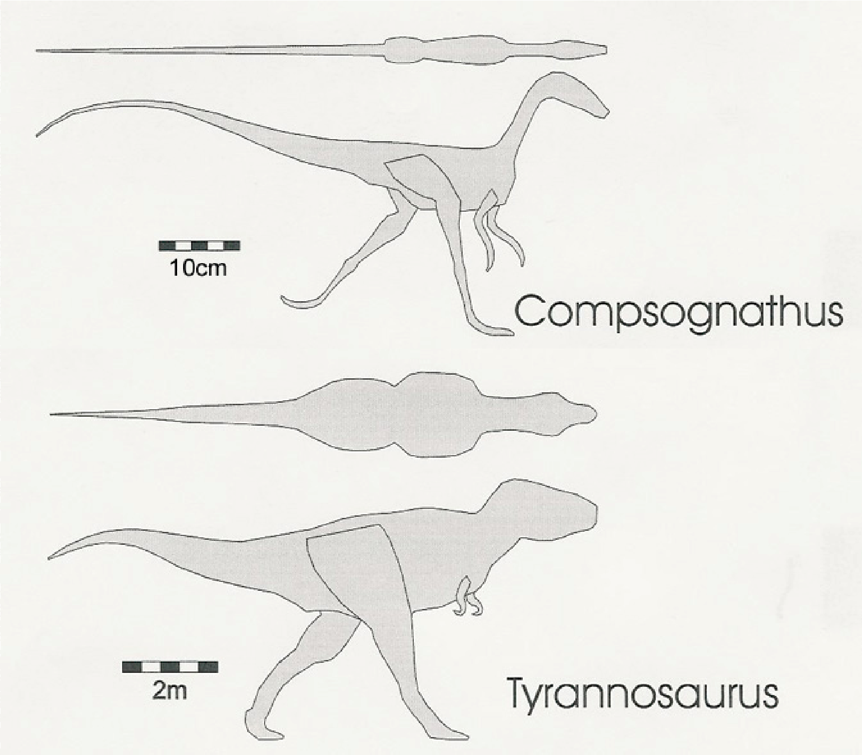


Figure 1.3: Comparison of the tail structures for Compsognathus and Tyrannosaurus.

rotation but does not allow extension. In this model the stiffness of each joint is characterized by an effective spring constant k_i for the ith joint and results in a discrete version of the Euler-Bernoulli expression for each of the tail segments. The paper finishes with some preliminary conclusions and directions for future work.

1.2 Dimensional Analysis

As a first model we suppose that the tail acts like a flexible beam with the periodic driving force of the rear legs modelled with a pendulum. If the beam has length L_{tail} and effective radius R_{tail} then the deflection of the beam $u(x,t) : [0, L_{\text{tail}}] \times [0, \infty) \to \mathbb{R}$ is given by

$$\frac{\partial^2}{\partial x^2}\left(EI(x)\frac{\partial^2 u}{\partial x^2}\right) = \rho A(x)\frac{\partial^2 u}{\partial t^2}$$

where E is the Young's modulus, I is the moment of inertia about the neutral axis, ρ is the density, and A is the cross section of the tail. If we nondimensionise by substituting

$$\begin{aligned} \hat{x} &= \frac{x}{L_{\text{tail}}}, & \hat{u} &= \frac{u}{L_{\text{tail}}}, & \hat{t} &= \frac{t}{T_{\text{tail}}}, \\ I &= R_{\text{tail}}^4 \hat{I}, & A &= R_{\text{tail}}^2 \hat{A} \end{aligned}$$

we find that

$$\frac{ER_{\text{tail}}^2 T_{\text{tail}}^2}{\rho L_{\text{tail}}^4} \frac{\partial^2}{\partial \hat{x}^2}\left(\hat{I}\frac{\partial^2 \hat{u}}{\partial \hat{x}^2}\right) = \hat{A}\frac{\partial^2 \hat{u}}{\partial \hat{t}^2}.$$

This indicates that the characteristic time to propagate a disturbance the complete length of the tail is

$$T_{\text{tail}} \sim \left(\frac{\rho}{E}\right)^{1/2} \frac{L_{\text{tail}}^2}{R_{\text{tail}}}.$$

At the same level of approximation assume that the legs of the dinosaur act like a pendulum of length L_{leg} so that the characteristic period for the motion of the legs is on the order of

$$T_{\text{leg}} \sim \left(\frac{L_{\text{leg}}}{g}\right)^{1/2}$$

where g is the acceleration due to gravity. As a result, if the tail plays a significant role in the locomotion with this model then $T_{\text{leg}} \sim T_{\text{tail}}$ and

$$\frac{L_{\text{tail}}^4}{L_{\text{leg}} R_{\text{tail}}^2} = \text{const.} \tag{1.1}$$

depending only on the composition of the tail. Notice that this expression predicts that for a fixed leg length, increasing the length of the tail necessarily increases its effective radius in contrast with the archaeological evidence.

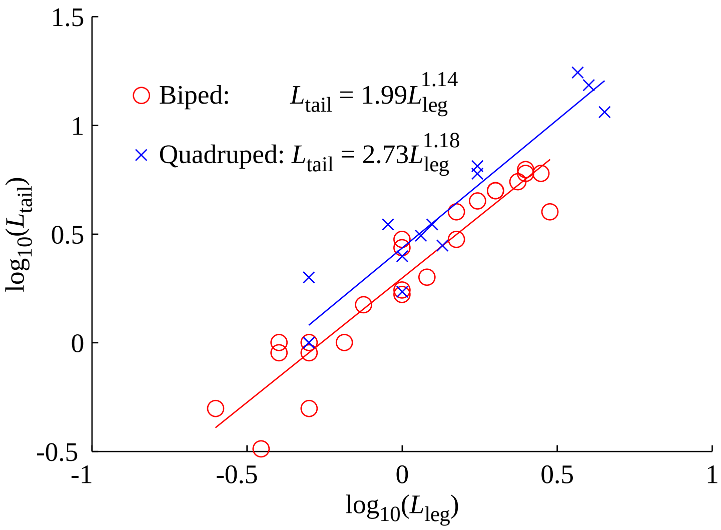


Figure 1.4: L_{tail} as a function of L_{leg} for a sample of bipedal and quadrupedal dinosaurs.

Figure 1.4 compares approximate values of the leg length to the tail length for a selection of bipedal and quadrupedal dinosaurs. For bipeds, the tail is typically twice the length of the leg whereas in quadrupeds the tail is, on average, an additional 40% longer. Figure 1.5 illustrates expression (1.1) and contrasts it with an optimal expression that minimizes the variation. Clearly the beam/pendulum model is not reflected in the sample but the variance is drastically reduced with the expressions

$$\text{Bipeds: } \frac{L_{\text{tail}}^{0.45} R_{\text{tail}}^{0.39}}{L_{\text{leg}}} = \text{const.} \qquad \text{Quadrupeds: } \frac{L_{\text{tail}}^{0.23} R_{\text{tail}}^{0.69}}{L_{\text{leg}}} = \text{const.} \tag{1.2}$$

These results imply that for a fixed leg length, if the length of the tail is doubled then the radius of the tail in a biped is halved, whereas in a quadruped the radius of the tail is reduced to one-eighth of its original value. So we see that bipeds tend to have much thicker tails than correspondingly sized quadrupeds.

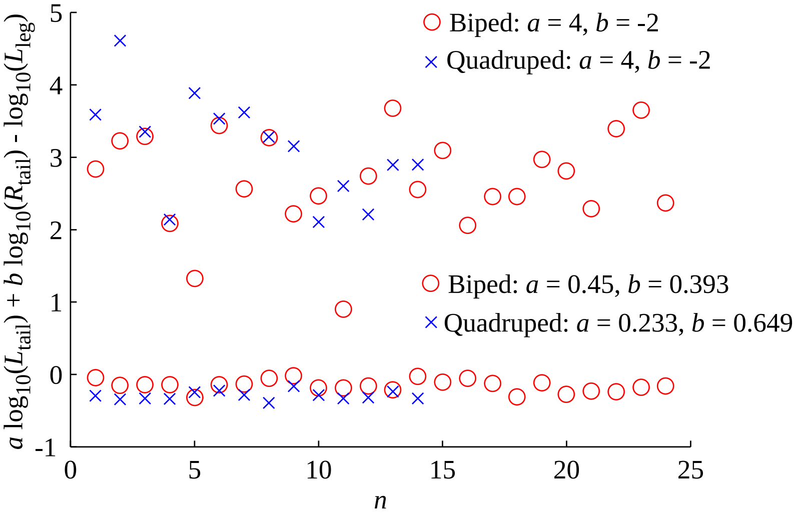


Figure 1.5: Scatter plot of the data assumed to satisfy (1.1) and the corresponding plot of the data when a and b are chosen to minimize the variance.

The question remains to find a mechanism that predicts the archaeological evidence described in (1.2). In the case of bipeds a similar expression can be recovered by analysing the simple cantilever depicted in Figure 1.6. Here the tail, rear legs, and forward torso are replaced with rectangular blocks. The condition that this effective dinosaur does not tip over is $L_{\text{leg}}^2(L_{\text{leg}}/2) - 2L_{\text{tail}}R_{\text{tail}}(L_{\text{tail}}/2) = 0$ or

$$\frac{L_{\text{tail}}^{2/3} R_{\text{tail}}^{1/3}}{L_{\text{leg}}} = \text{const.}$$

the point here is that scaling expressions like (1.2) are a result of balance equilibrium rather than synchronous locomotion. So it seems more likely that the physical dimensions of the tail are chosen to balance the dinosaur rather than complementing its locomotion dynamics.

1.3 Inextensible Rod Equations

We turn the discussion to the development of an appropriate model for the tail of the dinosaur and we begin with a derivation of the equations satisfied by an inextensible rod.

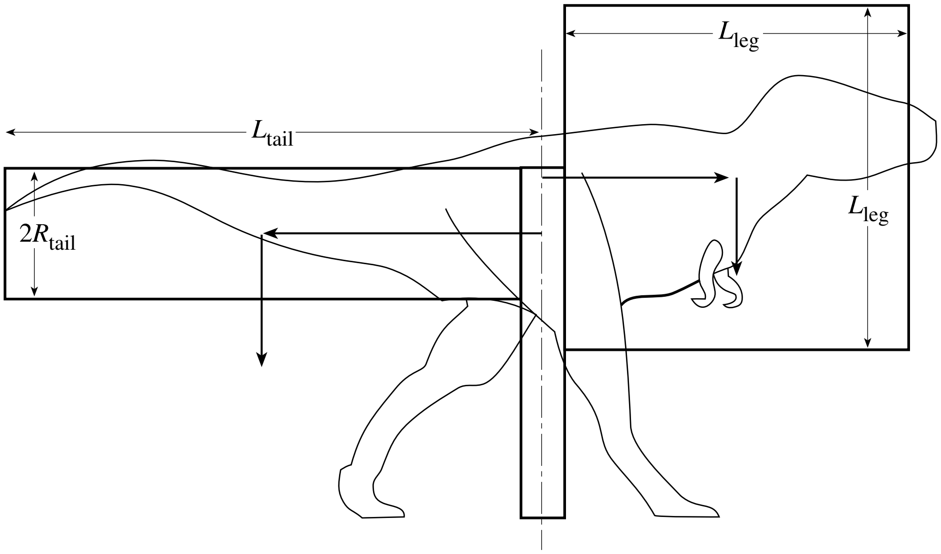


Figure 1.6: An effective dinosaur with the major anatomical structures replaced with blocks of a representative size.

To the left of Figure 1.7 is a segment of length Δs with orientation ϕ, internal tensions of F, G in the x, y directions, a bending moment M and an external force of $\vec{P}$ per unit length. In this case we are viewing the tail from above so that $\vec{P}$ is essentially zero since the weight of the tail acts perpendicular to this plane. It is simply left in for completeness. In any case, simple geometry gives the relationships

$$\cos\phi = \Delta x/\Delta s, \qquad \sin\phi = \Delta y/\Delta s. \tag{1.3}$$

If the density and cross sectional area of the segment are ρ and A respectively then the mass of the segment is $\rho A\Delta s$ and by resolving the linear motion in the x and y directions one obtains

$$(\rho A\Delta s)x_{tt} = \Delta F + P_1\Delta s, \qquad (\rho A\Delta s)y_{tt} = \Delta G + P_2\Delta s. \tag{1.4}$$

The angular motion is given by $I_0\phi_{tt} = \tau$ where I_0 is the moment of inertia of the cross section and τ is the net torque acting on the segment. From Figure 1.7, taking torques about the point A, one finds that $\tau = \Delta M - F\Delta y + G\Delta x$ so that

$$(\rho I\Delta s)\phi_{tt} = \Delta M - (F\Delta s)\sin\phi + (G\Delta s)\cos\phi \tag{1.5}$$

where we have used the moment of inertia of the cross section defined as

$$I = \iint_A y^2\, dA.$$

This quantity is analogous to the ordinary moment of inertia I_0 except that the mass element is replaced by the area element of the cross section. Note that the torque due to the external force is of a higher order of smallness.

A final relationship can be obtained by assuming that the amount of bending is small and that the material satisfies a linear constitutive relation $\sigma = E\epsilon$ relating the stress to the strain. Referring to the

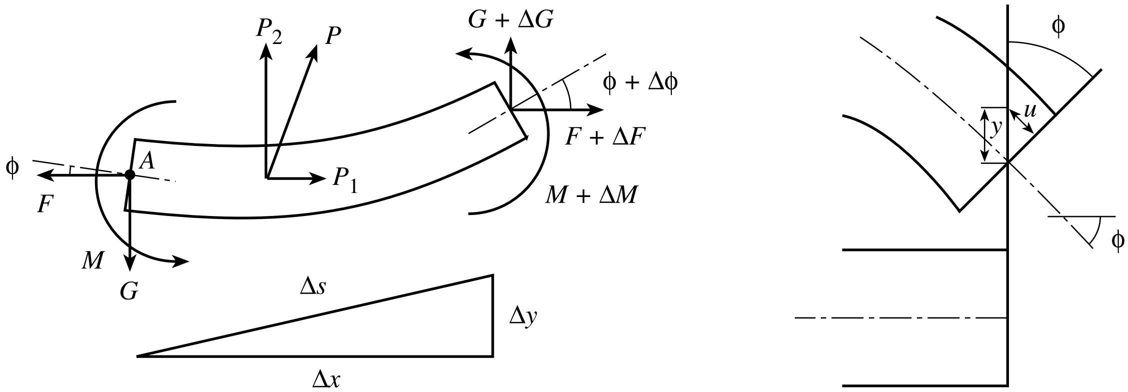


Figure 1.7: To the left a rod segment of length Δs experiences internal forces of F, G, a bending moment M and an external force per unit length of $\vec{P}$. To the right a portion of the rod is bent through an angle ϕ. If the normal to the cross section remains normal, the displacement $u = y\phi$.

right of Figure 1.7, a section of rod is bent through an angle ϕ. If this angle is sufficiently small then the normal to the cross section will remain normal after the bending distortion and the the displacement of the rod as a function of the distance from the neutral line is given by $u = y\phi$. Since the bending moment M for a given cross section is the sum of the moments about the neutral plane $y = 0$ these assumptions give the relationship

$$M = \iint_A y\sigma \, dA = E \iint_A y\epsilon \, dA = E\phi_s \iint_A y^2 \, dA = EI\phi_s. \tag{1.6}$$

In this expression σ is the stress in the rod, and $\epsilon = du/ds$ is the corresponding strain. Equation (1.6) is occasionally referred to as the Euler-Bernoulli assumption.

Letting the length of the segment Δs shrink to zero gives the final set of equations satisfied by an inextensible rod

$$\begin{aligned} x_s &= \cos\phi, && \text{(1.7a)} \\ y_s &= \sin\phi, && \text{(1.7b)} \\ \rho A x_{tt} &= F_s + P_1, && \text{(1.7c)} \\ \rho A y_{tt} &= G_s + P_2, && \text{(1.7d)} \\ \rho I \phi_{tt} &= M_s - F\sin\phi + G\cos\phi, && \text{(1.7e)} \\ M &= EI\phi_s. && \text{(1.7f)} \end{aligned}$$

To nondimensionalise we assume that there is a circular cross section so that the radius $R = R_0 r(s)$, $A = \pi R_0^2 r^2$, and $I = \pi R_0^4 r^4/4$. Scaling the lengths with the length of the tail L we let

$$\begin{aligned} \hat{x} &= \frac{x}{L}, & \hat{y} &= \frac{y}{L}, & \hat{s} &= \frac{s}{L}, & \hat{t} &= \frac{t}{T}, \\ \hat{F} &= \frac{F}{K}, & \hat{G} &= \frac{G}{K}, & \hat{M} &= \frac{M}{LK}, & \hat{P}_i &= \frac{P_i}{K} \end{aligned}$$

and identify three nondimensional quantities $C_1 = \pi\rho R_0^2 L^2/KT^2$, $C_2 = \pi\rho R_0^4/4KT^2$ and $C_3 = 4KL^2/\pi E R_0^4$. The characteristic magnitudes of the force and time should reflect the physical properties of the tail. Consider a rod of length L which is clamped horizontally at one end, free at the other, and bends under its own weight. If the rod has mass m and g is the gravitational constant then the shape satisfies

$$\zeta^{(iv)} = \frac{mg/L}{EI}, \qquad \zeta(0) = \zeta'(0) = \zeta''(L) = \zeta'''(L) = 0,$$

with solution

$$\zeta(s) = \frac{mg/L}{24EI} s^2(s^2 - 4Ls + 6L^2)$$

and a maximum displacement at $s = L$ that satisfies

$$\frac{\zeta(L)}{L} = \frac{1}{8}\,\frac{mg}{L^2/EI}.$$

In this case the characteristic force for a rod that bends under its own weight is $K = EI/L^2$ and choosing this value for K sets $C_3 = 1$. This leaves two natural choices for T. Either $T = L\sqrt{\rho/E}$ or $T = 2L^2\sqrt{\rho/E}/R_0$ in which $C_2 = 1$ or $C_1 = 1$ respectively. We choose the latter consequently

$$T^2 = \frac{4\rho\pi L^4}{ER_0^2} = \frac{\rho\pi R_0^2 L^4}{EI},$$

$C_1 = 1$, and $C_2 = (R_0/2L)^2$ is a small parameter for a long thin tail and is denoted as ϵ.

Dropping hats the nondimensional equations are

$$x_s = \cos\phi, \tag{1.8a}$$

$$y_s = \sin\phi, \tag{1.8b}$$

$$r^2(s)x_{tt} = F_s + P_1, \tag{1.8c}$$

$$r^2(s)y_{tt} = G_s + P_2, \tag{1.8d}$$

$$\epsilon r^4(s)\phi_{tt} = M_s - F\sin\phi + G\cos\phi, \tag{1.8e}$$

$$M = r^4(s)\phi_s \tag{1.8f}$$

with $0 \leq s \leq 1$, and $r(s) = R(s)/R_0$ a nondimensional radius of the rod. Since ϵ is small, equation (1.8e) implies that if initially $\tau(s) = M_s - F\sin\phi + G\cos\phi \neq 0$ then ϕ will change rapidly with time until $\tau = 0$. Conversely, equations (1.8c) and (1.8d) indicate that x and y will not appreciably change during this equalization process. On the time scale of T, the ϵ term can be omitted and expression (1.8e) can be replaced with

$$\tau(s) = M_s - F\sin\phi + G\cos\phi = 0.$$

1.3.1 Boundary Conditions and Initial Conditions

Since the tip of the tail $(s = 1)$ is free, the internal forces and bending moments vanish so that $F = G = M = 0$. At the base $(s = 0)$ it is not clear if one should consider a clamped, hinged, or simply supported condition. For a clamped base both the position and direction are specified fixing $x(t,0)$, $y(t,0)$, $x_s(t,0)$, and $y_s(t,0)$ for all t. If the base is hinged then the position is fixed but the bending moment M is zero. Finally, if the base is supported then it is free to slide and both the point of contact and the direction are unknown. In this final case $M = 0$ and the direction of vector $\langle F, G\rangle$ must be perpendicular to the rod.

By initially assuming that the tail is in equilibrium and that there is no external forces acting on the tail in the xy-plane $(P_1 = P_2 = 0)$ we find that $F_s = G_s = 0$ so that both $F(0,s) = F_0$ and $G(0,s) = G_0$ are constant. Since F and G are constant, the condition $\tau = 0$ can be integrated to give $M(0,s) = x(s)G_0 - y(s)F_0$ to avoid any fast dynamics.

1.3.2 Small Deflection Approximation

Suppose that the lateral deflection is small so that $\phi \simeq 0$, $P_1 = P_2 = 0$ and (1.8a)-(1.8f) become with $\epsilon = 0$

$$\begin{aligned} x_s &= 1, & y_s &= \phi, \\ r^2(s)x_{tt} &= F_s, & r^2(s)y_{tt} &= G_s, \\ M_s &= F\phi - G, & M &= r^4(s)\phi_s. \end{aligned}$$

This implies that the x co-ordinate of the rod coincides with the arc length, $x = s$, and the bending moment $M = r^4(s)y_{ss}$. Therefore

$$G_s = F_s\phi + F\phi_s - M_{ss} = r^2(s)x_{tt}y_s + Fy_{ss} - (r^4(s)y_{ss})_{ss} = r^2(s)y_{tt}$$

or by setting $x(t,s) = s$,

$$Fy_{xx} - (r^4(x)y_{xx})_{xx} = r^2(x)y_{tt}. \tag{1.9}$$

If the tension in the x direction $F = 0$, as we expect near the tip of the tail, then the deflection y satisfies

$$(r^4(x)y_{xx})_{xx} + r^2(x)y_{tt} = 0$$

which is the beam equation. Alternatively if the bending moment M and tension F are constant then

$$Fy_{xx} = r^2(x)y_{tt}$$

which is the wave equation satisfied by a string under tension. In this small deflection limit, aspects of both the wave and beam equations are contained in this inextensible rod model.

1.4 Discrete Block Model

Rather than a continuous tail, we can simplify the model by supposing that the tail consists of a finite sequence of N discrete blocks that are connected by a stiff joint that allows rotation but no normal displacement. The geometry and the applied forces are indicated in Figure 1.8.

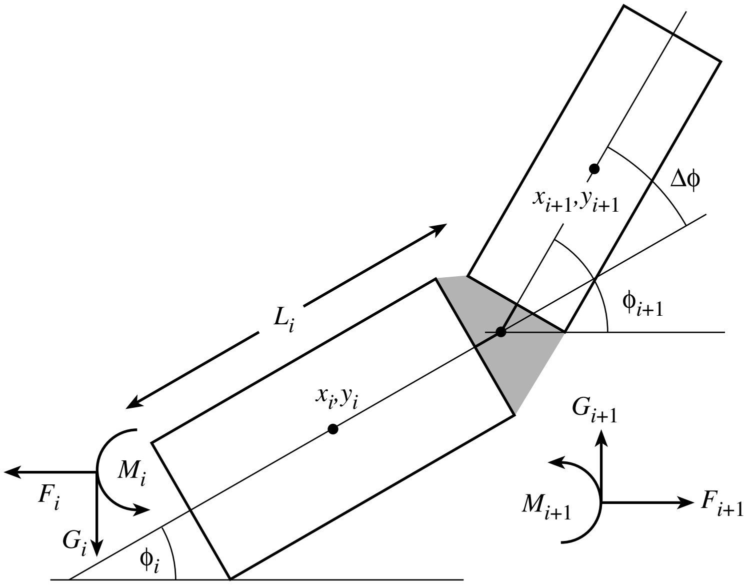


Figure 1.8: Each block is characterized by a mass of $\rho A_i L_i$ where A_i and L_i are the cross sectional area and the length of the ith block. Forces and moments F_i, G_i, M_i act to the left of the ith black and the segments are connected to one another by a stiff joint that allows rotation but no extension. The grey area can be thought of as a uniform distribution of collagen springs with a spring constant k_i.

