Eighteenth Mathematical and Statistical Modeling Workshop for Graduate Students

16 – 24 July 2012
North Carolina State University
Raleigh, NC, USA

Organizers:
Pierre Gremaud, Ilse C.F. Ipsen, Ralph C. Smith
Department of Mathematics
North Carolina State University

This report contains the proceedings of the *Industrial Mathematical and Statistical Modeling Workshop for Graduate Students*, held in the Department of Mathematics at North Carolina State University (NCSU) in Raleigh, North Carolina, 16 – 24 July 2012.

This was the eighteenth such workshop at NCSU. It brought together 35 graduate students from Mathematics and Statistics Departments at 34 different universities.

The goal of the IMSM workshop is to expose mathematics and statistics students from around the country to: real-world problems from industry and government laboratories; interdisciplinary research involving mathematical, statistical and modeling components; as well as experience in a team approach to problem solving.

On the morning of the first day, industrial and government scientists presented six research problems. Each presenter, together with a specially selected faculty mentor, then guided teams of 4 – 6 students and helped them to discover a solution. In contrast to neat, well-posed academic exercises that are typically found in coursework or textbooks, the workshop problems are challenging real world problems that require the varied expertise and fresh insights of the group for their formulation, solution and interpretation. Each

group spent the first eight days of the workshop investigating their project and reported their findings in half-hour public seminars on the final day of the workshop.

The IMSM workshops have been highly successful for the students as well as the presenters and faculty mentors. Often projects lead to new research results and publications. The projects can also serve as a catalyst for future collaborations between project presenter and faculty mentor. More information can be found at

http://www.samsi.info/workshop/2012-industrial-mathstat-modeling-workshop-graduate-students-july-16-24-2012

Sponsors

Statistical and Applied Mathematical Sciences Institute (SAMSI)
Center for Research in Scientific Computation (CRSC)
Department of Mathematics, North Carolina State University

Presenters

Agustin Calatroni, Jeremy Wildfire, and Herman Mitchell, Rho Inc.
Matthew Farthing, US Army Corps of Engineers
Haris Halilovic, Lord Corporation
Jordan Massad, Sandia National Laboratories
John Peach, MIT Lincoln Laboratory
Mark Wolff, SAS

Faculty mentors

Andreas Aristotelous, SAMSI and Duke
H.T. Banks, NCSU
Cammey Cole Manning, Meredith College
Shuhua Hu, NCSU
Lea Jenkins, Clemson University
Chia Ying Lee, SAMSI and UNC
Emily Lei Kang, University of Cincinnati
Ralph Smith, NCSU, Mathematics

Projects

Uncertainty-enabled design of an active MEMS valve for a high-pressure micro gas analyzer
Problem presenter: Jordan Massad
Faculty mentor: Ralph Smith
Students: Bruce Bugbee, Brianna Cash, Jun Liu, Helen Parks, Wei Qi, Deling Wei, Xi Zhang

Robot vehicles solving mazes
Problem presenter: John Peach
Faculty mentors: Andreas Aristotelous, Cammey-Cole Manning
Students: Abirbhab Bandyopadhyay, Rana Haber, Sunil Thulaseidan, Jia Zhao

Saltwater intrusion and freshwater supply in coastal aquifers
Problem Presenter: Matthew Farthing
Faculty mentor: Lea Jenkins
Students: Quanling Deng, Will Henry, Mona Karimi, Sudip Shrestha, Mikhail Smilovic, Jillian Trask, Liang Wu

Inspection of composite assemblies using a non-destructive approach
Problem Presenter: Haris Halilovic
Faculty mentors: H. Thomas Banks, Shuhua Hu
Students: Fernando Camacho, Marylesa Howard, Zachary Kenz, Ruiqiang Song, Wei Yijun, Xiaoqing Zhu

Establishing an automated measurement of allergic reactivity
Problem Presenters: Agustin Calatroni, Herman Mitchell, Jeremy Wildfire
Faculty mentor: Emily Lei Kang
Students: Gyuhyeong Goh, Nicole Mendoza, Oluseyi Odubote, Yao Yu

Signal detection for drug safety
Problem Presenter: Mark Wolff
Faculty mentor: Chia Ying Lee
Students: Viviana Garcia, Peng He, Xuefei Jia, Andrea Knezevic, Ying Lu, Mahmoud Shehadeh, Ranye Sun

Student Participants

Abirbhab Bandyopadhyay	Indian Statistical Institute
Bruce Bugbee	Colorado State University
Fernando Camacho	University of Kentucky
Brianna Cash	University of Maryland
Quanling Deng	University of Wyoming
Viviana Garcia	Harvard University
Gyuhyeong Goh	University of Connecticut
Rana Haber	Florida Institute of Technology
Peng He	University of Cincinnati
Will Henry	San Diego State University
Marylesa Howard	University of Montana
Xuefei Jia	Georgetown University
Mona Karimi	Texas A&M University
Zackary Kenz	North Carolina State University
Andrea Knezevic	Georgia Tech
Jun Liu	Southern Illinois University
Ying Lu	University of Minnesota-Twin Cities
Nicole Mendoza	UC Santa Cruz
Oluseyi Odubote	Georgia Southern University
Helen Parks	UC San Diego
Wei Qi	UC Berkeley
Mahmoud Shehadeh	Northern Illinois University
Sudip Shrestha	University of Georgia
Mikhail Smilovic	McGill University
Ruiqiang Song	Michigan Technological University
Ranye Sun	Texas A&M University
Sunil Thulasidasan	University of Washington-Seattle
Jillian Trask	University of Tennessee, Knoxville
Deling Wei	Pennsylvania State University
Liang Wu	University of Notre Dame
Wei Yijun	University of Michigan
Yao Yu	University of Rochester Medical Center
Xi Zhang	Utah State University
Jia Zhao	University of South Carolina
Xiaqing Zhu	Michigan State University

PROBLEM 1: UNCERTAINTY ENABLED DESIGN OF AN ACTIVE MEMS VALVE FOR A HIGH-PRESSURE MICRO-GAS-ANALYZER

Bruce Bugbee[1], Brianna Cash[2], Jun Liu[3], Helen Parks[4], Wei Qi[5], Deling Wei[6], Xi Zhang[7]

Faculty Mentor: Prof. Ralph Smith, North Carolina State University

Problem Presenter:
Dr. Jordan Massad
Sandia National Laboratories

Abstract

The ability to rapidly and accurately analyze an air sample to detect trace amounts of a toxin in ambient air is a critical need for portable gas analyzers. The combination of gas chromatography and microsystems, a Micro-Gas-Analyzer (MGA), offers a viable integrated technological solution. MGAs can increase sample separation capacity and speed, maintain high sensitivity, provide rapid response time, and allow for miniaturized systems that require no consumables and low power consumption. Sandia is developing an active micro-electromechanical-systems (MEMS) flow control valve that further enables an MGA to achieve low leak rates