The co-ordinates of the centre of mass of block $i+1$ is given by

$$\begin{aligned} x_{i+1} &= x_i + \frac{L_i}{2}\cos\phi_i + \frac{L_{i+1}}{2}\cos\phi_{i+1}, \qquad && i = 1, 2, \ldots, N-1 \\ y_{i+1} &= y_i + \frac{L_i}{2}\sin\phi_i + \frac{L_{i+1}}{2}\sin\phi_{i+1}, \qquad && i = 1, 2, \ldots, N-1 \end{aligned}$$

and the origin is taken so that $x_1 = y_1 = 0$. In a similar fashion the net force and bending moment acting on block i give for $i = 1, 2, \ldots, N-1$

$$\begin{aligned} \rho A_i L_i \ddot{x}_i &= F_{i+1} - F_i, \\ \rho A_i L_i \ddot{y}_i &= G_{i+1} - G_i, \\ \rho I_i L_i \ddot{\phi}_i &= M_{i+1} - M_i - \frac{L_i}{2}(F_{i+1} + F_i)\sin\phi_i + \frac{L_{i+1}}{2}(G_{i+1} + G_i)\cos\phi_i, \end{aligned}$$

where the dots denote differentiation with time. For $i = N+1$, $L_{i+1} = 0$ and we choose $F_{N+1} = G_{N+1} = M_{N+1} = 0$ since the last block has a free end. These are simply a discretised version of the original extension free equations (1.7a)-(1.7f). What remains is a discrete version of the Euler-Bernoulli expression relating the bending moment to the curvature of the tail.

Suppose that there is a uniform distribution of collagen springs in the gap between blocks i and $i+1$. Let k_i denote the spring constant measured so that a uniform displacement of length l_i, the springs equilibrium length, generates a restoring force of $-k_i$. In this case the units of k_i is force/area rather than the standard force/length. If we instead suppose there is an angular displacement of $\phi_{i+1} - \phi_i =$

$\Delta\phi \neq 0$ then the springs on one side of the pivot are compressed and the springs on the other side are stretched from their equilibrium length. Each of the springs contributes a force $F = -k_i x \Delta\phi / l_i$ where x is the distance of a given spring from the pivot point and the total force is

$$F_{\text{tot}} = -\iint_A \frac{k_i \Delta\phi}{l_i} x \, dA = 0$$

so that the internal forces are not modified by the rotation of $\Delta\phi$.

The bending moment does changes since

$$M_i = \iint_A xF \, dA = \iint_A \frac{k_i \Delta\phi}{l_i} x^2 \, dA = \frac{k_i I_i \Delta\phi}{l_i} \tag{1.10}$$

where I_i is the moment of inertia of the cross section of block i. If we compare this with a discrete version of the Euler-Bernoulli expression we have

$$M_i = E_i I_i \frac{\Delta\phi}{\Delta s} = E_i I_i \frac{\Delta\phi}{(L_i/2 + L_{i+1}/2}) = \frac{k_i I_i \Delta\phi}{l_i}.$$

So we see that equation (1.10) is a discrete version of the Euler-Bernoulli expression with an elastic modulus of

$$E_i = \frac{k_i}{2l_i}(L_i + L_{i+1})$$

consistent with a stress of magnitude k_i generating a strain of $l_i/(L_i/2 + L_{i+1}/2)$.

In summary the discrete block solution must satisfy for $i = 1, 2, \ldots, N-1$

$$x_{i+1} = x_i + \frac{L_i}{2}\cos\phi_i + \frac{L_{i+1}}{2}\cos\phi_{i+1}, \tag{1.11a}$$

$$y_{i+1} = y_i + \frac{L_i}{2}\sin\phi_i + \frac{L_{i+1}}{2}\sin\phi_{i+1}, \tag{1.11b}$$

$$\rho A_i L_i \ddot{x}_i = F_{i+1} - F_i, \tag{1.11c}$$

$$\rho A_i L_i \ddot{y}_i = G_{i+1} - G_i, \tag{1.11d}$$

$$\rho I_i L_i \ddot{\phi}_i = M_{i+1} - M_i - \frac{L_i}{2}(F_{i+1} + F_i)\sin\phi_i + \frac{L_{i+1}}{2}(G_{i+1} + G_i)\cos\phi_i, \tag{1.11e}$$

$$M_i = \frac{k_i I_i}{l_i}(\phi_{i+1} - \phi_i) \tag{1.11f}$$

where the $\ddot{x}_i$ and $\ddot{y}_i$ in the third and fourth expressions must be consistent with the first two expressions. Furthermore, the internal tension and moments of the first block F_1, G_1, M_1 should be chosen to emulate the time dependent forces that the hip exerts on the tail and $F_N = G_N = M_N = 0$ at the free end of the tail.

1.5 Conclusion

Having considered a cross section of both bipedal and quadrupedal dinosaurs we found that the scaling laws give a first indication that the proportions are typically chosen to balance the dinosaur as opposed to acting as an aid to locomotion.

Two models for the motion of the tail were explored. The first of these was a continuous model and for small deflections it was shown to include aspects of the dynamics of a thin beam as well as the dynamics of wave motion. This is very encouraging since both of these behaviours are seen in the tails of modern day animals. Unfortunately the resulting equations are a set of strongly coupled partial differential equations and more time is required to fully develop a solution consistent with the mass distribution of a given dinosaur.

To simplify the situation a discrete version of the continuous model was developed where the tail is broken into N blocks joined together with a stiff connection that allows rotation but no extension. Once again the result is a set of strongly coupled equations, but there is improvement. First, we are left with ordinary differential equations and second, the Euler-Bernoulli equation in the continuous model is recovered in the discrete model as a result of the behaviour of the springs in each joint. In some sense this result is not really unexpected since the discrete model is simply the continuous model written as a numerical implementation of the method of lines.

The next step is to simulate the motion of a tail predicted with the discrete block model for a living animal to estimate the model predictability. Choosing many segments for the tail of varying stiffness ($E_{\text{bone}} \simeq 20$GPa, $E_{\text{collagen}} \simeq 1$GPa) should produced reasonable dynamics. Once this has been accomplished, one can assess the degree to which a tail would have aided in the locomotion of pedal and quadrupedal dinosaurs.

1.6 Acknowledgements

The authors would like to acknowledge the Pacific Institute for the Mathematical Sciences (PIMS) for their sponsorship of the IPSW. We would also like to take this opportunity to thank the University of Calgary hospitality in hosting this event and special thank-you is reserved for Elena Braverman and Marian Miles. Their hard work and dedication ensured the success of the workshop.

Bibliography

[1] Berger, S.A., Goldsmith, E.W. & Lewis, E.R. (2000). *Introduction to Bioengineering.* Oxford University Press.

[2] Chouaieb, N. & Maddocks, J.H. (2004). Kirchhoff's problem of helical equilibria of uniform rods. *Journal of Elasticity.* **77**, pp. 221-247.

[3] Coleman, B.D., Tobias, I. & Swigon, D. (1995). Theory of the influence of end conditions on self-contact in DNA loops. *Journal of Chemical Physics.* **103**(20), pp. 9101-9109.

[4] Dill, E.H. (1992). Kirchhoff's theory of rods. *Archive for History of Exact Sciences.* **44**(1), pp. 1-23.

[5] Goriely, A. & Tabor, M. (2000). The nonlinear dynamics of filaments. *Nonlinear Dynamics.* **21**, pp. 101-133.

[6] Landau, L.D. & Lifshitz, E.M. (1970). *Theory of Elasticity*, 2nd ed. Pergamon Press: New York.

[7] Nizette, M. & Goriely, A. (1999). Towards a classification of Euler-Kirchhoff filaments. *Journal of Mathematical Physics.* **40**(6), pp. 2830-2866.

[8] Swigon, D., Coleman, B.D. & Tobias, I. (1998). The elastic rod model for DNA and its application to the tertiary structure of DNA minicircles and mononucleosomes. *Biophysical Journal.* **74**, pp. 2515-2530.

[9] Tam, D., Radovitzky, R. & Samtaney, R. (2005). An algorithm for modelling the interaction of a flexible rod with a two-dimensional high-speed flow. *International Journal for Numerical Methods in Engineering.* **64**(8), pp. 1057-1077.

[10] Tobias, I., Swigon, D. & Coleman, B.D. (2000). Elastic stability of DNA configurations. I. General theory. *Physical Review E.* **61**(1), pp. 747-758.

[11] Woodall, S.R. (1966). The large amplitude oscillations of a thin elastic beam. *International Journal of Non-linear Mechanics.* **1**, pp. 217-238.

[12] Animated dinosaurs:
`http://palaeo.gly.bris.ac.uk/dinosaur/animation.html`

[13] Anatomical data:
`http://internt.nhm.ac.uk/jdsml/nature-online/dino-directory/datafiles.dsml`

[14] Dinosaur support information:
`http://www.ucmp.berkeley.edu/diapsids/saurischia/saurischia.html`

`http://www.ucmp.berkeley.edu/diapsids/ornithischia/ornithischia.html`

`http://www.enchantedlearning.com/subjects/dinosaurs/index.html`

Chapter 2

Force-Control for the Automated Footwear Testing System

Problem presented by: Dr. Gerald K. Cole (Biomechanigg Research Inc.)

Mentors: Greg Lewis (University of Ontario IT), Rex Westbrook (University of Calgary)

Student Participants: Ojenie Artoun (Concordia University), Kent Griffin (Washington State University), Jisun Lim (University of Colorado), Xiao Ping Liu (University of Regina), James Odegaard (University of Western Ontario), Sarah Williams (University of California, Davis)

Report prepared by: Greg Lewis (`Greg.Lewis@uoit.ca`)

2.1 Introduction

The Automated Footwear Testing System (AFTS) is a robotic system designed to replicated the movement and loading of a shoe as it contacts the ground during common human movements. By doing so, the AFTS can serve as a system for the functional testing of different footwear designs in a manner that is difficult to achieve by standard testing systems. The AFTS consists of four main components: a robotic Stewart platform, a rigid fixed frame, a load cell and a prosthetic foot. Motion of the foot relative to the ground is created by rigidly fixing the foot to the frame and moving the platform relative to the foot. See Figure 2.1. The Stewart platform has six degrees of kinematic freedom and can reproduce the required complex three-dimensional motion path within the limitations of its range of motion. While the platform is in contact with the footwear, the six-axis load cell measures the three-dimensional forces and moments acting on the prosthetic foot.

It has been shown that when a human subject performs the same movement with two different pairs of shoes, she will adjust her stride so that she feels similar forces on her legs, regardless of the footwear. That is, when testing footwear, it can be assumed that the force profiles will be the same for the different shoes, while the movement path will differ from shoe to shoe. Thus, a good shoe is a one that does not lead to an unstable or unnatural movement path, e.g., one that might lead to an overturned ankle, or one that might lead to the need for an overcompensation that could result in an ‘over-use’ injury. In order for the AFTS to be most effective at testing a wide variety of design features, it would be necessary to develop a means of determining, for any given shoe, a movement path that would

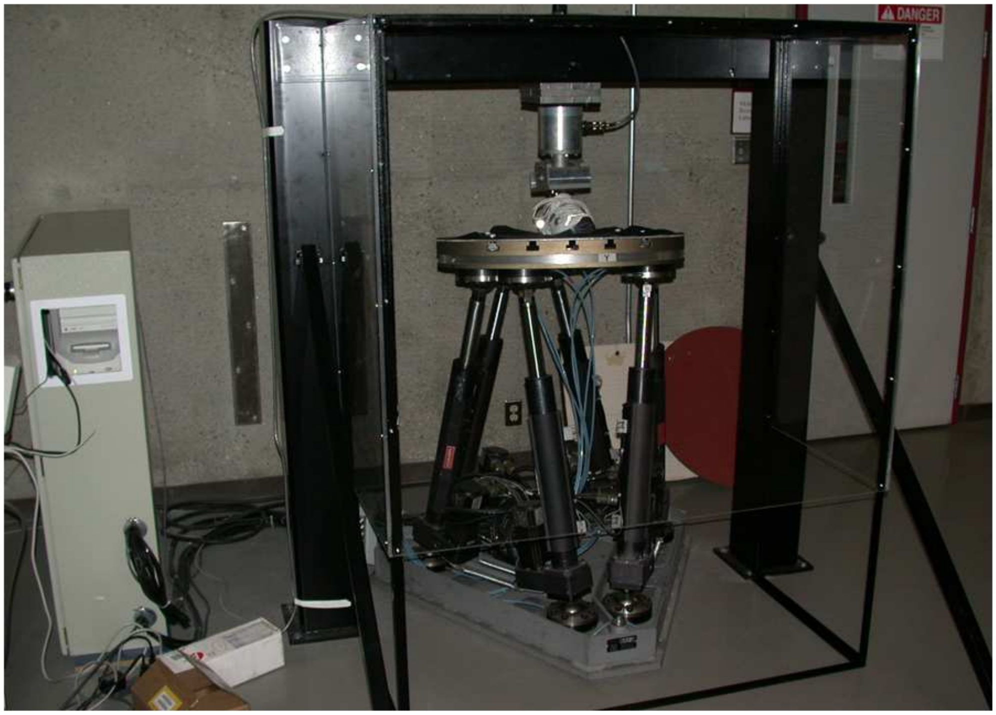


Figure 2.1: The Automated Foot Testing System consisting of the robotic Stewart platform, the prosthetic foot (shown here with shoe attached), the rigid frame, and the load cell (the cylinder just above the shoe).

generate some specified forces and moments that are representative of those that would be generated during the stride of some ‘typical’ human. This movement path could then be analyzed to determine if it is more or less likely to lead to injury.

The force profiles and movement paths for specific types of movements can be acquired experimentally. A time-series of forces can be acquired as a human subject’s foot impacts a footplate during a stride, and markers on the shoe can be tracked in order to acquire a time-series of the position of the foot, i.e., a movement path. The position data includes the x, y, z positions, as well as the angles that the foot rotates about the x, y and z axes. These angles are generally referred to as roll, pitch and yaw, respectively, and we will denote them as α, β, and γ, respectively. The forces measured by the footplate are used to calculate forces in three directions F_x, F_y and F_z, and the moments M_x, M_y, and M_z, about the x, y and z axes, respectively, in the foot coordinate system. These forces and moments can be used as those felt by a typical human, i.e., the ‘target forces’.

For the AFTS, a movement path is specified, translated into platform coordinates and executed on the machine. During the execution, the load cell measures the forces and moments that act on the prosthetic foot. We wish to find the particular movement path of the Stewart platform that will generate the target force profile. Thus, we are interested in solving an inverse problem. The main goal of the workshop was to investigate potential solution methods for this ‘force-control’ problem, including looking into its feasibility.

When the same shoe used by the human subject is mounted on the prosthetic foot of the AFTS, and the experimentally measured movement path is replicated on the Stewart platform, the forces and moments measured to be acting on the prosthetic foot do not match the experimental data. The forces acting normal to the ground/platform are similar in magnitude for both cases. However, the forces acting parallel to the ground/platform are not similar. Thus, before using the AFTS to test different footwear, it is necessary to determine the platform movement path that leads to the target force profiles for the ‘control’ shoe, i.e., the shoe used during the acquisition of the target force profile. This may also be viewed as a force-control problem. It may be reasonable to use such an approach if the discrepancies between the movement paths for the human subject and platform are relatively small. However, if they are sufficiently large, it would lead to difficulty in the interpretation of any testing results. That is, the causes of these discrepancies may reveal information regarding the feasibility of using force-control as a means of testing footwear. Thus, we seek possible origins of the discrepancies.

We first investigate the possibility of performing a closed-loop control of the forces. That is, we investigate the possibility of adjusting the position of the platform at discrete points along the movement path until the forces measured by the load cell of the AFTS match the target forces at that point. The results, discussed in Section 2.2, indicate that there are some fundamental issues that must be considered before the AFTS can reliably be used as a testing system. We study two such issues. The first study looks at the effects due to the choice of the origin of the platform coordinate system. See Section 2.3. This choice might effect how the Stewart platform executes the specified motion, and thus might effect the measured forces. In the second study, presented in Section 2.4, the system is modelled as a simple elastic body in order to gain some information regarding the feasibility of solving the inverse problem. The results suggest that it may be more appropriate to take a global rather than local approach to controlling the forces. In Section 2.5, we discuss the possibility of parameterizing the movement path using cubic splines, and then minimizing, with respect to the parameters of the curves, a functional that is small when the measured forces are near the target forces. Conclusions follow.

2.2 Closed-loop Force Control

We wish to find the series of platform positions (i.e. the movement path) that will lead to the force and moment profiles that are measured in the human subject (i.e. the target forces). One possible means of achieving this would be to perform a ‘closed-loop’ force control. That is, at discrete intervals along the path, the platform position is adjusted until the forces and moments that are measured at the load cell match the target forces. Ideally, it would be possible to perform a Newton-type iteration, where the initial guess could be either the experimentally measured position or the position found at the previous step, and an approximate Jacobian could be computed by measuring the changes that occur in the three forces and three moments as the six position variables are incremented successively by a small amount, while the other position variables are held constant at their initial values. That is, the approximate Jacobian could be given by

$$J = \begin{bmatrix} \frac{\Delta F_x}{\Delta x} & \frac{\Delta F_x}{\Delta y} & \frac{\Delta F_x}{\Delta z} & \frac{\Delta F_x}{\Delta \alpha} & \frac{\Delta F_x}{\Delta \beta} & \frac{\Delta F_x}{\Delta \gamma} \\ \frac{\Delta F_y}{\Delta x} & \frac{\Delta F_y}{\Delta y} & \frac{\Delta F_y}{\Delta z} & \frac{\Delta F_y}{\Delta \alpha} & \frac{\Delta F_y}{\Delta \beta} & \frac{\Delta F_y}{\Delta \gamma} \\ \frac{\Delta F_z}{\Delta x} & \frac{\Delta F_z}{\Delta y} & \frac{\Delta F_z}{\Delta z} & \frac{\Delta F_z}{\Delta \alpha} & \frac{\Delta F_z}{\Delta \beta} & \frac{\Delta F_z}{\Delta \gamma} \\ \frac{\Delta M_x}{\Delta x} & \frac{\Delta M_x}{\Delta y} & \frac{\Delta M_x}{\Delta z} & \frac{\Delta M_x}{\Delta \alpha} & \frac{\Delta M_x}{\Delta \beta} & \frac{\Delta M_x}{\Delta \gamma} \\ \frac{\Delta M_y}{\Delta x} & \frac{\Delta M_y}{\Delta y} & \frac{\Delta M_y}{\Delta z} & \frac{\Delta M_y}{\Delta \alpha} & \frac{\Delta M_y}{\Delta \beta} & \frac{\Delta M_y}{\Delta \gamma} \\ \frac{\Delta M_z}{\Delta x} & \frac{\Delta M_z}{\Delta y} & \frac{\Delta M_z}{\Delta z} & \frac{\Delta M_z}{\Delta \alpha} & \frac{\Delta M_z}{\Delta \beta} & \frac{\Delta M_z}{\Delta \gamma} \end{bmatrix}. \tag{2.1}$$

In practice, this could be computed by incrementing a single position variable, measuring the forces, incrementing that position variable back to its original value, and repeating this for each of the position variables. Once the approximate Jacobian is computed, it could be used to choose the next iterate. The platform would then be moved into the corresponding position, and the forces would be measured. If the forces are still not sufficiently close to the targets, another iteration could be performed, perhaps via a quasi-Newton iteration, or perhaps a new Jacobian could be computed. This procedure would continue until the desired forces are obtained to within a given tolerance. We could then proceed to the next point along the movement path, and find the platform position corresponding to the target forces that are required at this new point.

For this method to be feasible, the computed Jacobians must be non-singular. In order to test this, we computed the approximate Jacobian on the AFTS at two points along the movement path. The most striking result was that when certain position variables were incremented and returned to their starting values, the measured forces did not return to their original values. Even when the position variables were incremented by as little as 0.1mm and returned to their original values, the forces and moments could be as much as 5% different from their starting values.

Upon inspection of the AFTS in use, it was found that certain movements caused the shoe to slip along the platform. Such irreversible behaviour will greatly hinder any force-control procedure. Indeed, the discrepancies between the forces measured for the human subject and those measured for the platform for the same movement path could be caused to a large extent by the slipping. This is consistent with the observation that the forces normal to the ground/platform are sufficiently similar, while the tangent forces are not.

It is not surprising that when the approximate Jacobian was formed, we found that it was singular. It was seen, however, that only certain directions were irreversible, and it was speculated that this was caused by slipping when increments were made in these directions. In order for force-control to be possible, steps must be taken to reduce the slipping as much as possible. The platform being used for the data acquisition was quite worn, which likely exacerbated the problem. Thus, it is possible that the installation of a new platform surface designed to limit slipping would greatly improve the prospects. Either way, a method that minimizes slipping, in particular in the specific directions, will greatly increase the chances of success.

2.3 Stewart Platform Dynamics

It is expected that even when steps are taken to reduce slipping, a path-dependence of the forces measured at the load cell will likely linger. Thus, it is not only important to improve reversibility, but also to maximize reproducibility.

A run of the AFTS begins by raising the platform until it comes in contact with the shoe. The origin of the platform coordinate system is chosen as this initial point of contact. Currently, care is not taken to ensure that this point of contact is the same for each run. However, due to the method that is used to transform the experimentally measured movement path into platform positions, the choice of the origin of the platform coordinate system will affect the resulting platform movement path (see below). We therefore investigate the magnitude of this effect so that we may determine whether this is a potential cause of error, and whether care must be taken to choose the origin to be the same for each run of the AFTS.

Thus, we need to look into the dynamics of the Stewart platform. The platform has six degrees of freedom determined by the length of the actuators (legs), where each set of leg lengths corresponds to a unique position and orientation of the platform.

The actual platform path is not directly specified by the user. The user supplies a series of ‘way-points’, which are a series of positions that the platform must pass through, but the user does not have control over the path that is taken to go from one way-point to the next. The path between each pair of way-points is determined by an algorithm that requires all actuators to start and stop at the same time. Thus, these intermediate paths may be quite different depending on where the shoe initially contacts the platform (i.e., the choice of origin for the platform coordinate system). Because we were not able to test this on the AFTS itself, we performed a theoretical investigation of the path differences that might occur for three different origin locations, as shown in Figure 2.2. A central location was chosen, then the two other locations were chosen 5cm away from this central location along the x and y axes, respectively. We computed the platform paths by first computing the leg lengths corresponding to a series of way-points using the software package designed for this purpose. By assuming that during the transition between the way-points all the actuators would start and stop at the same moment, we determined a series of leg lengths that would occur between each of the way-points. We then used numerical methods to invert the nonlinear relation between the leg lengths and platform position, and obtained the intermediate platform positions that corresponded to the intermediate leg lengths.

A sample of results is plotted in Figure 2.3. An interesting observation is that the movement path has kinks at the way-points. It can also be seen that indeed the paths are different depending on the location of the coordinate system, although they are not more than 0.002mm for any given position variable. However, we did see that increments as little as 0.1mm could cause significant changes in the forces. Furthermore, it might be expected that there would be a cumulative effect. Thus, it is not clear that these small path differences would not have an effect on the measured forces. Therefore, to be sure that errors are not introduced, we suggest that care be taken to ensure that the origin is chosen as much as possible in the same location for each run. This may increase the reproducibility, and thus the reliability of the testing system.

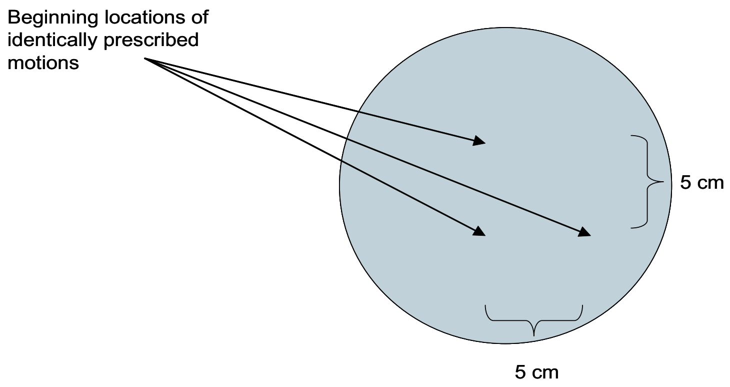


Figure 2.2: Three different origins of the platform coordinate system.

2.4 Feasibility Study

In the problem described above, we are trying to determine the displacements that must be imposed at the lower boundary (i.e., the bottom of the shoe) in order to generate some specified forces at the upper boundary (i.e., the load cell). To demonstrate the difficulty involved in solving these types of inverse problems for elastic bodies, we study a simple forward problem. We determine the displacements that occur in a planar elastic block for three different sets of lower boundary conditions (i.e., displacements that are imposed at the lower boundary). See Figure 2.4. We then calculate the forces that are generated at the upper boundary due to the resulting displacements. If we assume that the lower boundary is linear (i.e. that the displacements at the lower boundary vary linearly), we can choose the three boundary conditions such that they form a 'basis' for all possible boundary conditions. The situation when the lower boundary is linear corresponds to the case when the platform is in contact with the whole shoe. Although this is not a good assumption for many of the motions of interest, it is sufficient for the purposes of this feasibility study.

The displacements in the elastic block are described by the Navier equations

$$(\lambda+2\mu)\frac{\partial^2 u}{\partial x^2}+\mu\frac{\partial^2 u}{\partial y^2}+(\lambda+\mu)\frac{\partial^2 v}{\partial x\partial y} = 0, \tag{2.2}$$

$$\mu\frac{\partial^2 v}{\partial x^2}+(\lambda+2\mu)\frac{\partial^2 v}{\partial y^2}+(\lambda+\mu)\frac{\partial^2 u}{\partial x\partial y} = 0, \tag{2.3}$$

where $u(x,y)$ is the displacement from the 'no force' position in the x direction, $v(x,y)$ is the displacement in the y direction, and the constants λ and μ are the Lamé coefficients. The lower and upper boundary are taken to be at $y=0$ and $y=1$ respectively, while the side boundaries are taken to be at $x=0$ and $x=1$. At the upper boundary, we assume no displacements, i.e. we have $u(x,1)=0$ and $v(x,1)=0$, while we take the boundary conditions on both sides to be stress free, i.e. we take

$$(\lambda+\mu)\frac{\partial u}{\partial x}+\lambda\frac{\partial v}{\partial y}=0 \quad \text{and} \quad \mu\left(\frac{\partial u}{\partial y}+\frac{\partial v}{\partial x}\right)=0 \tag{2.4}$$

at $x=0$ and $x=1$.

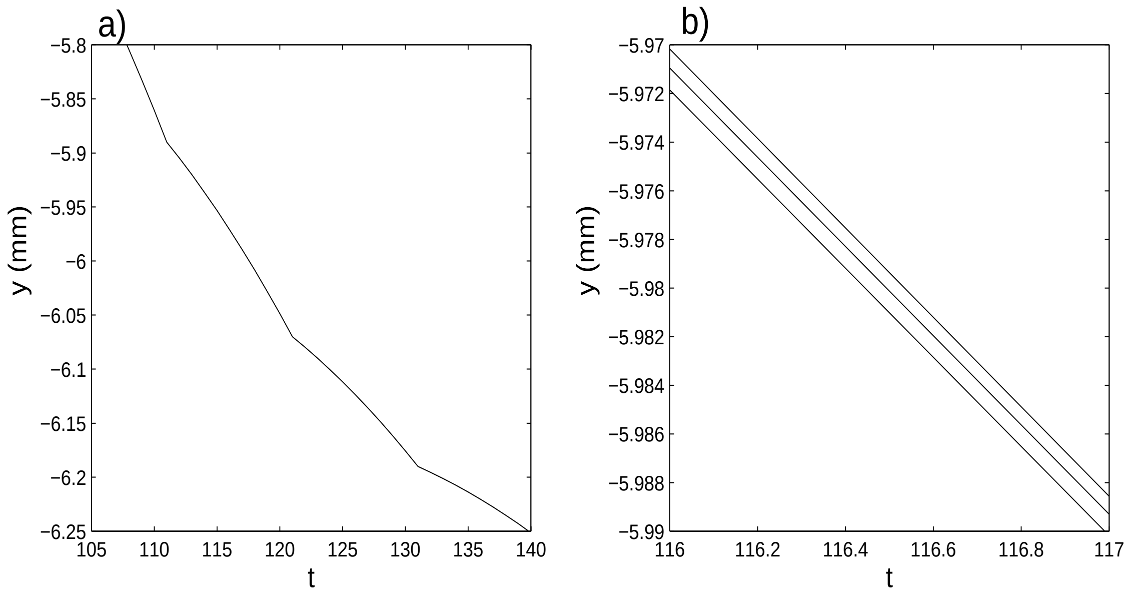


Figure 2.3: Platform dynamics. (a) A portion of a typical path taken by platform; kinks occur at the way-points, (b) three different paths corresponding to the three different 'platform origins' depicted in Figure 2.2.

For the three different cases, we chose three different lower boundary conditions. See Figure 2.4. In the first case, we choose $u(x,0)=c$ and $v(x,0)=0$, where c is some constant. This case corresponds to pure shear in the x direction. For the second case, we consider pure compression in the positive y direction, i.e., we have $u(x,0)=0$ and $v(x,0)=a$, where a is some constant. In the third case, we take $u(x,0)=0$ and $v(x,0)=bx$, where b is some constant, which corresponds to a lower boundary that ramps linearly from zero compression at $x=0$ to a maximum compression at $x=1$. Any linear condition on the displacements at the lower boundary can be represented as linear combinations of these three input displacements.

We solve this system of partial differential equations (2.2–2.3) numerically using finite differences on a 50×50 grid. We choose the constants $\lambda=1$, $\mu=1/2$, $a=0.5$, $b=0.5$, and $c=0.5$.

Once the displacements have been found, the forces and moment can be calculated using

$$F_y = \int \left((\lambda+2\mu)\frac{\partial v}{\partial y} + \lambda\frac{\partial u}{\partial x} \right) dx \qquad \text{(normal)} \qquad (2.5)$$

$$F_x = \int \mu \left(\frac{\partial u}{\partial y} + \frac{\partial v}{\partial x} \right) dx \qquad \text{(shear)} \qquad (2.6)$$

$$M = \int x \left((\lambda+2\mu)\frac{\partial v}{\partial y} + \lambda\frac{\partial u}{\partial x} \right) dx \qquad \text{(moment).} \qquad (2.7)$$

We are interested in the values for these forces at the upper boundary ($y=1$). We denote the normal force at the upper boundary as F_1, the stress along the upper boundary as F_2, and the moment as M, and calculate each of these for each of the three boundary conditions. We obtain a 3×3 matrix that

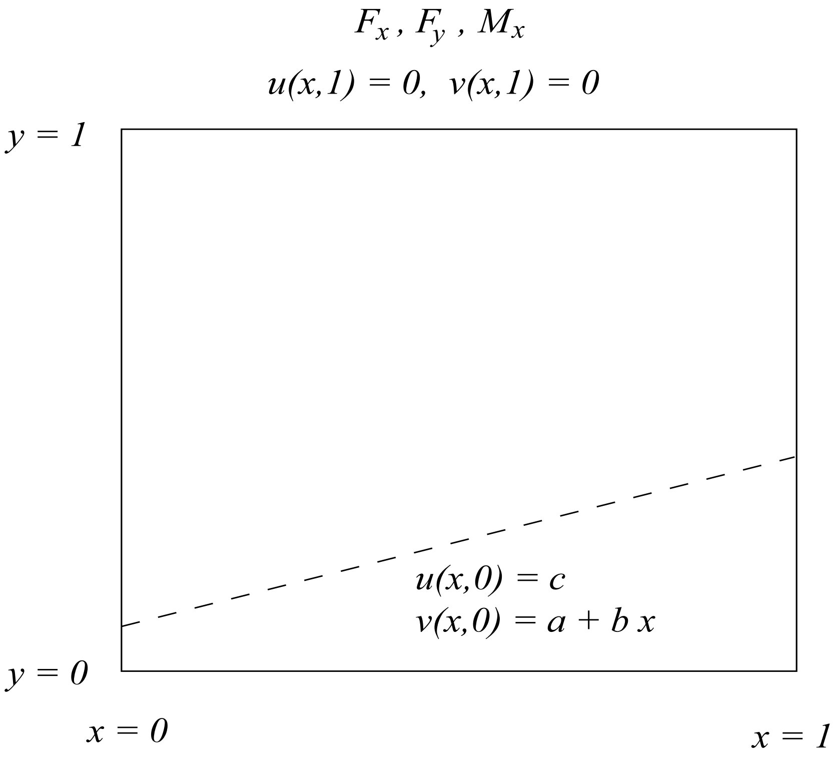


Figure 2.4: An elastic block of unit length and height. Lower boundary conditions for the displacements $u(x, y)$ and $v(x, y)$ are different for the three different cases that are studied (corresponding to different values of the constants a, b, and c), while the side and upper boundary conditions are the same.

defines the relationship between our input displacements and our output forces:

$$A = \begin{bmatrix} F_{11} & F_{12} & F_{13} \\ F_{21} & F_{22} & F_{23} \\ M_1 & M_2 & M_3 \end{bmatrix}$$

where F_{ij} is the force i in the jth case, and M_j is the moment in the jth case. With the displacement fields computed above, this matrix becomes

$$A = \begin{bmatrix} -0.0053 & 0.5768 & 0.2845 \\ -0.0147 & 0.0041 & -0.0099 \\ 0.0112 & 0.2845 & 0.1426 \end{bmatrix}.$$

The condition number of this matrix, which is the ratio of singular values, indicates the sensitivity of the forces to the changes in the boundary conditions. If the condition number is small, then we would expect that changes in the lower boundary would cause changes of a similar magnitude in the force, which would indicate that the inverse problem was well-conditioned. However, if the condition number is large, then we might expect that the matrix is close to singular, which would imply that the columns are close to being linearly dependent, which in turn would imply that different combinations of inputs would produce very similar outputs. That is, the forces are not very sensitive to changes in the lower boundary, and thus, the inverse problem is not very well conditioned.

The condition number for the matrix A is given by

$$\mathrm{cond}(A) = 147.7,$$

indicating that the forces are not very sensitive to changes in the lower boundary. This can be seen more clearly in Figure 2.5, in which the forces (represented by the three plots on the right side of the figure) generated by variation of the lower boundaries conditions (shown on the left of the figure) are presented on the same plot. It can be seen that even large differences in the lower boundary conditions can result in only small changes in the forces.

This example provides evidence regarding the difficulty that may be involved in attempting to determine the lower boundary conditions given the forces at the upper boundary. That is, the inverse problem may not be well-conditioned. In such cases, finding solutions becomes difficult. Iterative methods tend to converge slowly, and it is possible that they may not converge at all.

However, these results depend on the specific choices of the parameters of the problem λ and μ. Because we did not know the actual values of these parameters for the shoe, reasonable approximations were chosen. Errors in this choice may affect the conclusions of this example.

Factors that effect the conditioning that we have not considered include the movement of the shoe on the prosthetic foot, which is expected to lead to poorer conditioning. Such movement would decrease the sensitivity of the forces due to changes in the lower boundary, and thus increase the condition number.

The zero displacement condition assumed at the top the prosthetic foot is almost certainly not satisfied by the human foot and hence, no matter what continuum model is used for the foot-shoe combination, any attempt to examine the problem analytically will lead to different results for the two problems even if the displacement conditions at the shoe plate interface can be accurately reproduced.

2.5 Movement Path Parameterization

The evidence presented above indicates the difficulty involved in using closed-loop force control to solve this problem. We, therefore, explore the possibility of non-locally controlling the forces along a parametrized movement path. It is expected that the conditioning of the inverse problem will still be an issue for this approach. However, variation of the parameters of the movement path would not lead to unnatural movements, which would reduce (perhaps eliminate) the need to make platform adjustments in directions that would cause unavoidable slipping. Because we did not have sufficient time during the workshop for a full investigation, we describe only briefly how one might go about using path parametrization in this problem.

We begin by parameterizing the position data obtained from the human subject. We proceed by choosing several points on the curves of each of the position variables. Examples for the spatial coordinates are shown in Figure 2.6. The number and position of the points are chosen in such a way as to maximize the reproduction of the qualitative features of the curves while minimizing the number of parameters needed. For example, for the z position data, it was judged that four points were necessary to obtain a parametrized curve that could approximate both the sharp increase and decrease that is observed at the beginning and end, respectively, of the time-series. After the points have been chosen, cubic spline interpolation can be used to obtain the parametrized approximation to each of the curves.

For data acquired during the contact phase of a heel-toe run, the time-series of the six position variables could be reproduced reasonably well by fitting cubic splines to a total of 34 points. The position data for the 3 spatial coordinates is shown in Figure 2.6; data for the 3 angles is not shown. Thus, the path can be written as a function of the 34 parameters

$$\text{Path} = P(p_1, p_2, \ldots, p_{34}),$$

where p_i are the path parameters that can be adjusted to vary the movement path. An example of how the path might change when one of the parameters is varied is shown in Figure 2.7.

The forces can now be measured as the AFTS executes the initial parameterized path taken from the human subject data. That is, we have

$$\text{Force} = F(P),$$

and we would like to find the 34 parameters p_i that will reproduce the target force profile F_{target}. In practice, we will try to minimize some functional (e.g. with respect to the L^2 norm) of the force and target force profiles over all possible parameter values. That is,

$$\min_{P} \|F_{\text{target}} - F(P)\|_2.$$

A standard method, perhaps a non-Jacobian method such as a polytope algorithm, might be used for the minimization.

For the closed-loop force-control problem, we look for zeros of a function of six variables for each interval along the path, whereas, here we are minimizing a single functional over 34 parameters. Although only one minimization problem needs to be solved, a large number of parameters are involved. The question arises whether such a method is feasible. Indeed, even if a Jacobian need not be calculated, a single ‘function evaluation’ consists of a full run of the AFTS, which took much longer than one minute. It is not known how many such function evaluations would need to be executed to determine the path parameters. However, it could possibly be in the hundreds. It is possible that a sufficient solution could be obtained by variation over only a smaller subset of the path parameters. These and other considerations require extensive further study before such a method could be implemented effectively.

2.6 Conclusions

We would like to determine the particular movement path that would generate a specified target force profile. We examined the feasibility of performing a closed-loop control of the forces, and found that the nature of the problem does not lend itself well to this method. We do not conclude that it is impossible to use closed-loop force control to solve the problem. However, the evidence indicates that it would be very difficult to do so.

Parametrization of the movement path is one possible alternative. The promising feature of this method is that it would not lead to unnatural movements that could cause the shoe to slip along the platform. Thus, it is expected that the reproducibility of force measurements would be significantly improved. The conditioning of this method is not known; further study is required before conclusions regarding the method's feasibility can be made. Such investigations would require extensive data acquisition using the AFTS itself.

Regardless of the method used, we discovered that it is necessary to reduce slipping of the shoe along the platform as much as possible. Simply resurfacing the platform may lead to significant improvements in this respect. We also found that the platform will follow a different trajectory depending on the origin of the platform coordinate system. Although the path differences are small, significant cumulative errors may arise. Thus, it would be prudent to ensure that the platform origin does not vary from run to run.

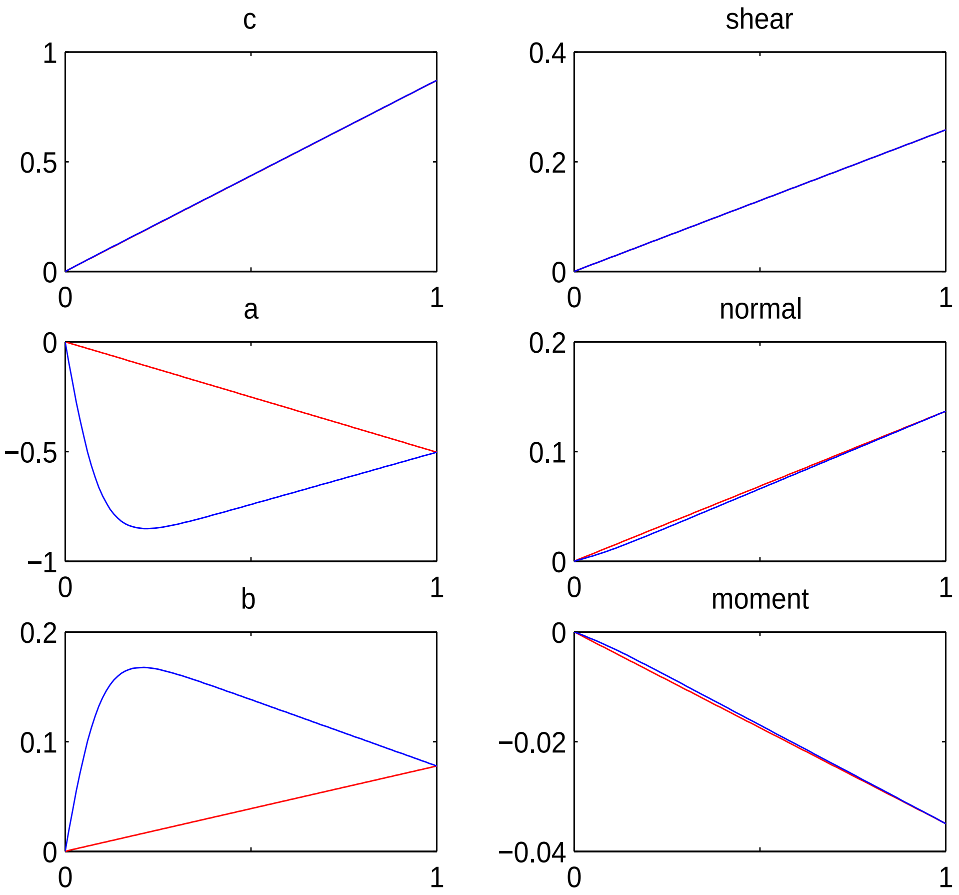


Figure 2.5: The two curves in the left panels represent two different sets of boundary conditions, i.e. of the constants a and b representing the amount of compression at the bottom boundary and slope of the bottom boundary, respectively, where the y-axis gives the values of the constants and the x-axis represents a parametrization for the changes in the constants. The curves in the right panels show the resulting differences in the forces, i.e., large changes in the boundary conditions only result in small changes in the forces, where the y-axis gives the values of the forces and the x-axis represents the same parametrization as in the left panels.

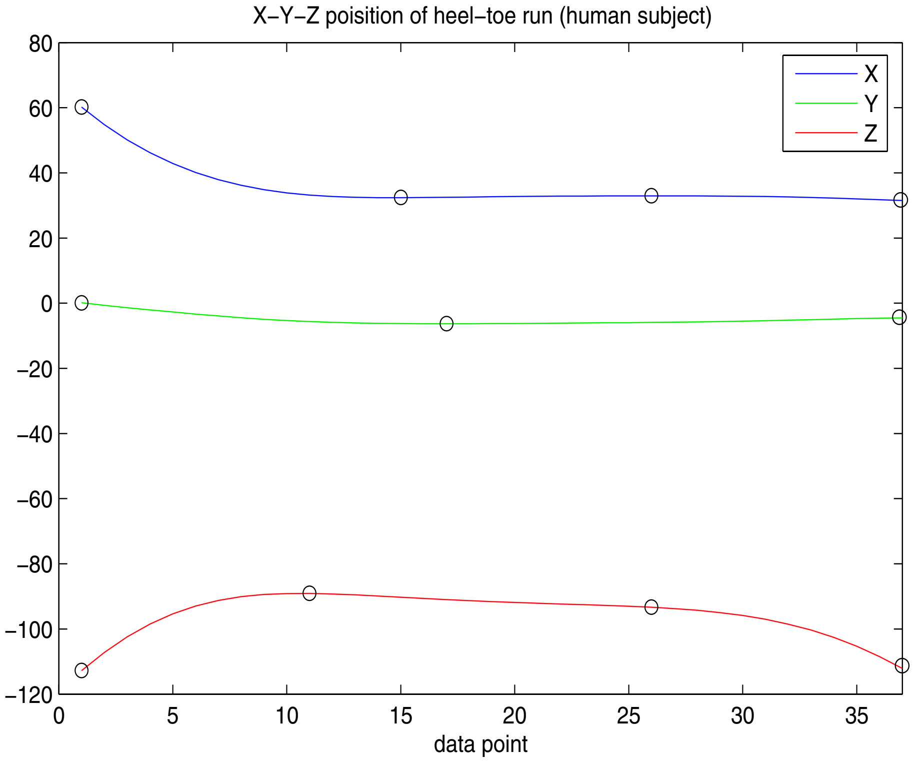


Figure 2.6: Position data in the x, y, and z directions for a heel-toe run of a human subject. Points along the curves (the circles) have been chosen such that spline interpolants through these points will reasonably reproduce the curves.

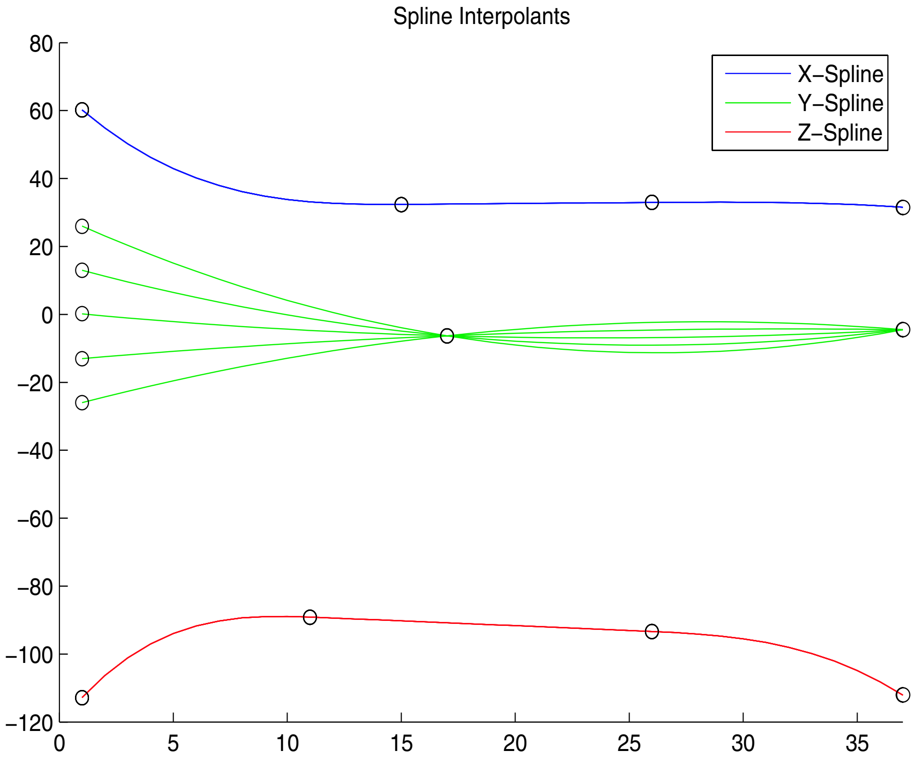


Figure 2.7: Spline interpolants of the points taken from the x, y, and z position data of a heel-toe run of a human subject, as shown in Figure 2.6. Variation of one of the points represents how the path can change as the path parameters are varied.

Chapter 3

Seismic Image Analysis Using Local Spectra

Problem presented by: Pierre Lemire & Rob Pinnegar (Calgary Scientific Inc.)

Mentors: Elena Braverman (University of Calgary), Michael Lamoureux (University of Calgary), Qiao Sun (University of Calgary)

Student Participants: John Gonzalez (Northeastern University), Hui Huang (University of British Columbia), Parisa Jamali (University of Western Ontario), Yongwang Ma (University of Calgary), Hatesh Radia (University of Massachusetts at Lowell), Jihong Ren (University of British Columbia), Dallas Thomas (University of Lethbridge), Pengpeng Wang (Simon Fraser University)

Report prepared by: M. Lamoureux (`mikel@math.ucalgary.ca`)

3.1 Introduction

During the one week Industrial Problem Solving Workshop held in June 2005 at the University of Calgary, hosted by the Pacific Institute for the Mathematical Sciences, our group was asked to consider a problem in seismic imaging, as presented by researchers from Calgary Scientific Inc. The essence of the problem was to understand how the S-transform could be used to create better seismic images, that would be useful in identifying possible hydrocarbon reservoirs in the earth.

3.2 Problem Description

Our group was presented with the following summary of the problem under consideration:

> Calgary Scientific Inc. is currently developing a technique to classify pixels in seismic pseudosections based on their local spectral characteristics. These are obtained from a modified Gabor transform, in which only certain (k_x, k_y) wavevectors are represented in the local spectrum of each pixel. This classification technique involves finding the dominant peak in each local spectrum, and identifying the corresponding pixel with the wavevector and amplitude of the dominant peak.

> This method proved ineffective since it is inflexible when one wants to investigate any interesting features not associated with the dominant peak. The identification of secondary peaks is complicated by the fact that dominant peaks typically cover several pixels of its local spectrum; hence, the wavevector with the second largest amplitude is likely to contain a significant contribution from the primary peak. The main goals were to efficiently identify secondary peaks and to identify correlations amongst peaks from pixel to pixel.
>
> Ideally, a fully processed seismic pseudosection should contain all the information the interpreter needs to unambiguously identify potential drilling targets, such as reefs, anticline traps, and so forth. This involves identifying layer boundaries on the pseudosection, a process that can be considered as more or less equivalent to identifying the layers themselves as continuous groups of pixels. The presence of significant amounts of noise in the data, and the accumulation of errors during the numerical steps of seismic processing, can complicate this process by obscuring important features in the pseudosection. A final goal is to improve the signal to noise ratio to facilitate interpretation of the pseudosection.

There were a number of researchers on the project with expertise in seismic imaging, and an important first step was to understand what aspect of the imaging problem we were being asked to study. However, since we would not be working directly with raw seismic data, traditional seismic techniques would not be required. Rather, we would be working with a two dimensional image, either a migrated image, a common mid-point (CMP) stack, or a common depth point (CDP) stack. In all cases, the images display the subsurface of the earth with geological structures evident in various layers. Figure 3.1 is a typical sample image.

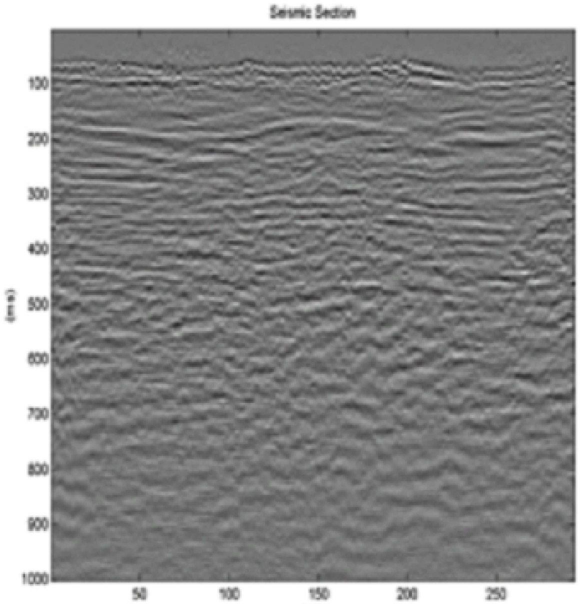


Figure 3.1: A sample seismic section.

For a given image the local spectrum is computed at each point resulting in a surface similar to Figure 3.2. The various peaks in the spectrum are used to classify each pixel in the original seismic image resulting in an enhanced and hopefully more useful seismic pseudosection.

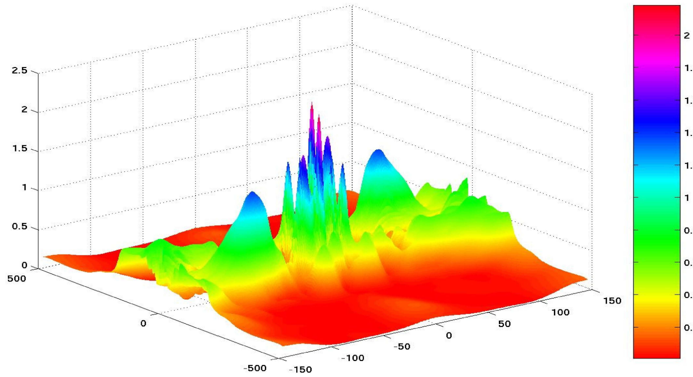


Figure 3.2: A typical local spectrum.

Thus, the objective of this project was to improve the identification of layers and other geological structures apparent in the two dimensional image (a seismic section, or CDP gather) by classifying and coloring image pixels into groups based on their local spectral attributes.

3.3 Spectral Background

The local spectra of the pixels are obtained from the Stockwell, or S-transform, which is a time-frequency spectral localization method similar to the short-time Fourier transform (STFT) [7]. The S-transform of the one-dimensional function $h(x)$ is defined as

$$S(x,k) = \int_{-\infty}^{\infty} h(x') \frac{|k|}{\sqrt{2\pi}} e^{-(x-x')^2k^2/2} e^{-2\pi i k x'}\, dx'.$$

Because $\int_{-\infty}^{\infty} S(x,k)\, dx = H(k)$, the Fourier transform of $h(x)$, one can verify that $h(x)$ is recoverable from its S-transform. One can interpret the S-transform as a continuous wavelet transform where the mother wavelet is a Gaussian in which the wavenumber k plays the role of a dilation. This definition is easily generalized to higher dimensions and for the two-dimensional seismic images under consideration the Stockwell transform takes the form

$$S(x,y,k_x,k_y) = \int_{-\infty}^{\infty}\int_{-\infty}^{\infty} h(x',y')w(x-x',y-y',k_x,k_y)e^{-2\pi i x' k_x} e^{-2\pi i y' k_y}\, dx'\, dy', \tag{3.1}$$

where $h(x,y)$ is the pixel intensity of a given location and $w(x,y,k_x,k_y)$ is the corresponding two-dimensional Gaussian window. The Stockwell transform $S(x,y,k_x,k_y)$ indicates the strength of the frequencies with wavenumbers (k_x,k_y) in a neighbourhood of the point (x,y). The attributes of interest in this localized spectrum were identified to be the wavenumber (k_x,k_y) and the magnitude of any peaks in the function $S(x,y,k_x,k_y)$ as (x,y) varies across the image. These peaks are then used to classify the pixels.

The focus, then, is on identifying peaks in the local spectrum. There remained three main issues that the company wished to be addressed:

1. In Calgary Scientific's current algorithm, the four-dimensional S-transform is collapsed to a two-dimensional image by identifying each pixel with the wavevector at which its local S-spectrum is largest. That is, one particular peak is identified. This approach assumes that only one waveform is important at any pixel in the image.

2. A reliable means of automated edge detection is elusive. Although the approach described above can be used to produce a visually interpretable pseudo-image, it cannot be used to divide the image into layers because the peak wavevector can vary from pixel to pixel within a particular layer.

3. In Calgary Scientific's implementation, the S-transform is evaluated at a set of positions $\{(x, y)\}$ which changes as the wavevectors change. This complicates any pattern matching because the local S-spectra of adjacent (x, y) pixels are typically not defined on the same set of wavevectors. There are some standard techniques to account for this but they all introduce artifacts in the transformation. Is there a better way to compute the transform at all interesting positions (x, y) and wavenumbers (k_x, k_y)?

3.4 Addressing Issue One: Identifying Spectral Peaks

In the sample data sets that we considered, the spectral peaks were all remarkably smooth, and Gaussian-like. Whether this was a property of the data, or a property of the S-transform was not investigated in detail. However, it appears that it is a reflection of the redundancy in the S-transform. The smoothness, and Gaussian shape, suggested a number of approaches for identifying peaks. A summary of the ones we considered were:

- Reduce the two-dimensional peak selection problem to a series of one-dimensional problems. Using the slicing technique, we can consider all one-dimensional sections of the spectrum and analyze the peak structure of each slice. From this we identify the peaks of the entire spectrum. In the one-dimensional slice, the spectrum appears smooth enough (twice differentiable), to find and classify all the maxima.

- Use a discrete, derivative to find critical points, and identify peaks. Since the function is smooth, this technique should rapidly identify the few peaks there are. Ranking them by height is simple.

- Represent the function as a sum of Gaussians and use a least squares technique to specify their individual parameters. After that, the highest peaks are easily classified.

- Identify the highest peak and subtract an appropriate Gaussian-like function centered at this peak. After the subtraction, find the highest peak and repeat the procedure. It was believed that this could expose high peaks shadowed by adjacent higher peaks.

- Use clustering techniques to group the data around peaks, thus dividing all the data into several groups associated with various peaks. Later these groups can be ranged in accordance to the peak values.

Code for each of these attempted techniques can be found in the Appendix. The subtraction method was the most successful and is worth discussing in further detail. The location of the first peak is identified by searching for the highest data point in the matrix of spectral points. In a neighbourhood of this peak the function is approximated by a two-dimensional Gaussian with a width optimized to best fit the data. This best-fit Gaussian is then subtracted from the spectrum, and we have a data set with one peak removed. The process is then repeated (identify new peak, fit with Gaussian, subtract Gaussian) until all significant peaks are identified. Empirically, the first three or four peaks might be of interest.

Figure 3.3 gives a demonstration of how the method works. In the first panel, we see two large narrow peaks. By matching with an appropriate Gaussian, those peaks are subtracted leaving the spectrum in the middle panel. Two large, wide Gaussians are then identified as the next peaks and are again subtracted leaving the third panel. This sequence is repeated until all the major peaks have been identified. Symmetry in the spectrum allow the peaks to be removed in pairs.

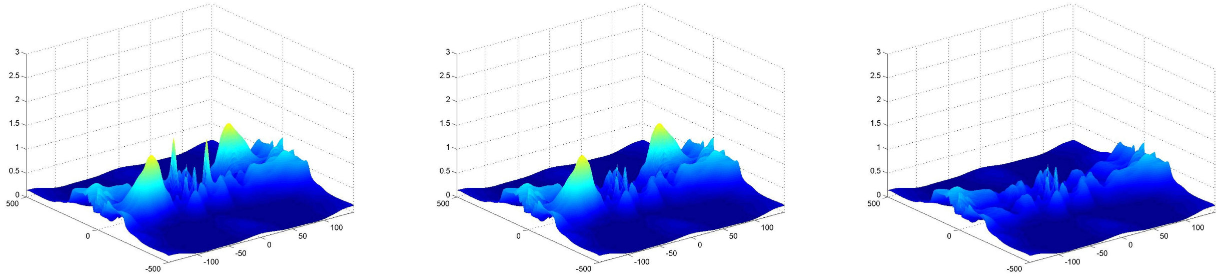


Figure 3.3: A sequence of peaks being subtracted from the data.

Also interesting is why the other methods failed. The slicing technique failed because one-dimensional slices do not uniquely identify two-dimensional peaks. However, the analysis of this proposed technique lead naturally to the two-dimensional derivative method.

The two-dimensional derivative technique identified far too many local extrema. Although most of these were down in the noise level of the data, it took a significant amount of computing time to distinguish between real peaks and extrema at the noise level of the spectrum. In the end, this method was not computationally efficient.

The least squares technique also failed because of high computational costs. On the surface this seems counterintuitive since the least squares methods is well-known for its speed. However, here we were looking to optimize over a sum of Gaussians, where the heights, widths, and locations were variables to be optimized. This is a highly non-linear problem and a computationally efficient method was not obtained in the week of the workshop.

Finally, the clustering technique also failed for computational reasons. We tried using a standard implementation of the k-mean algorithm [4], and discovered that classifying pixels for one-tenth of our image took over two hours to compute. This was judged excessive, and well beyond what we hoped for in classification.

3.5 Addressing Issue Two: Edge Detection

The idea here is that near a linear feature in the original two-dimensional image, such as along an edge, the local spectra should not vary much from pixel to pixel. Consequently, it should be possible to identify correlations between spectral peaks as one moves from pixel to pixel. To test this idea, one can begin with the peak identification algorithm in Section 3.4, and apply some statistical tests for correlations.

Unfortunately, one week was not long enough to allow us to get both the peak detection algorithm working and statistically analyse the results. A larger database of seismic images and corresponding S-transforms would be required to pursue this avenue of research.

It is worth noting that the local spectral information of the S-transform varies slowly from pixel to pixel, so there should be quite a high level of correlation. What is required is a statistical test that can distinguish correlations that come from data with those that are merely a reflection of the slowly changing nature of the S-transform.

There were also suggestions to use some alternatives of S-transform. In particular, the application of methods which identify local spectra not at each individual pixel but from a group of adjacent pixels. Some adaptive methods were proposed as follows:

- The localized Fourier method. This method also has an adaptive version which can be applied to various time grids.

- Certain wavelet-like techniques for directional analysis, such as brushlets, ridgelets, or curvelets [8]. These wavelet bases were originally developed for directional image processing and identifying slanted patterns. Such techniques seem directly applicable to these seismic images.

3.6 Addressing Issue Three: Computing the S-transform

This was probably the most challenging issue to address. A direct implementation of the S-transform is highly redundant and leads to a massive amount of data, as each data point is transformed into a full spectrum. For instance, a simple 100 by 100 image transforms into 100 million data points $(100 \times 100)^2$. The company does have some impressively fast algorithms for computing an approximation to the complete transform, but with this level of redundancy, the question must be asked: are there better approaches? We did not have access to their implementation of the algorithm, so direct investigation of how it was implemented, or how it could be improved, were not possible. Some possible ideas to pursue include:

- compute a sparse subset of the transform and apply interpolation for the intermediate points;

- compute a sparse subset of the transform and use this directly to estimate peaks;

- investigate if an analog of a fast transform (FFT, wavelet, etc.) could be applied to the S-transform. In particular, the S-transform looks very similar to a phase-shifted continuous wavelet transform, so there may be some close connections to the wavelet speedup;

- consider using a different transform, such as the Gabor or wavelet transforms, which also produce local spectral information, but have a fast implementation.

Unfortunately, we did not have time to investigate these approaches in detail.

3.7 Additional Issues: Stepping Back to Seismic

In the end, this problem is really about seismic data imaging and analysis. It is important to realize that in all the images so created, there are the important physical aspects that should be properly modelled. That is, the images come from seismic experiments involving wave propagation; errors in the image come from errors in our physical model; the geological structures we hope to identify come from physical processes such as sedimentary deposition, geological faulting, and so on. It is a bit frustrating to begin the mathematical analysis at the end of the data processing, where we have an image that came from somewhere, and we want to analyse it. With sufficient mathematical and computational tools, it makes sense to go back to the original seismic data, and apply our research techniques to understand how this raw data can be used to reveal more accurately the features we seek to identify. This is a big task, not easily accomplished in a week-long problem solving workshop. An interested reader could refer to the work of the POTSI project (`potsi.math.ucalgary.ca`) or the CREWES project (`www.crewes.org`) for further information on research in the mathematics and geophysics of seismic imaging.

3.8 Acknowledgments

We thank Peirre Lemire and Rob Pinnegar of Calgary Scientific Inc. for all their help and guidance during our weeklong project. We also thank the Pacific Institute for Mathematical Sciences for their support of the Industrial Problem Solving Workshop.

3.9 Appendix

We include here some of our key MATLAB routines.

The main script for identifying peaks, in 2D and 3D

```
% PIXEL -- A script program for picking-up peaks in a local spectrum
%          created by the S-Transforms and removing them
%
%      By Yongwang Ma
%      University of Calgary, Alberta, Canada
%      May 20, 2005
%
%      You may copy, modify and distribute the codes for purpose of
%      research.
%

% process and plot in 2D
```

```
load SH02_LocalSpec_117_177.mat;
dimension = 2;
npick = 10; % number of pairs of peaks to remove
[F,nmovie] = pick(LocalSpec,npick,dimension);

% process and plot in 3D

load SH02_LocalSpec_117_177.mat;
dimension = 3;
npick = 10; % number of pairs of peaks to remove
[F,nmovie] = pick(LocalSpec,npick,dimension);
```

The subtractional subroutine

```
function [F,nmovie] = pick(SparseLS,npick,dimension)
% pick - finds peaks and creates humps (gaussian-like function in 2D)
%   in an image (spectrum) and substracts it from the original image
%
%   By Yongwang Ma, May 18, 2005
%      The University of Calgary
%      Calgary, Alberta, Canada
%
% Input:
% SparseLS = input image, a matrix
% npick = the number of pairs of peaks needs to be found
% dimension = dimension of final images (2- 2D images; 3-3D images)
%
% Output:
% F = a structure consisting of parameters of the frames for a movie
% nmovies = a vector storing the order of the images (figures)

pix_mat = SparseLS;
NF_half = floor(size(pix_mat,1)/2);
NK_half = floor(size(pix_mat,2)/2);
f = [-NF_half:NF_half];
k = [-NK_half:NK_half];
M = size(pix_mat,1);
N = size(pix_mat,2);
sig = 1.50; % parameter controlling the width of Gaussian-like window
figure; % using only one figure, so finally only the most updated image is sho
for ipeak = 1:npick
    pmax = max(max(pix_mat));
    % find the position of maximum (peak)  in the spectrum
    for m = 1:M
        for n = 1:N
            temp = pix_mat(m,n);
            if (temp == pmax)
                f0 = m; k0 = n;
            end
        end
    end
    % record the coordinates of peaks identified
    fpeak(ipeak) = f0;
    kpeak(ipeak) = k0;
    % transform the coordinates
    if (f0 >= NF_half)
        f_0 = f0-NF_half;
    else
        f_0 = -(NF_half-f0);
    end
    if (k0 >= NK_half)
        k_0 = k0-NK_half;
    else
        k_0 = -(NK_half-k0);
    end
    f_01 = - f_0;
    k_01 = -k_0;
    gaus_p = gaus1(f_0,k_0,M,N,k,f,sig);
    gaus_p1 = gaus1(f_01,k_01,M,N,k,f,sig);
    sub_pmat = pix_mat-gaus_p.*pix_mat;
    sub_pmat1 = sub_pmat-gaus_p1.*sub_pmat;

    % Plot out results of picking
    if (dimension ==2)
        %figure;
        imagesc(sub_pmat1);
    else % (3D)
        %figure;
        surf(k,f,sub_pmat1);
        axis([k(1) k(end) f(1) f(end) 0 3])
        shading flat
    end
    if ipeak==1
        boundz=get(gca,’CLim’);
    end
    set(gca,’CLim’,boundz);
%     figure;imagesc(gaus_p);
%     figure;
%     amat=gaus_p.*pix_mat;
%     plot(amat(f0,:));
%     hold;
%     plot(pix_mat(f0,:));
%     plot(sub_pmat(f0,:));
    pix_mat = sub_pmat1;
    F(ipeak) = getframe;
end
% create a movie
nmovie = zeros(1,24*npick);
nrmovie = zeros(1,24*npick);
nn1 = 1;
nn2 = 24;
nnr1 = 24*npick;
nnr2 = 24*(npick-1)+1;
for in = 1:npick
    nmovie(1,nn1:nn2) = in;
    nrmovie(1,nnr1:-1:nnr2) = in;
    nn1 = nn1+24;
    nn2 = nn2 + 24;
    nnr1 = nnr1-24;
    nnr2 = nnr2-24;
end
%play the movie forwards
movie(F,nmovie);
% play the movie backwards
% movie(F,nrmovie);
```

The subroutine to compute the Gaussian-like function

```
function gaus = gaus1(f0,k0,M,N,k,f,sig)

% gaus1 = subroutine to calculate a gaussian_like function (2D)
%
% f0 = frequency, coordinate at which the centre of gaussian-like function
% located
% k0 = wavenumber, coordinate at which the centre of gaussian-like
% funcition located
% M = total number of frequency samples
% N = totlal number of wavenumber samples
% k = wavenumber coordinates, a row vector
% f = frequency coordinates, a row vector
% sig = a factor, which determines the width of the gaussian-like function
```

```
p11 = zeros(M,N);
p22 = zeros(M,N);
stab = 1.e-8; % stability factor
for m = 1:M
    p11(m,:) = (f(m)-f0).ˆ2/Mˆ2+(k-k0).ˆ2/Nˆ2;
    p22(m,:) = (f(m)/M)ˆ2 + (k/N).ˆ2;
end
gaus = exp(-2*piˆ2*(p11./(p22+stab))/(sigˆ2));
```

Bibliography

[1] Aki, K. & Richards, P.G. (2002). *Quantitative Seismology* (2nd edition). University Science Book: Sausilito.

[2] Averbuch, A.Z., Meyer, F.G., Strömberg, J.O., Coifman, R.R. & Vassiliou, A. (2001). *Low Bit-Rate Efficient Compression for Seismic Data*. IEEE Transactions on Image Processing, 10(12), pp. 1801-1814.

[3] Coifman R.R. & Meyer, Y. (1991). *Remarques sur l'analyse de Fourier à fenêtre*. Comptes Rendus de l'Académie des Sciences, Série I, 312(3), pp. 259-261.

[4] MacQueen, J.B. (1967). *Some methods for classification and analysis of multivariate observations*. In Proceedings of the Fifth Berkeley Symposium on Mathematical Statistics and Probability. Vol I, Statistics. Edited by Lucien M. Le Cam and Jerzy Neyman. University of California Press, pp. 281-297.

[5] Meyer F.G. & Coifman, R.R. (1997). *Brushlets: a tool for directional image analysis and image compression*. Applied and Computational Harmonic Analysis, 4, pp. 147-187.

[6] Press, W.H., Flannery, B.P., Teukolsky, S.A. & Vettering, W.T. (1992). *Numerical Recipes in C* (2nd edition). Cambridge University Press: Cambridge.

[7] Stockwell, R.G., Mansinha, L. & Lowe, R.P. (1996). *Localization of the Complex Spectrum: The S Transform*. IEEE Transactions on Signal Processing, 44(4), pp. 998-1001.

[8] Welland, G.V. (2003). Studies in Computational Mathematics, Vol 10: *Beyond Wavelets*. Academic Press.

Chapter 4

Adaptive Statistical Evaluation Tools for Equity Ranking Models

Problem presented by: Brad Bondy (Genus Capital Management)

Mentors: Tony Ware (University of Calgary), Len Bos (University of Calgary)

Student Participants: Amir Amiraslani (University of Western Ontario), Thomas Holloway (University of Alberta), Hua Li (University of Calgary), Maryam Mizani (University of Victoria), Mahyar Mohajer (University of Calgary), Comron Nouri (University of Western Ontario), Gergeli Orosi (University of Calgary), Nargol Rezvani (University of Western Ontario), Liang Xu (University of Washington), Oulu Xu (York University)

Report prepared by: Tony Ware (aware@ucalgary.ca)

4.1 Introduction

Based in Vancouver, Genus Capital Management (http://www.genuscap.com) is an independent investment provider who for over 15 years have been managing assets for private individuals and families, trusts, foundations and pension funds from across Canada. Their investments span a range of equity and fixed income, and they manage assets to the value of about $1.3-billion.

Genus offer investment portfolios with a variety of flavours. Each portfolio consists of investments in stocks from some given 'universe'. Competitive advantage comes from consistently out-performing competing funds. One of the approaches that Genus use to achieve competitive advantage is to make use of auxiliary information about the stocks in the universe in deciding how to adjust their portfolio month by month. The problem that Genus brought to the IPSW was concerned with how to improve the way this information is utilised.

4.2 Problem Description

A major challenge in the investment management business is to identify which stocks are likely to outperform in the future, and which are likely to perform relatively poorly. To this end the strategy adopted by Genus is to identify *factors* (auxiliary information about the stock such as earnings-to-price ratio or dividend yield) that they believe are associated with future out-performance (i.e. factors that have predictive ability). The best of these factors are then combined (Genus use a weighted average) into a model which is used to rank the universe of stocks month-by-month. This ranking is then used to as the input to a trading strategy, resulting in a modified portfolio.

A critical step in this process is to identify the ‘good’ factors and to determine how heavily each should be weighted in the final model. Currently Genus use in-sample back-tests in which they simulate a trading strategy of, for example, selling stocks that drop below the 40th percentile (of the ranking) and reinvesting the proceeds in stocks in the top 20th percentile. These simulations are run over a wide range of potential weighting schemes or models, and the model with the most attractive attributes is identified. Attributes that Genus typically look for in a model include:

- High *spread* between returns for top and bottom quintiles.

- High *information ratio*: the ratio of the mean and standard deviation of the excess monthly return on portfolio versus the index (a portfolio in which all the stocks in the universe are equally-weighted).

- High *hit ratio*: the percentage of months in which the portfolio outperforms the index.

The in-sample back-tests are typically run over a 10-year period, excluding the most recent 2 years. Once a model is identified, out-of-sample tests of the model over the most recent 2 years are run in order to ensure that the model works out-of-sample.

A major issue for Genus is the inflexibility that results from the 2-year delay for out-of-sample testing. Factors tend to lose their predictive ability as other market participants incorporate them, and useful new factors seem to be getting cleaned out faster and faster. With the 2-year delay Genus run the risk of delaying introduction of a new factor until the market has already cleaned it out. On the other hand, if the out-of-sample tests are eliminated, they run the risk of over-fitting the data.

The challenge posed for the workshop team was:

1. to recommend adaptive statistical evaluation tools (alternatives to out-of-sample tests) that could be used to improve confidence in a model and to help decide in a timely fashion if a new factor should be added to a model, or if an existing factor should be removed;

2. to suggest algorithms that could be used to dynamically update the models with a view to exploring a dynamic strategy in which model factors and weights are updated monthly based on the evaluation measures.

4.3 Rising to the Challenge

It was clear from the outset that if the team were to be able to address either of these challenges realistically they would need to build tools that would enable them to implement the various components of Genus’ portfolio management strategy. As the week progressed, it became apparent that a major stumbling block would be implementation of an efficient procedure for finding a ‘good’ model for ranking the stocks. This became the focus of the team’s efforts.

Genus had provided the team with sample data, consisting of just over 12 years worth of monthly returns on a universe of 60 stocks, along with time series of 34 factors for each of the stocks. Using these data, the approach was to build software (MATLAB) models for:

- ranking the stocks based on factor information;
- implementing a trading strategy based on a stock ranking and assessing the performance of a given trading strategy by looking at measures such as hit ratio, information ratio and spread.

The IPSW team implemented a simplified trading strategy of selling the entire portfolio each month, and using the proceeds to invest equally in the top 20% of stocks as given by the computed ranking. They also implemented the following measures of portfolio performance: excess return, hit ratio and information ratio.

4.3.1 Ranking

As noted above, determining an an effective procedure for ranking the stocks is a critical step, but the design of such a procedure cannot be separated from the other tasks, and in particular the choice of trading strategy, or performance assessment measures.

A simplified description of the approach taken by Genus is the following. Given a set of factor values $f_{i,j}(t)$ for the jth factor corresponding to ith stock at time t, and given a linear weighting vector $\vec{w} = (w_j)$, they produce a ranking at each time t from the score vector

$$S_i(t;w) := \sum_j w_j f_{i,j}(t).$$

A trading strategy, such as buying the top quintile of stocks (those obtaining the top 20% scores) in equal measure, is assumed (the actual strategy adopted by Genus is rather more complicated than this, but the details were not available to the IPSW team). The trading strategy takes the weighting vector $\vec{w}$ and the factor values for a given date and produces a set of portfolio weights $\alpha_i(t)$. The actual stock returns $R_i(t)$ can then be used to compute the portfolio return

$$\sum_i \alpha_i(t) R_i(t).$$

Given such a strategy, the performance of the model can be assessed by measuring the resulting excess portfolio returns, the information ratio, the hit ratio, and other parameters.

In order to determine a 'good' weighting vector $\vec{w}$, Genus select a training period (typically a ten-year period of time excluding the most recent two years). They construct a grid of possible weighting vectors, spanning a subset of the total set of possibilities; for each vector in the grid they produce rankings for each date in the training period, and (using the trading strategy) compute the corresponding portfolio excess returns, hit ratios, information ratios, etc. They then select the best-performing weighting vector from this sample and test it against the most recent two-year period before putting it to use.

The determination of a ranking model is an optimisation problem, with a highly nonlinear objective function (irrespective of which of the portfolio measures is used). The IPSW team decided that there could be significant advantages to be gained from exploring alternative approaches to the design of optimal ranking algorithms. They considered three distinct techniques: a genetic optimisation algorithm, a neural network, and a constrained least-squares approach.

4.3.2 Genetic Optimization

Genetic optimization is an evolutionary search technique (see [2] for more information), in which a population of abstract representations of candidate solutions (*individuals*) evolves towards better solutions. The evolution begins with a population of randomly selected individuals and proceeds in *generations*. The transition from generation to generation involves evaluating the *fitness* of the current population; individuals are randomly selected for *reproduction* (with a probability based on their fitness). Individuals selected for reproduction are randomly selected for *mating* (each pair produces two offspring, each of which has some combination of the features of their parents). In addition, each feature of the offspring can mutate independently with a given probability.

Parameters and Results

Genetic optimisation was used to search for an optimum weight vector $\vec{w}$ for use in a linear ranking model as described above. The first 113 months of factor and return data were used to construct the objective function, which took a weighting vector as input and gave the resulting excess returns as output. The populations consisted of 30 individuals, with a 0.01 probability of mutation of the offspring features, and the algorithm was allowed to evolve for 1000 generations before being terminated. The resulting weight vector $\vec{w}_0$ is shown in Figure 4.1.

In Figure 4.2, the objective function in the neighbourhood of $\vec{w}_0$ is illustrated. Each of the graphs in the figure depicts a 'slice' of the objective function in the direction of one of the 34 coordinate directions, centred at $\vec{w}_0$. The non-smooth nature of the objective function can be readily seen, as is the fact that the genetic algorithm has attained something close to, but not exactly at, a maximum.

4.3.3 Neural Network

An artificial neural network (see [3] for more information) is a computational model for information processing, represented by a nonlinear function ν that maps a vector of input values $\vec{x}$ to a vector of output values $\vec{y}$. The nonlinear function has a particular form given by a network structure together with weights associated with the network connections. A feed forward network typically has an input layer, with a node for each element of $\vec{x}$, an output layer, with a node for each element of $\vec{y}$, and one or more hidden layers in between. Each node in a given layer can be connected to each node in the next layer.

The nodes in the input layer do no processing: they simply pass on the input values to the nodes in the next layer to which they are connected.

Nodes in the hidden layers receive values from the incoming connections; they process these values in some way before determining a value to pass on to the nodes in the next layer. A common choice for computing the output value is to compute a weighted linear combination of the inputs (the weights being those associated with the incoming connections) and to apply a logistic function, for example $\psi(s) = \frac{1}{1+\mathrm{e}^{-\mathrm{s}}}$, to the result. Nodes in the output layer compute the elements of the vector $\vec{y}$ by computing the weighted combination of their inputs. A network of this sort is an example of a *multi-layer perceptron*.

Even with just one hidden layer, such a network has a universal approximation property, and can—with an appropriate choice of connection weights—reproduce arbitrarily closely the action of any continuous function that maps intervals of real numbers to intervals of real numbers [1].

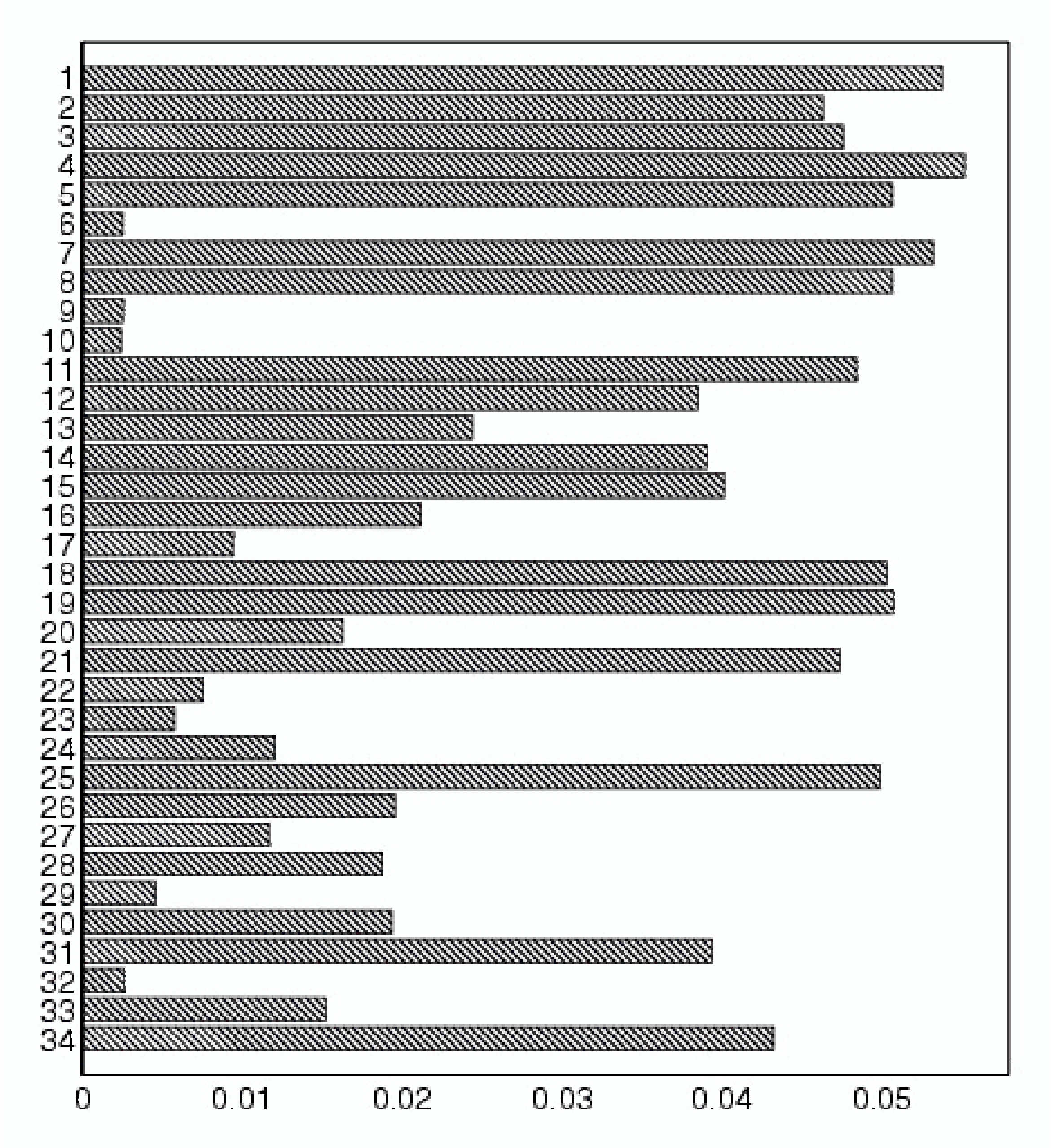


Figure 4.1: Factor weights from the genetic algorithm.

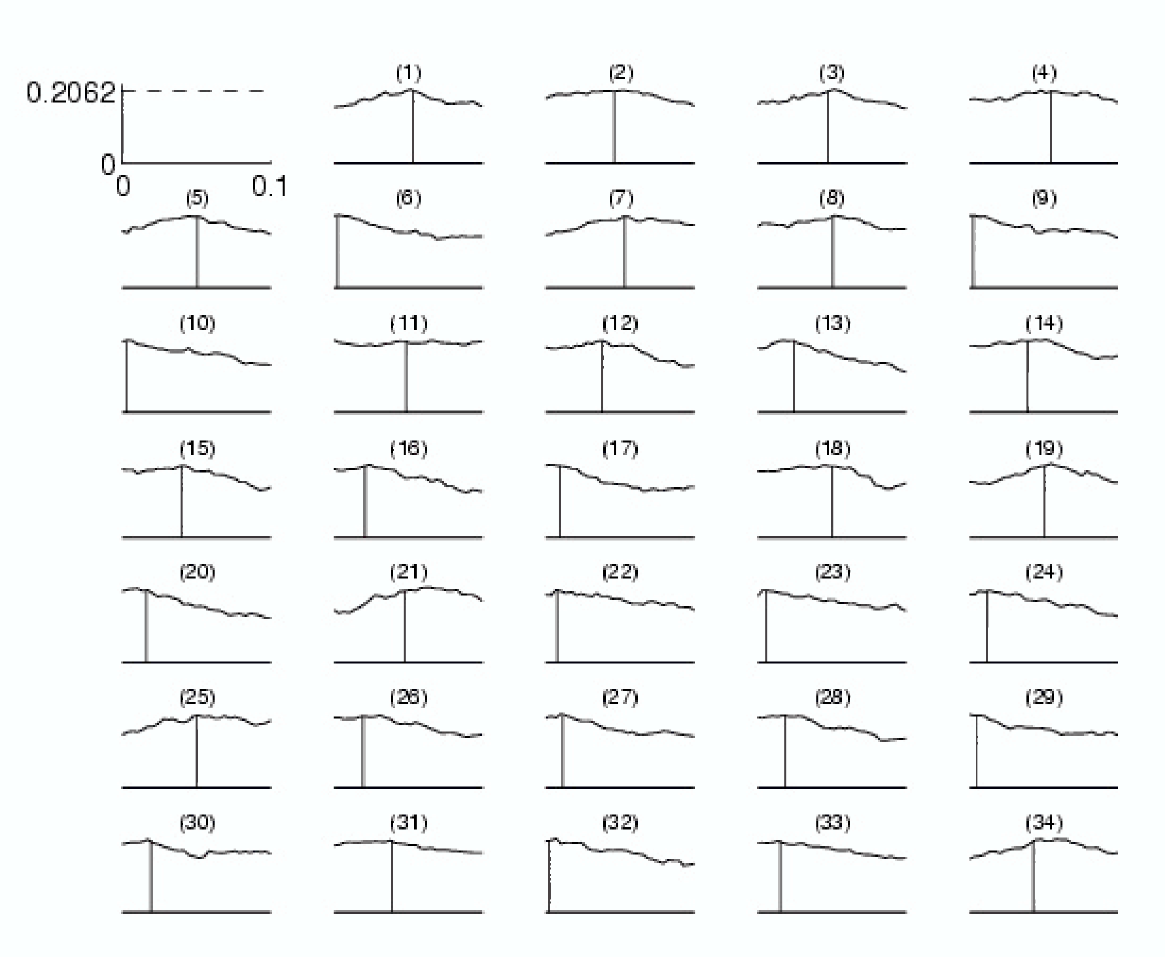


Figure 4.2: Slice view of the 34-dimensional objective function near the chosen maximum for the genetic optimisation. The first graph shows the limits of the weight values, as well as the value of the objective function at $\vec{w}_0$. In each of the other graphs the vertical line denotes the value of the corresponding component of $\vec{w}_0$.

In order for the network to be of any practical use, the connection weights need to be determined. A training set of inputs and corresponding outputs is used for this purpose. One popular choice for adapting the network to the training set is *back-propagation*, which is an implementation of a gradient-descent approach to minimising the sum of the squares of the differences between the given outputs and the outputs from the network.

Parameters

A single-layer perceptron was employed, with the hidden layer having just two nodes. The network was allowed to have six inputs (six of the the possible thirty-four factors were selected) and one output: the stock return. The input factors were chosen by determining which of the factors were most highly correlated with the subsequent months' returns.

The training data set was selected to be the first 113 periods. Back-propagation, with approximately 1000 iterations, and a learning rate of 0.1 (the distance moved in the direction of the gradient vector at each iteration), was used to determine the weights.

Once the network was trained it was used to produce forecasted returns on stocks for each of the remaining 36 months. In each month, the forecast returns were used to rank the stocks, and the trading strategy of investing in the top 20% used to construct a portfolio for that month.

4.3.4 Constrained Least Squares Optimization

A third approach was also used; as with the genetic optimisation, an optimal weight vector $\vec{w}$ was sought, with the entries constrained to be positive. Subject to this constraint, the weights were chosen to minimise the sum of squared discrepancies between the individual stock returns[1] and the corresponding score vector for each time t in the first 113 months:

$$\sum_{t=1}^{113}\sum_{i}(S_i(t;w) - R_i(t))^2.$$

The positive weight vector that minimised this error term was computed using MATLAB's `quadprog` routine, which will compute the minimum of a quadratic function subject to bounds on the input variables [4]. The resulting weight vector is shown in Figure 4.3.

4.4 Results

Each of the resulting models were applied to the most recent 36 months of data (i.e. not the data used in determining the models). The performances of their portfolios are illustrated in Figure 4.4, where the portfolio returns are shown in raw form in the upper graph, along with the performance of an index portfolio. In the lower graph, the excess returns, relative to the index, are shown. Also, in Figure 4.5, the values of several other measures of portfolio performance are shown.

[1] Other objectives could be designed: for example, the ranking of the stock returns could be used.

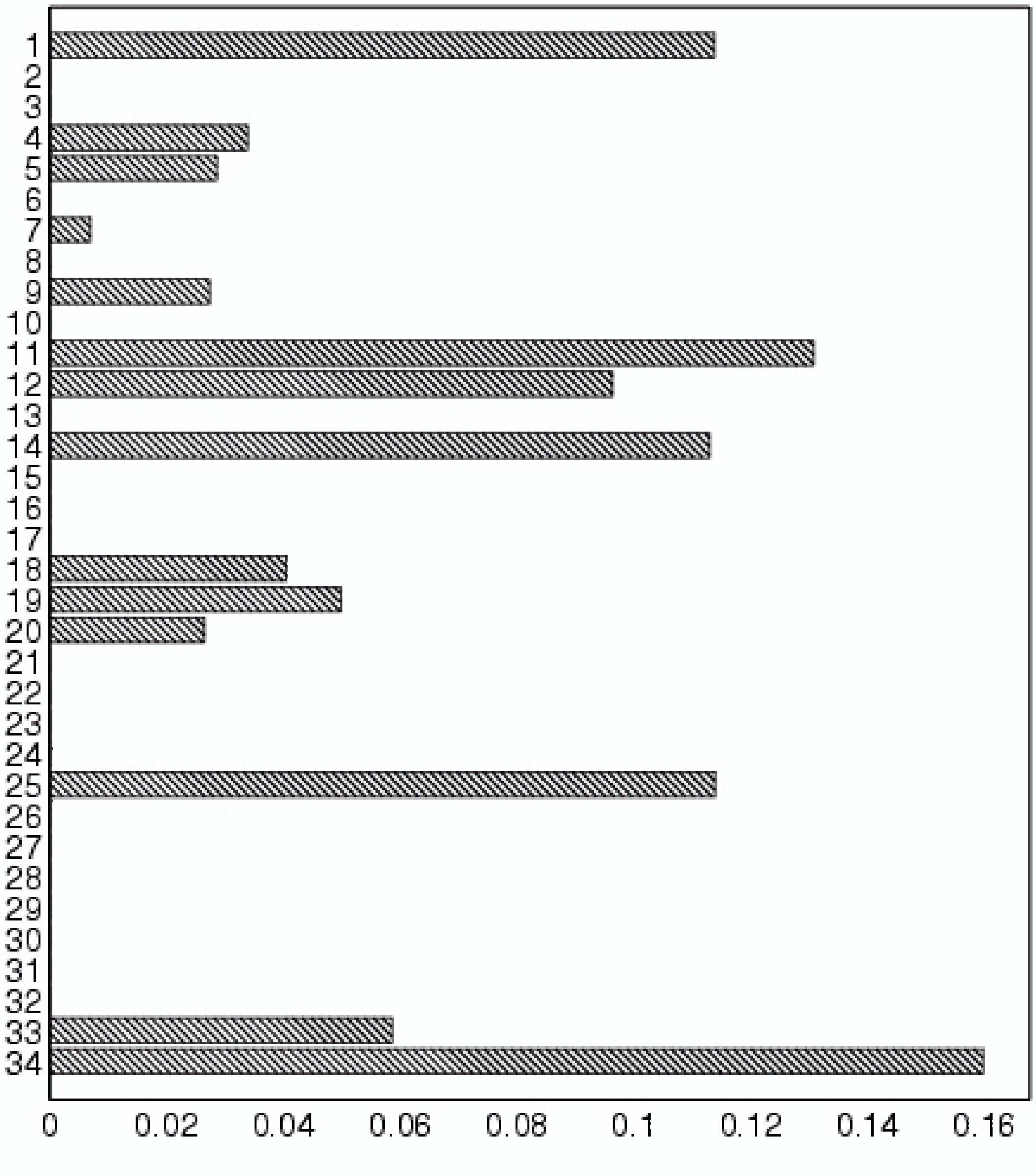


Figure 4.3: Factor weights from the constrained least squares optimisation.

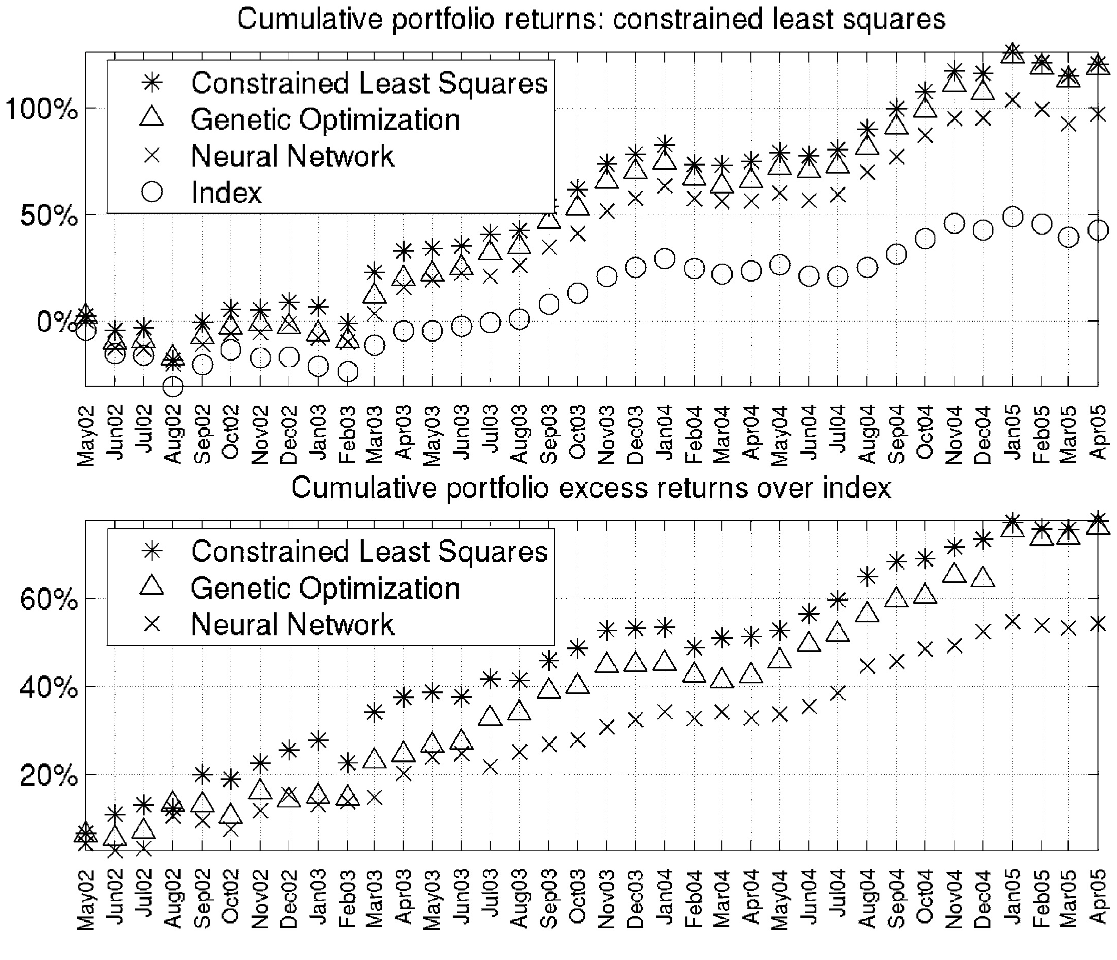


Figure 4.4: Three-year results from the constrained least-squares model, the genetic optimisation model, and the neural network.

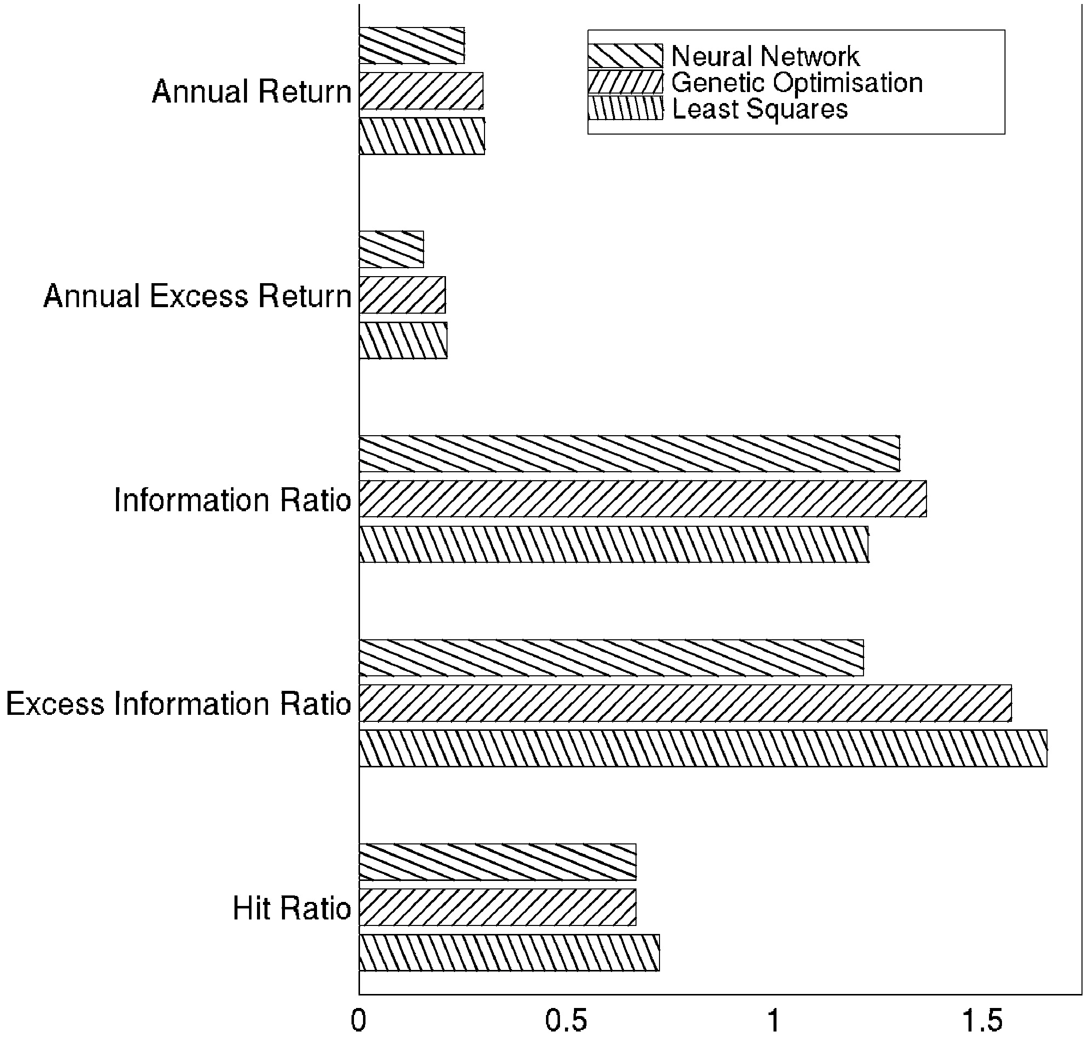


Figure 4.5: Values of Hit Ratio, Excess Information Ratio, Excess Information Ratio (using portfolio returns relative to the index), Annual Excess Return and Annual Return for the portfolios generated by the three different ranking models.

4.5 Conclusions and Directions for Further Work

The results from the three ranking models shown in the previous section are illustrative only. Each of the approaches is capable of refinement. However, the results obtained—even without refinement—compare well with the performance of Genus' current model. Each of the approaches has the potential to deliver significant improvements for Genus. Moreover, especially in the case of the constrained least-squares optimisation, they can be implemented much more quickly. Future work might involve:

- refinement of the neural network and genetic optimisation approaches: each offers a vast array of possible implementations and the ones presented here can certainly be improved upon;
- experimentation with alternative objective functions for the constrained least squares (quadratic programming) approach;
- addressing Genus' original challenge. . . .

Bibliography

[1] Hornik, K., Stinchcombe, M. & White, H. (1989). Multi-layer feedforward networks are universal approximators, *Neural Networks*, 2, pp. 359-366.

[2] Bäck, T. (1996). Evolutionary Algorithms in Theory and Practice. Oxford University Press.

[3] Rojas, R. (1996). Neural Networks: A Systematic Introduction. Springer-Verlag.

[4] Coleman, T.F., & Li, Y. (1996). A reflective Newton method for minimizing a quadratic function subject to bounds on some of the variables, *SIAM Journal on Optimization*, 6(4), pp. 1040-1058.

Chapter 5

Seismic Prediction of Reservoir Parameters

Problem presented by: Brian Russell (Hampson Russell Software)

Mentors: Len Bos (University of Calgary), Gemai Chen (University of Calgary), Lou Fishman (MDF International, University of Calgary)

Student Participants: Mahmud Akelbek (University of Regina), Mark Braverman (University of Toronto), Sandra Fital (University of Regina), Yaling Yin (University of Saskatchewan), Zhidong Zhang (University of Saskatchewan)

Report prepared by: Yaling Yin (yin@snoopy.usask.ca)

5.1 Introduction

In seismic analysis, our goal is to determine the properties of the subsurface of the earth using seismic measurements made on the surface of the earth. Seismic measurements are made by sending elastic waves into the earth (by setting off an artificial explosion at the earth's surface) and recording the reflected elastic waves at the earth's surface (using geophones on land and hydrophones offshore). This is illustrated in Figure 5.1. Although this technique has been practiced since the beginning of the twentieth century, the major advances in seismic analysis have occurred during the last fifty years, and were spurred by the joint development of digital recording and the digital computer. Most of the techniques that have been developed for the analysis of seismic data can be classified as deterministic. That is, we develop a deterministic physical model which relates the material properties of the subsurface of the earth to the physics of the transmitted and reflected seismic waves.

One such deterministic model is called the convolutional model. In the convolutional model, we assume that the amplitudes of the seismic reflections are directly related to the changes in impedance (the product of velocity and density) of the various geological formations below the surface, convolved with a "wavelet" that represents the oscillations caused by the seismic source. By "de-convolving" and inverting the seismic recording, we can therefore derive both the velocity (which can be either compressional or shear) and density distributions of the earth's subsurface. From these velocity and density values, we can then infer other properties of the subsurface, such as the porosity and fluid content of the seismic reservoir. This inversion is illustrated in Figure 5.2 for the Blackfoot case study.

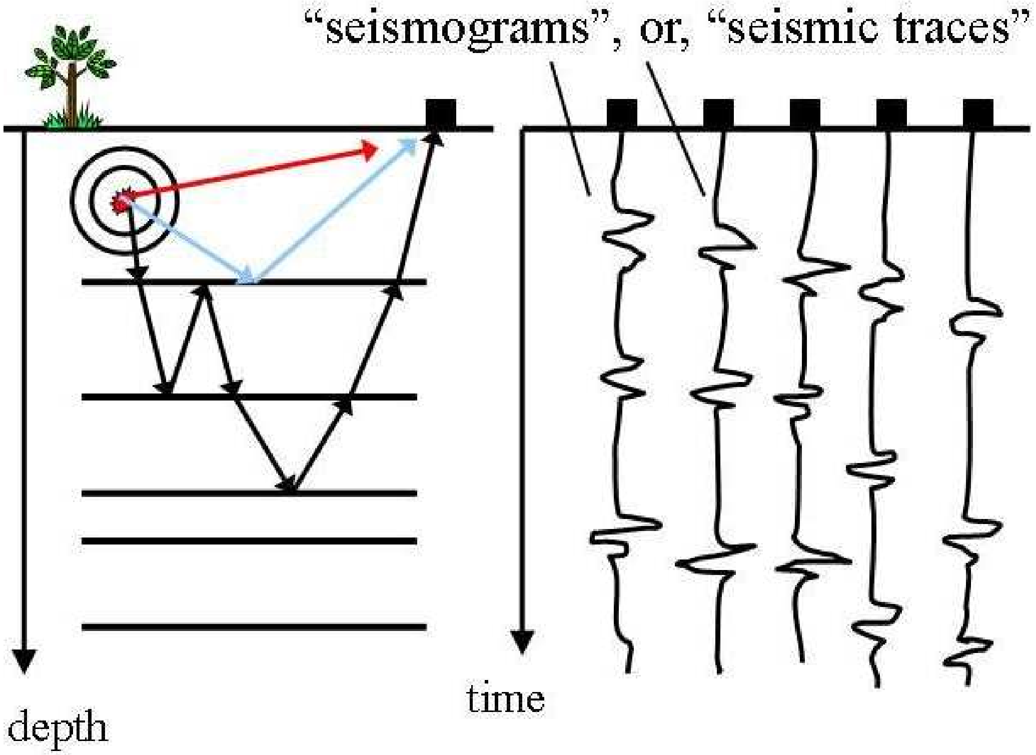


Figure 5.1: Elastic waves are sent into the earth and the reflected signals from the subsurface layers are recorded on the surface as a function of time.

More recently, researchers have started using a statistical, rather than deterministic, approach to the determination of the earth's subsurface and its reservoir parameters. That is one of the main approaches that we will use here. Thus, instead of assuming an underlying model, we will use multivariate statistical techniques to determine the earth's subsurface, using a set of derived "attributes" from the seismic data. This mathematical approach is described in Section 5.2. In Section 5.3, the previous statistical approaches to addressing this problem are reviewed and evaluated. Sections 5.4 and 5.5 describe and discuss two approaches that were developed in the IPSW. Section 5.4 considers a statistical approach, while Section 5.5 addresses spline approximation. Finally, Section 5.6 provides a general summary and discussion.

5.2 Mathematical Statement of the Problem

We assume that we have N, M-dimensional multivariate observations which can be written as $x_j = (x_{1j}, x_{2j}, \ldots, x_{Mj})^T$, $j = 1, 2, \ldots, N$, and that we have N scalar training values t_j, $j = 1, 2, \ldots, N$. Our objective is to find some linear or nonlinear scalar function y such that

$$y(x_j) = t_j, \qquad j = 1, 2, \ldots, N. \tag{5.1}$$

In our case, the multivariate observations are a set of seismic attribute values at a given depth or

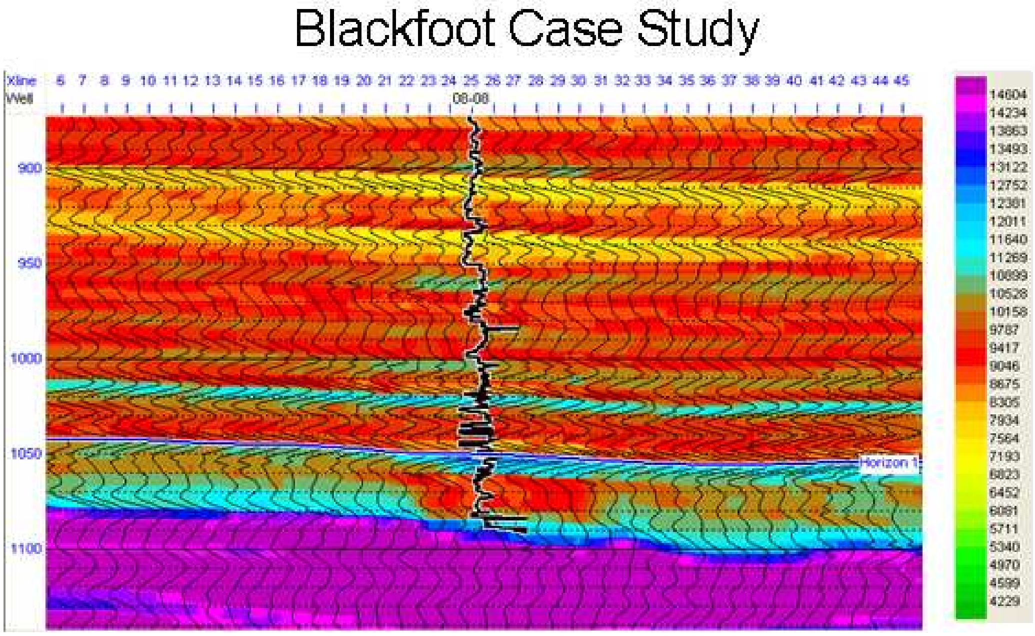


Figure 5.2: Here is the inversion of a seismic line from the survey, showing a channel sand at a time of 1075 ms and the intersection of well 08-08. The values are in impedance (velocity·density) and we wish to predict velocity.

time, and the training values are a set of well-log derived reservoir parameter values at the same depth or time. The well-logging process to obtain sets of high resolution values is illustrated in Figure 5.3. The physical situation is illustrated in Figure 5.4 for $M = 3$. Note from Figure 5.4 that each of the M attributes can be thought of as a N-dimensional vector function $a_i = (a_{1i}, a_{2i}, \ldots, a_{Ni})^T$, $i = 1, 2, \ldots, M$. The attributes were chosen from a set of over 50 attributes by multilinear regression.

Our objective function for success is that the answer y is as close to t as possible using a least-squares criterion. However, since we can make the error as small as we want by adding more model complexity, the least-squares error is computed using cross-validation. That is, the N training points are sub-divided into K separate classes (in our case, the K individual wells), of lengths $N_1, N_2, \ldots, N_K$. We then leave out each of the K subsets of points in turn, and use the points in the other $K - 1$ wells to predict the removed wells. Next, we compute the least-squared error between the known values and the predicted values. The final error is the average error for the K individual cases.

5.3 Previous Approaches

The following approaches had already been applied to the solution of the problem:

1. The standard multilinear regression approach.

2. The Nadaraya-Watson estimator of multivariate statistics. This approach was re-discovered in the context of neural networks and named the generalized regression neural network (GRNN).

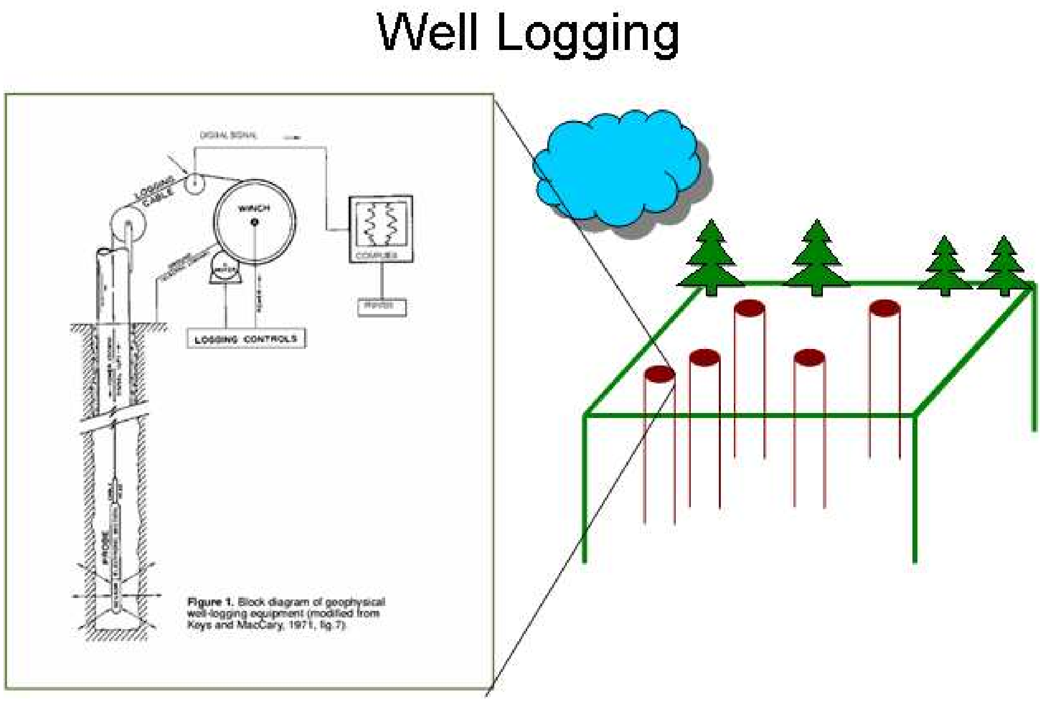

Well Logging
DIGITAL SIGNAL
LOGGING CABLE
WINCH
COMPUTER
PRINTER
LOGGING CONTROLS
PROBE
Figure 1. Block diagram of geophysical well-logging equipment (modified from Keys and MacCary, 1971, fig.7).

Figure 5.3: The well logging procedure directly obtains high resolution values of the desired parameters.

3. The radial basis function neural network (RBFN).

The multivariate dataset can be written $X = A^T$, where,

$$X = \begin{bmatrix}\mathbf{x}_1 \cdots \mathbf{x}_N\end{bmatrix} = \begin{bmatrix} x_{11} & \cdots & x_{1N} \\ \vdots & \ddots & \vdots \\ x_{M1} & \cdots & x_{MN}, \end{bmatrix}, \qquad A = \begin{bmatrix}\mathbf{a}_1 \cdots \mathbf{a}_M\end{bmatrix} = \begin{bmatrix} a_{11} & \cdots & a_{1M} \\ \vdots & \ddots & \vdots \\ a_{N1} & \cdots & a_{NM} \end{bmatrix}. \tag{5.2}$$

5.3.1 Multilinear Regression

Multilinear regression of the attributes against the training values involves solving for the weights in the equation,

$$\mathbf{t} = w_0\mathbf{a}_0 + w_1\mathbf{a}_1 + \cdots + w_M\mathbf{a}_M = \hat{A}\mathbf{w}, \quad \hat{A} = \begin{bmatrix} 1 & a_{11} & \cdots & a_{1M} \\ \vdots & \vdots & \ddots & \vdots \\ 1 & a_{N1} & \cdots & a_{NM} \end{bmatrix}, \quad \mathbf{w} = \begin{bmatrix} w_0 \\ \vdots \\ w_M \end{bmatrix}. \tag{5.3}$$

The solution is found by generalized least-squares to be

$$\mathbf{w} = (\hat{A}^T\hat{A} + \lambda I)^{-1}\hat{A}^T\mathbf{t}, \tag{5.4}$$

where λ is a prewhitening factor used to regularize the solution since $\hat{A}^T\hat{A}$ may not have a stable inverse. Also, too much prewhitening can create artifacts.

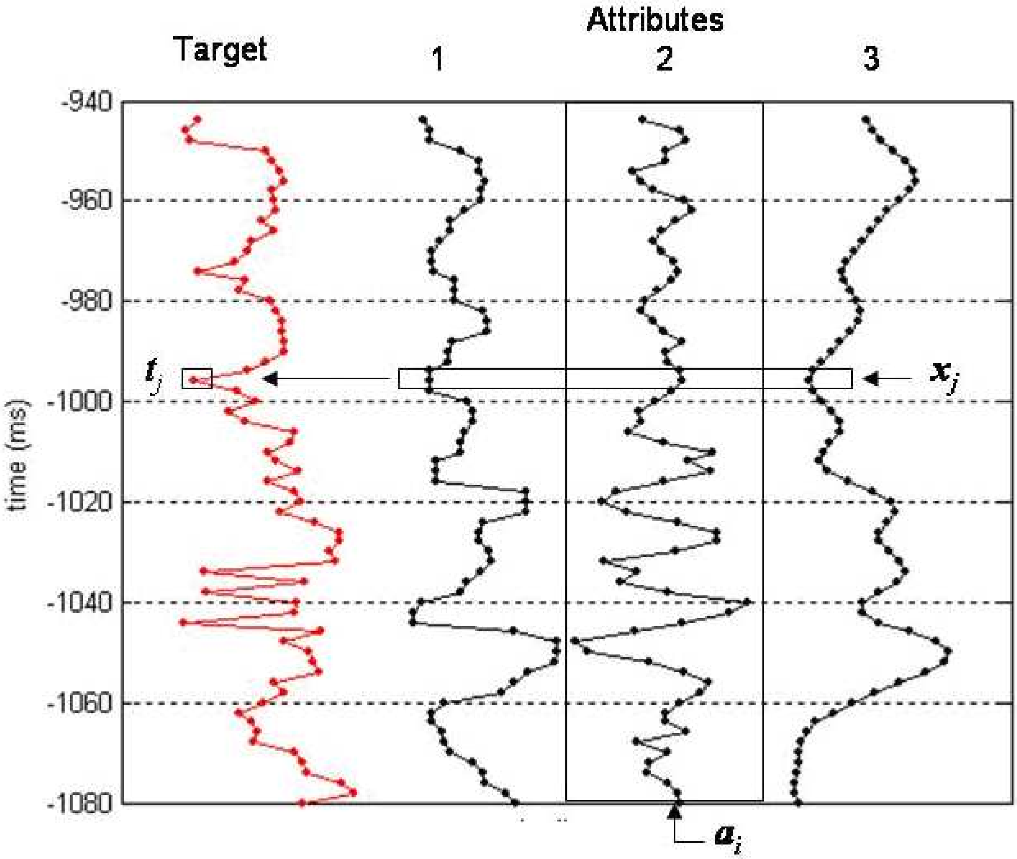


Figure 5.4: The basic prediction problem, where we want to predict t_j from x_j.

5.3.2 GRNN and RBFN

These two neural networks use basis functions. The basis functions are Gaussian functions of distance in attribute space, which can be written as

$$\phi_{ij} = \phi(d_{ij}) = \exp\left(-\frac{d_{ij}^2}{\sigma^2}\right), \qquad d_{ij} = |x_i - x_j|\,. \tag{5.5}$$

where σ is a smoothness parameter. The GRNN computes the predicted values "on the fly" from the training values, using the basis functions defined below,

$$f(x_k) = \frac{\sum_{j=1}^{N} t_j \phi_{kj}}{\sum_{j=1}^{N} \phi_{kj}}, \qquad k = 1, 2, \ldots, M. \tag{5.6}$$

In the RBFN, the computation of the predicted values is similar,

$$f(x_k) = \sum_{j=1}^{N} w_j \phi_{kj}, \qquad k = 1, 2, \ldots, M. \tag{5.7}$$

However, the weights are computed from the training data using the following linear equations,

$$t(\mathbf{x}_k) = \sum_{j=1}^{N} w_j \phi_{kj}, \qquad k = 1, 2, \ldots, N. \tag{5.8}$$

In both GRNN and RBFN, the key parameter to optimize is the σ value, which controls the width of the basis functions. For RBFN, we estimate a global value of σ. For GRNN, we optimize σ so that it varies as a function of the number of parameters in the multivariate observation, or

$$y(\mathbf{x}) = \frac{\sum_{i=1}^{N} t_i \exp\left[-\frac{(x_1-x_{i1})^2}{\sigma_1^2} - \cdots - \frac{(x_M-x_{iM})^2}{\sigma_M^2}\right]}{\sum_{i=1}^{N} \exp\left[-\frac{(x_1-x_{i1})^2}{\sigma_1^2} - \cdots - \frac{(x_M-x_{iM})^2}{\sigma_M^2}\right]}. \tag{5.9}$$

A comparison of GRNN and RBFN is as follows. The GRNN has a simpler structure and computes its weights directly from the training data without having to pre-compute these weights. The RBFN pre-computes the weights by solving a $N \times N$ matrix, where N is the number of training points. The two methods give similar results for large numbers of training points, but RBFN is superior when the number of points is small.

5.3.3 Results

Amongst the three methods, GRNN gives the best results having the highest average correlation coefficient between the prediction and the validation target of 12 wells, 0.633. Figure 5.5 shows the best result obtained by GRNN, with the correlation coefficient 0.74. It shows the target (red) and predicted (blue) values of velocity versus time of well 11. The worst case has the correlation coefficient 0.47, which is obtained from well 5, and it is shown in Figure 5.6.

5.3.4 Challenges

Thus we have the following challenges:

1. Can we find other methods that may be more accurate?
2. Can we understand the nature and limitations of the methods from our results?
3. Can we devise other more local estimation procedures?

To address these challenges, we have tried the generalized additive model method and the spline method. In next two sections, we will describe the two methods and show our results. Finally we will give the summary of our work in the last section.

5.4 Generalized Additive Model

5.4.1 Introduction of Generalized Additive Model

The Generalized Additive Model (GAM), a generalization of the linear regression model, can be viewed as the combination of the additive model with the generalized linear model. Suppose that

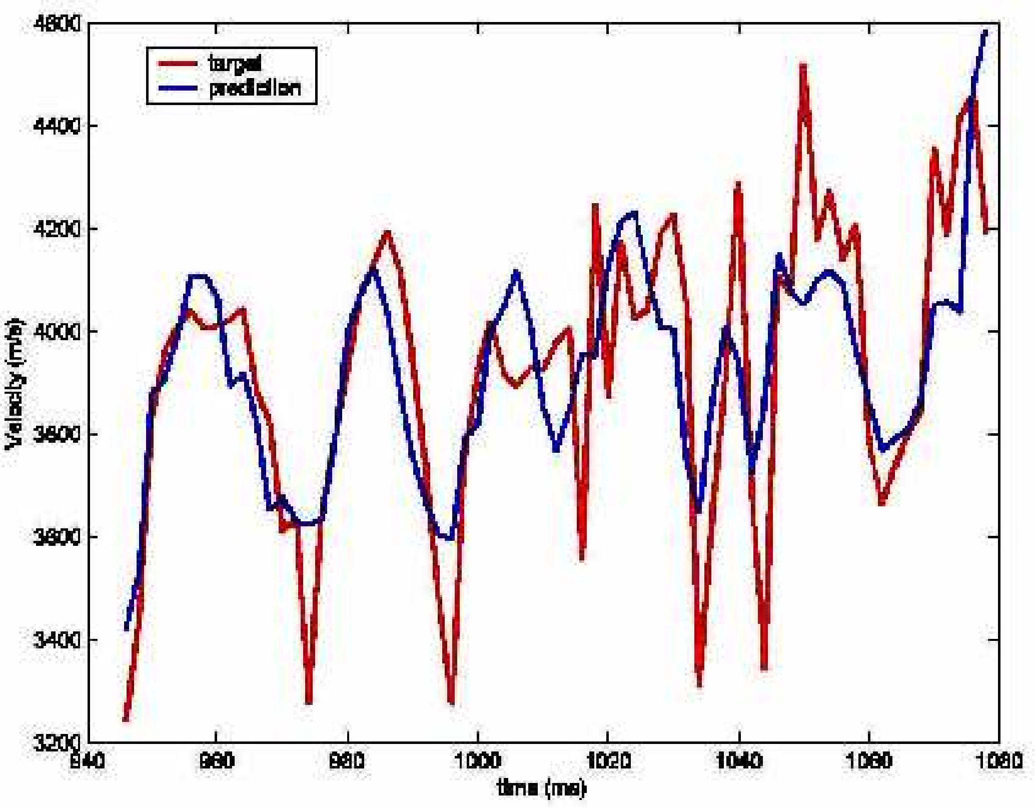


Figure 5.5: The true (red) and predicted (blue) values of velocity Y vs. time (Well 11–the best case).

Y is a dependent random variable and $X_1, X_2, \ldots, X_p$ is a set of independent random variables. The standard linear regression model assumes the expected value of Y has a linear form,

$$E(Y) = \alpha + \sum_{j=1}^{p} \beta_j X_j. \tag{5.10}$$

Therefore the linear regression model assumes the effect of X_j has a linear form $\beta_j X_j$ as well. Given a sample of values for Y and $X_1, X_2, \ldots, X_p$, the parameters β_j are often estimated by the least squares method.

The additive model relaxes the linear form $\beta_j X_j$ and assumes the effect of X_j has a general form $f_j(X_j)$, where $f_j(\cdot)$ is an unspecified (nonparametric) function. These functions are not given a parametric form but instead are estimated in a nonparametric fashion. However, the additive model still assumes the expected value of Y has an additive form,

$$E(Y) = \alpha + \sum_{j=1}^{p} f_j(X_j). \tag{5.11}$$

To extend the additive model to a wide range of distribution families, Hastie and Tibshirani [2] proposed the generalized additive model. This model assumes that the expected value of Y depends on

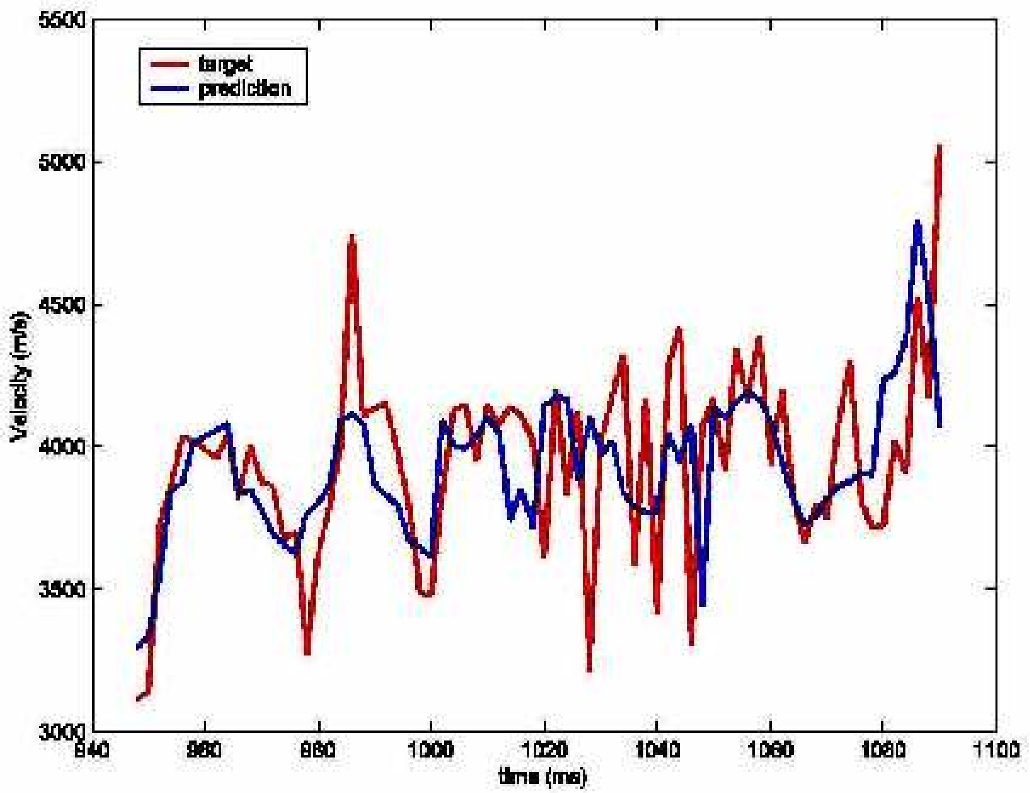


Figure 5.6: The true (red) and predicted (blue) values of velocity Y vs. time (Well 5–the worst case).

additive independent random variables X_j through a nonlinear link function $g(\cdot)$. GAM has the form,

$$g(E(Y)) = \alpha + \sum_{j=1}^{p} f_j(X_j). \tag{5.12}$$

5.4.2 Advantages of GAM

GAM has two main advantages. First, GAM is a kind of nonparametric regression and it relaxes the usual assumption of linearity, enabling users to reveal the hidden structure in the relationship between the independent variables and the dependent variable. Thus, GAM is more flexible than linear models. GAM includes the linear models as a special case, when g is the identity. Secondly, GAM allows for a link between $\sum_{j=1}^{p} f_j(X_j)$ and the expected value of Y. This amounts to allowing for an alternative distribution for Y besides the normal distribution. Actually it permits the dependent random variable Y to be any member of the exponential family of distributions.

5.4.3 Fitting GAM

In S-plus, the GAM is fit by the local scoring algorithm, which is based on the backfitting algorithm. The backfitting algorithm is a general algorithm that can estimate the smoothing terms $f_j(X_j)$ in the additive model using any nonparametric smoothers. The GAM procedure in S-plus has two primary smoothers: splines and Loess (Locally weighted regression). The backfitting algorithm is iterative, starting with initial functions $f_j(X_j)^0$, and in each iteration, fitting the $f_j(X_j)$ to its partial residuals. Iteration proceeds until $f_j(X_j)$ does not change. The local scoring algorithm is used to estimate the smoothing terms $f_j(X_j)$ in the GAM. It is also an iterative algorithm and starts with initial estimates of $f_j(X_j)$. During each iteration, an adjusted dependent variable and a set weight are computed, and $f_j(X_j)$ are estimated using a weighted backfitting algorithm. The scoring algorithm stops when the deviance of the estimates ceases to decrease.

5.4.4 Results

The predicted values of Y are computed using a cross-validation procedure. That is, 855 training points are divided into 12 wells. At each time, we leave out the data points of a given well, and use the points in the other 11 wells to predict of the removed well. Then we calculate the correlation coefficient of the true values and the predicted values in each well which are used to determine the quality of prediction, and finally obtain the average correlation coefficient for 12 wells.

Figure 5.7 shows the true and predicted values of velocity Y versus time of well 6. The correlation coefficient of this well is 0.722, which is the best case among the 12 wells. From this figure, we can see that the predicted values have a similar trend as do the true values. The worst case, well 5 with correlation coefficient 0.364, is shown in Figure 5.8. The true and predicted values of the velocity Y of this well have no obvious correlation.

The average correlation coefficient between the prediction and the validation target of 12 wells is 0.5860 computed by using GAM. Compared to the 0.5857 obtained by the linear regression, we couldn't find an obvious improvement. Moreover, since the purpose of GAM is to maximize the quality of prediction of the dependent variable Y from various distributions, the correlation coefficients of all the wells are expected to be improved by using GAM instead of the linear model. However, we find there are four wells whose correlation coefficients decrease. The reason might be that only one distribution, Gaussian, was considered due to limited time. The smoother methods could also influence the results as well. We chose the spline smoother. One might try different smoothers in S-plus or various combinations of parametric linear functions and non-parametric smooth functions to get better results. Another insight might be provided by the scatter plots in Figure 5.9, which capture the information in the covariance matrix. Only the first attribute (impedance) displayed any significant correlation.

5.5 The Spline Method

5.5.1 Introduction of the Spline Method

Approximation of the target is computed using a function from a certain class. In the case of the spline method, the class of functions consists of the set of all linear combinations of centrally symmetric functions with centres at the training points. That is, $f(x)$ is chosen of the form,

$$f(x) = \sum_{i=1}^{n} w_i \varphi(|x - x_i|). \tag{5.13}$$

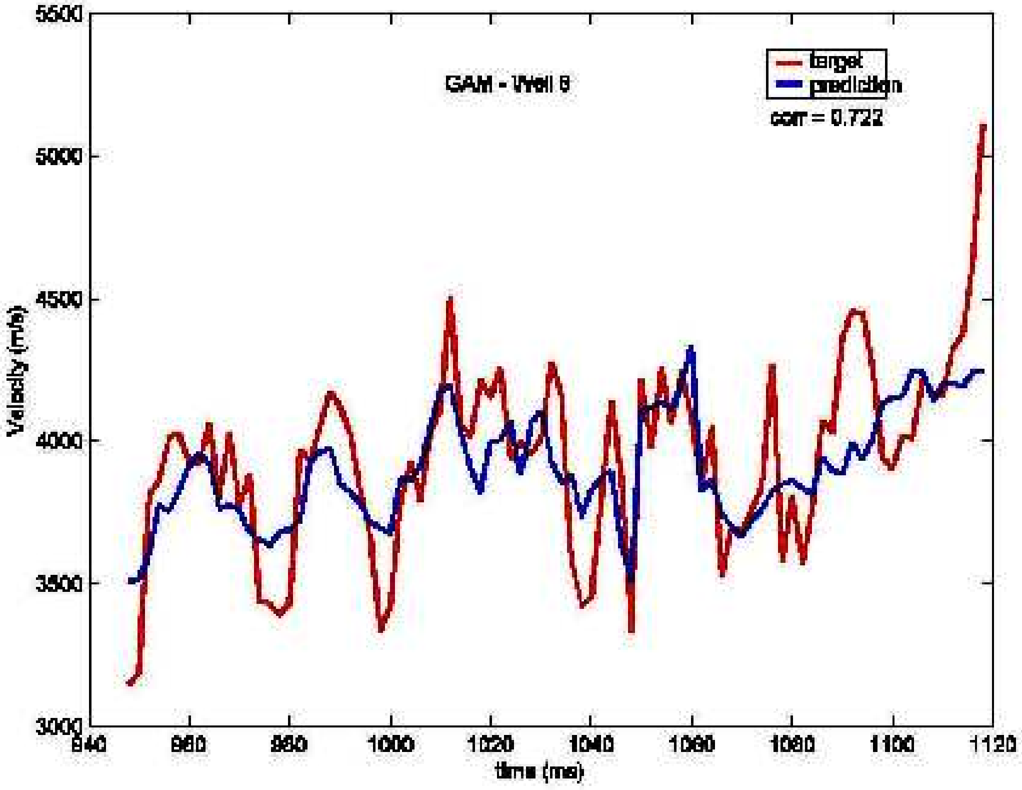


Figure 5.7: The true (red) and predicted (blue) values of velocity Y vs. time (Well 6 –the best case).

where $\varphi(z)$ is a one-variable function which can be experimentally chosen for optimized results, and x_i are the training points. The coefficients w_i are then selected to make the equality $f(x_i) = t_i$ hold exactly at the training points. This gives rise to the following set of n linear constraints on the coefficients w_i,

$$t_i = f(x_i) = \sum_{j=1}^{n} w_i \varphi(|x_i - x_j|), \qquad i = 1, 2, \ldots n. \tag{5.14}$$

This system of equations has a unique solution that can be easily found as long as the matrix,

$$M = \begin{bmatrix} \varphi(|x_1 - x_1|) & \varphi(|x_2 - x_1|) & \cdots & \varphi(|x_n - x_1|) \\ \varphi(|x_1 - x_2|) & \varphi(|x_2 - x_2|) & \cdots & \varphi(|x_n - x_2|) \\ \vdots & \vdots & \ddots & \vdots \\ \varphi(|x_1 - x_n|) & \varphi(|x_2 - x_n|) & \cdots & \varphi(|x_n - x_n|) \end{bmatrix}, \tag{5.15}$$

is invertible and well-conditioned. Solving the system for w_i gives us the predicting function $f(x)$ we are looking for.

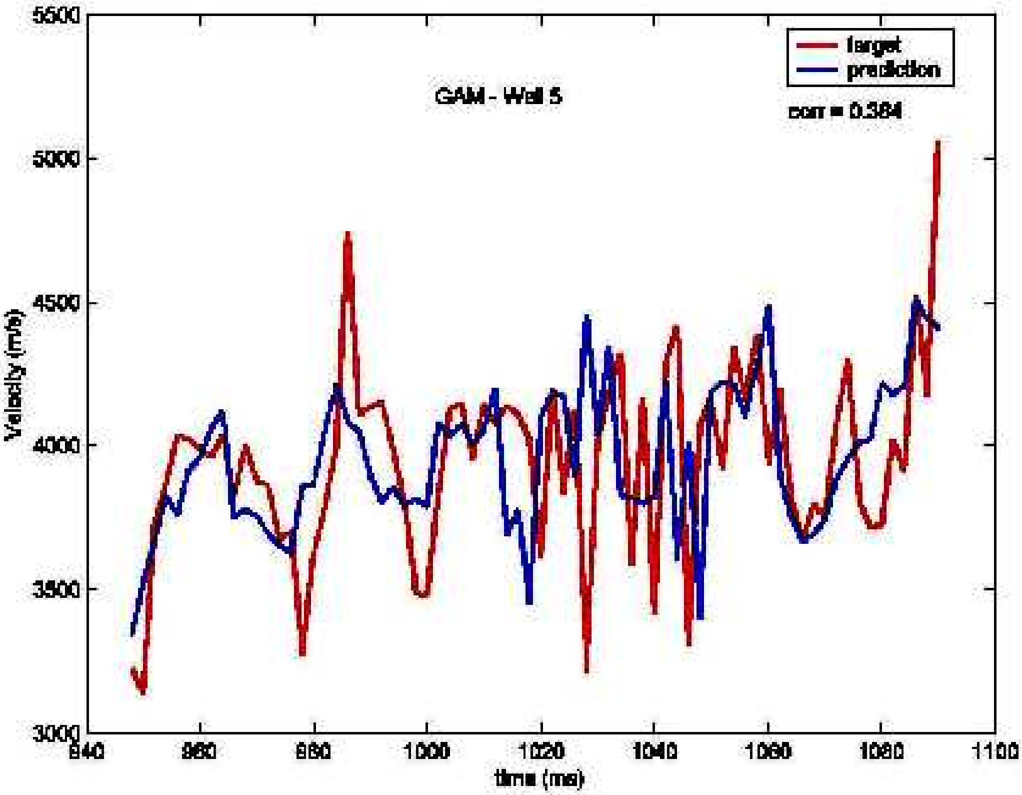


Figure 5.8: The true (red) and predicted (blue) values of velocity Y vs. time (Well 5–the worst case).

5.5.2 Improvement of the Spline Method

The spline method generally works better where there is no linear "drift" in the target function. We improve on the spline method by "taking out" the multi-linear regression component before applying the spline method. Given the input we first compute a multi-linear regression $L(x)$. We then use the spline method on the training set of pairs $(x_i, t_i - L(x_i))$ to obtain a prediction function $g(x)$ as described above. We then output the prediction function for the original input: $f(x) = g(x) + L(x)$. Note that $f(x)$ fits perfectly on the training set, $f(x_i) = t_i - L(x_i) + L(x_i) = t_i$. Figure 5.10 is the schematic diagram of this process described here.

5.5.3 Results and Analysis

We have experimented with functions $\varphi(z)$ of the form $\varphi(z) = z^\alpha + \beta$. The best estimates for the given sample are achieved with,

$$\varphi(x) = x^{0.15}. \tag{5.16}$$

The average correlation coefficient between the prediction and the validation target for 12 wells is 0.631. The average error in the l_2 norm is 0.76σ and the average error in the l_1 norm is 0.57σ, where σ is the sample standard deviation of true target values. Sample performance of the method is illustrated on Figure 5.11 for well 6 (best case) and Figure 5.12 for well 9 (worst case).

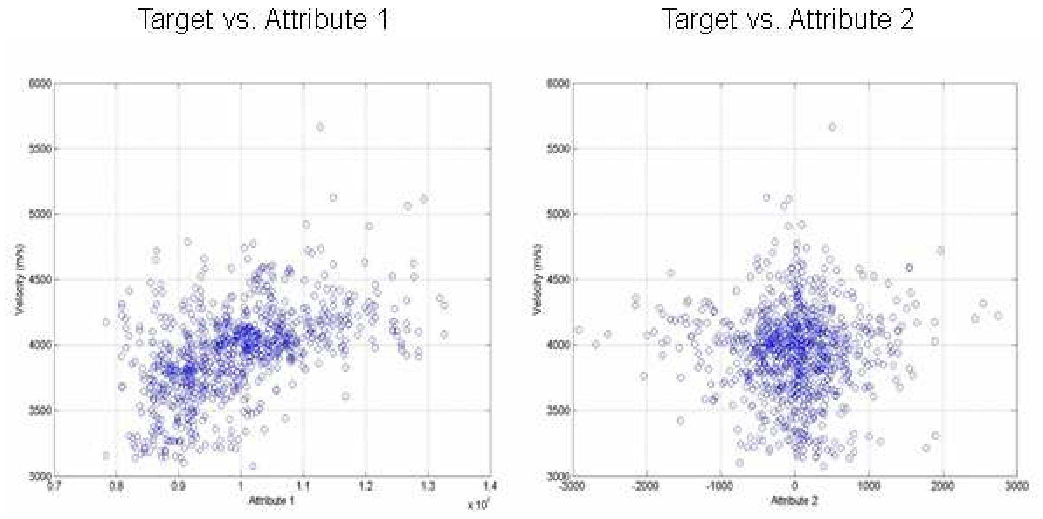


Figure 5.9: Scatter plots of the target versus the attributes. Attribute 1 is the impedance, and it is the only attribute that displays a significant correlation.

The correlations for well 6 and well 9 are 0.773 and 0.489, respectively. Visually, one can see that the prediction quality for well 6 is much better than that for well 9. The main place where the method fails is in predicting spikes. In particular, its performance is worse on well 9 because the target data for this well contains several sharp spikes which the method fails to predict.

5.6 Discussion

All of the methods have limited resolution. They cannot capture the rapid variations occurring on the high resolution scale of the well logging data. The supplemental information contained in the attributes derives from the (relatively) low resolution seismic survey data, thus the statistical attempt to combine the two sets of information into a single predictor will ultimately have a resolution limit as well. Figure 5.13 provides a typical illustration of this limitation.

Throughout the calculations, the number of attributes taken varied from 4 to 7, with the first attribute (impedance) common to all cases. There was relatively little sensitivity to the number of attributes, however, since only the first attribute exhibited any significant correlation (see Figure 5.9), this result is not surprising.

An alternative approach is to consider local estimation; that is to estimate properties in one layer based solely on the deterministic information from the log wells in that layer. This idea is illustrated in Figure 5.14. This is particularly appropriate for localized parameter estimation. Figure 5.15 illustrates a spline fit in the chosen depth plane. Figure 5.16 illustrates the predicted values in the chosen well based on the local spline estimation compared with the true values. This fit appears to be consistent with the best of the earlier spline fit results.

In conclusion, the spline method seemed to offer the best results. Given the limited duration of the IPSW, however, this is a tentative conclusion. There are other statistical approaches beside the GAM,

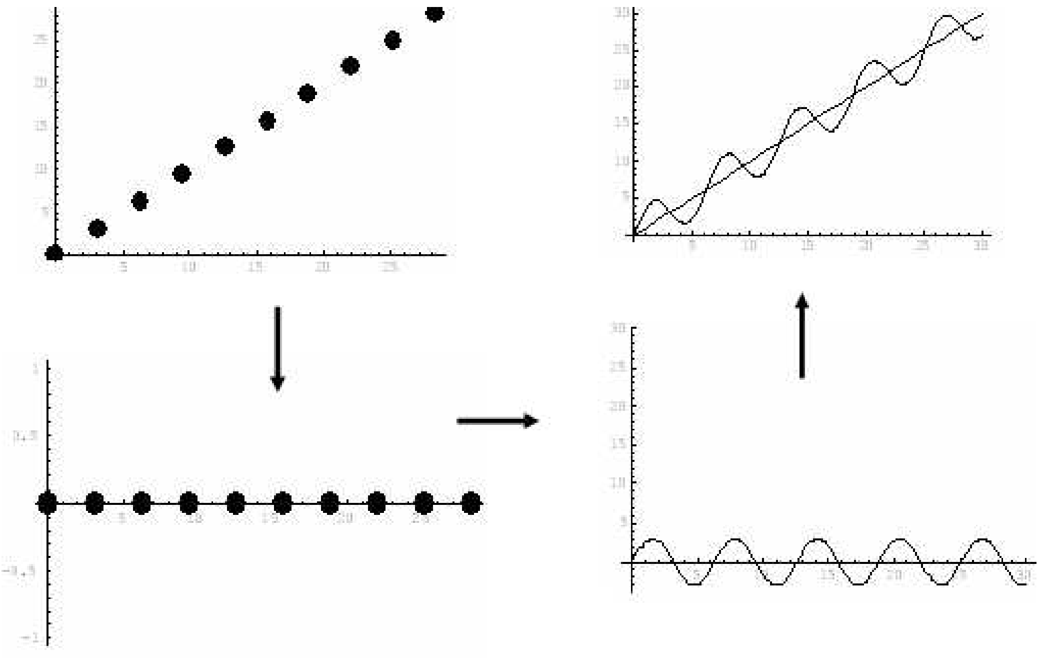


Figure 5.10: Improving the Spline Method.

and they should probably be considered. There is also the question of to what degree these examples were dominated by the first attribute.

5.7 Acknowledgements

The authors would like to acknowledge the Pacific Institute for the Mathematical Sciences (PIMS) for their sponsorship of the IPSW. The University of Calgary is also acknowledged for their generous hospitality in hosting the IPSW. Finally, specific thanks to Elena Braverman and Marian Miles for all of their hard work.

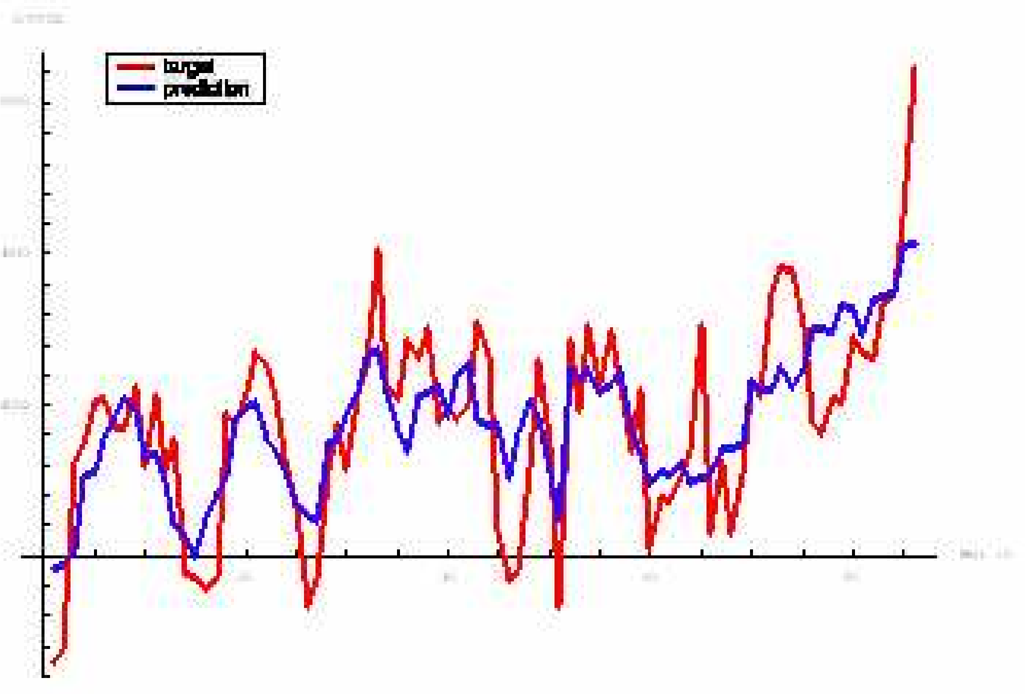


Figure 5.11: The true (red) and predicted (blue) values of velocity Y vs. time (Well 6–the best case).

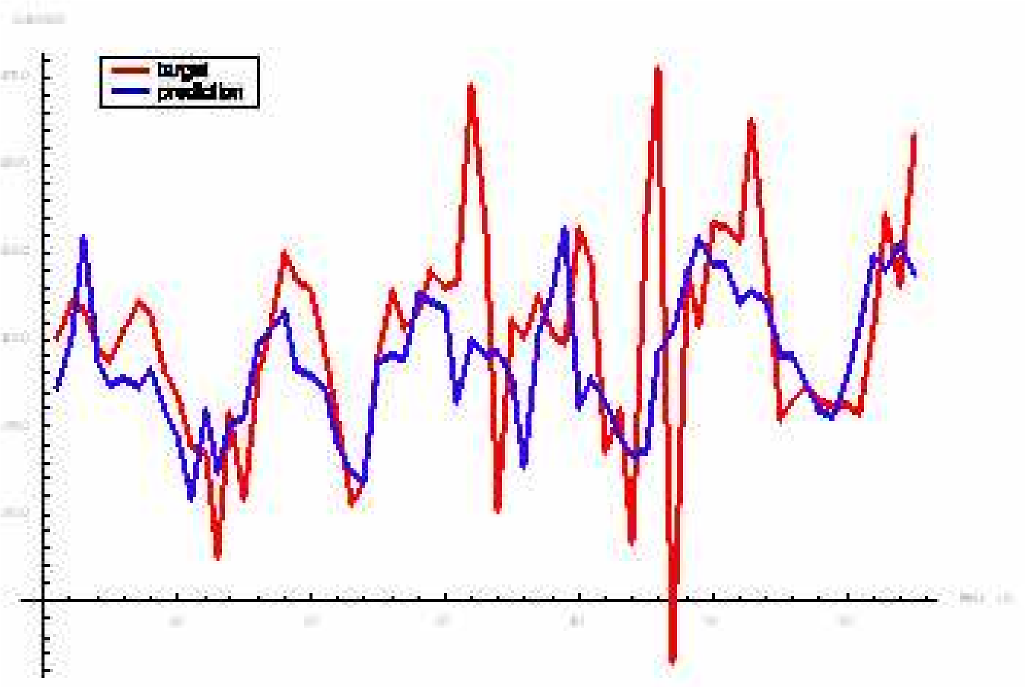


Figure 5.12: The true (red) and predicted (blue) values of velocity Y vs. time (Well 9–the worst case).

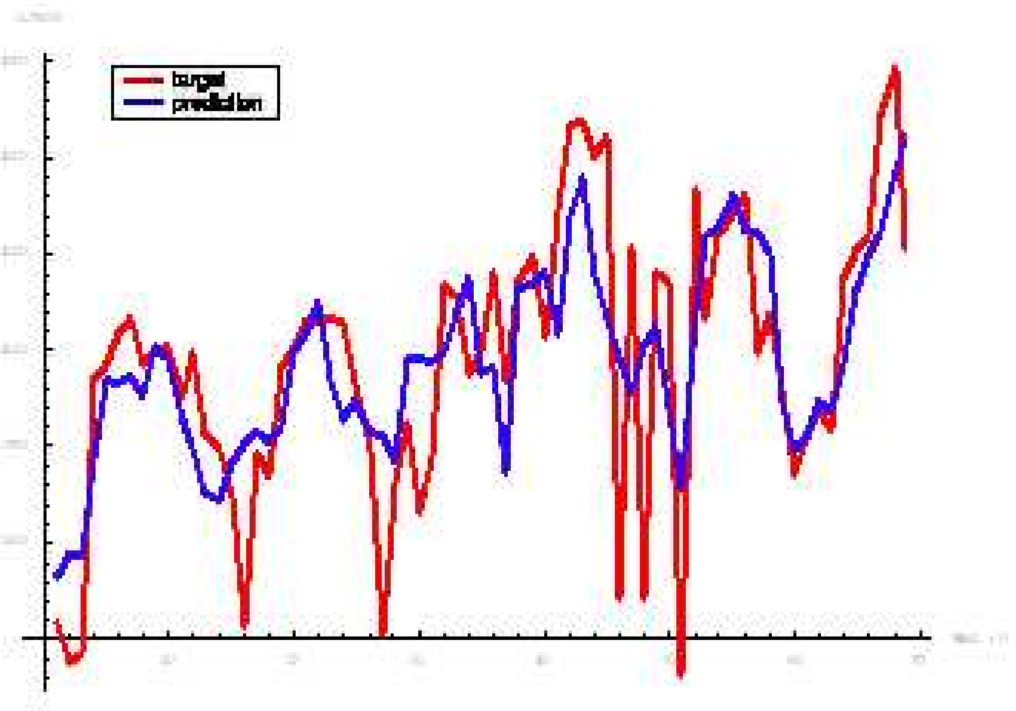


Figure 5.13: The true (red) and predicted (blue) values of the velocity Y vs. time for a typical example in the study. Notice that the rapidly changing features are very hard to predict correctly.

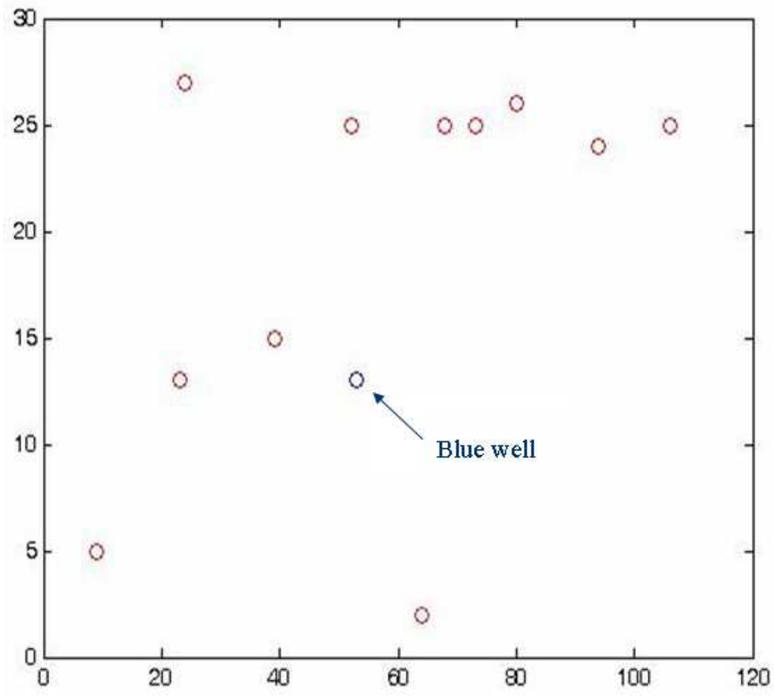


Figure 5.14: The location of the 12 wells in a particular depth plane, with the blue well signifying the well to be predicted.

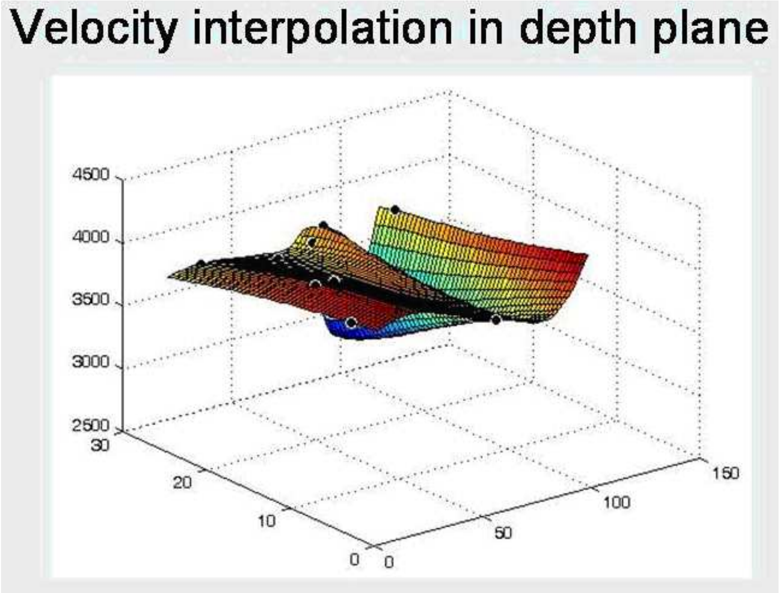
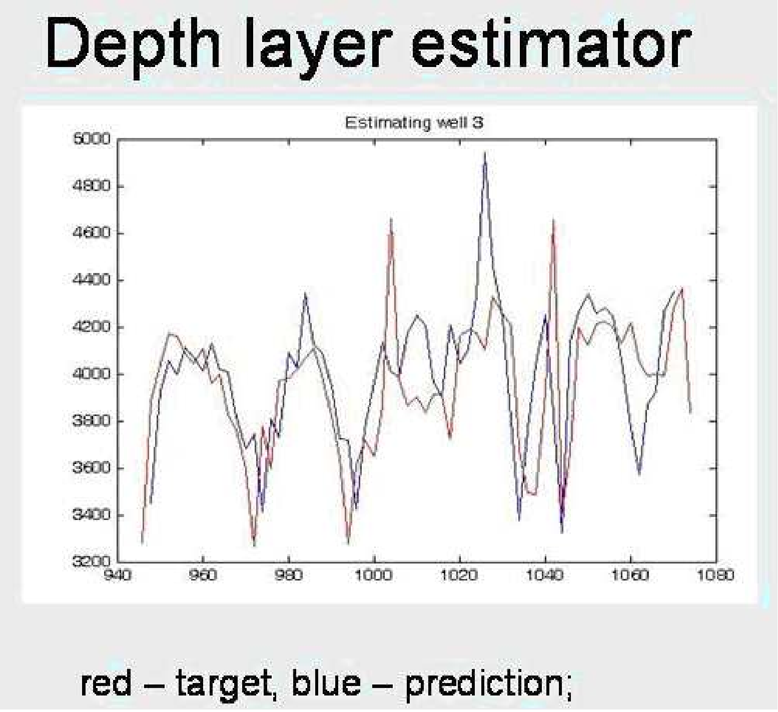


Figure 5.15: A spline fit in a given depth plane.



Depth layer estimator

red – target, blue – prediction;

Figure 5.16: The true (red) and the predicted (blue) values for the velocity Y vs. time for the local depth spline estimator.

Bibliography

[1] Bishop, C. M. (1995). Neural Networks for Patten Recognition. Oxford University Press.

[2] Hastie, T. & Tibshirani, R. (1990). Generalized additive models. Chapman & Hall: New York.

[3] Johnson, R. A. & Wichern, D. W. (1998). Applied Multivariate Statistical Analysis. Prentice Hall.

[4] Masters, T. (1995). Advanced Algorithms for Neural Networks. John Wiley & Sons, Inc.

[5] Russell, B. (2005). The Application of Multivariate Statistics and Neural Networks to the Prediction of Reservoir Parameters from Seismic Attributes: Ph.D. dissertation, University of Calgary.

[6] Russell, B. (2005). Seismic Prediction of Reservoir Parameters (An IPSW Problem).

Appendix A

List of Participants

Mahmud Akelbek	University of Regina	akelbek@math.uregina.ca
Amir Amiraslani	University of Western Ontario	aamiras2@uwo.ca
Ojenie Artoun	Concordia University	eugenie_a72@yahoo.com
Brad Bondy	Genus Capital Management	bondy@genuscap.com
Len Bos	University of Calgary	lpbos@math.ucalgary.ca
Elena Braverman	University of Calgary	maelena@math.ucalgary.ca
Mark Braverman	University of Toronto	mbraverm@cs.toronto.edu
Gemai Chen	University of Calgary	gchen@math.ucalgary.ca
Robin Clysdale	University of Calgary	raclysda@ucalgary.ca
Diana David-Rus	Rutgers University	ddavid@rci.rutgers.edu
Matthew Emmett	University of Calgary	memmett@math.ucalgary.ca
Lou Fishman	MDF International, University of Calgary	lou@math.ucalgary.ca
Sandra Fital	University of Regina	fital20semath@uregina.ca
John Gonzalez	Northeastern University	gonzalez.jo@neu.edu
Kent Griffin	Washington State University	kgriffin@math.wsu.edu
Donald Henderson	University of Calgary	dmhender@ucalgary.ca
Chad Hogan	University of Calgary	cmhogan@ucalgary.ca
Thomas Holloway	University of Alberta	tholloway@math.ualberta.ca
Hui Huang	University of British Columbia	hhzhiyan@math.ubc.ca
Mark Hughes	University of Calgary	mchughes@ucalgary.ca
Parisa Jamali	University of Western Ontario	parissaarg2001@yahoo.com
Gerald K. Cole	Biomechanigg Research Inc.	cole@kin.ucalgary.ca
Michael Lamoureux	University of Calgary	mikel@math.ucalgary.ca
Pierre Lemire	Calgary Scientific Inc.	pierre.lemire@telus.net
Greg Lewis	University of Ontario	Greg.Lewis@uoit.ca
Hua Li	University of Calgary	lih@math.ucalgary.ca
Jisun Lim	University of Colorado	jisun@colorado.edu
Enkeleida Lushi	Simon Fraser University	elushi@sfu.ca
Yongwang Ma	University of Calgary	yongma@ucalgary.ca
Donald Mackenzie	University of Calgary	
Maryam Mizani	University of Victoria	mmizani@ece.uvic.ca

Mahyar Mohajer	University of Calgary	mmohajer@ucalgary.ca
Bertrand Monthubert	Institut de Mathematiques de Toulouse	bertrand@monthubert.net
Comron Nouri	University of Western Ontario	mnouri@uwo.ca
James Odegaard	University of Western Ontario	jodegaa@uwo.ca
Gergeli Orosi	University of Calgary	gorosi@math.ucalgary.ca
Xiao Ping Liu	University of Regina	liuxp@math.uregina.ca
Rob Pinnegar	Calgary Scientific Inc.	
Hatesh Radia	University of Massachusetts at Lowell	hmathre@yahoo.com
Jihong Ren	University of British Columbia	jihong@co.ubc.ca
Nargol Rezvani	University of Western Ontario	nrezvani@uwo.ca
Brian Russell	Hampson Russell Software	brian_russell@veritasdgc.com
C. Sean Bohun	Pennsylvania State University	csb15@psu.edu
Peter Smith	Memorial University of Newfoundland	smitty@physics.mun.ca
Qiao Sun	University of Calgary	qsun@enme.ucalgary.ca
Dallas Thomas	University of Lethbridge	dallas.thomas@uleth.ca
Naveen Vaidya	York University	nvaidya@mathstat.yorku.ca
Pengpeng Wang	Simon Fraser University	pwangf@cs.sfu.ca
Tony Ware	University of Calgary	aware@ucalgary.ca
Rex Westbrook	University of Calgary	westbroo@ucalgary.ca
Sarah Williams	University of California, Davis	sawilliams@math.ucdavis.edu
Liang Xu	University of Washington	lxu@math.washington.edu
Oulu Xu	York University	ona@mathstat.yorku.ca
Yaling Yin	University of Saskatchewan	yin@snoopy.usask.ca
Zhidong Zhang	University of Saskatchewan	zhang@math.usask.ca

PIMS Contact Information

email: pims@pims.math.ca

http://www.pims.math.ca

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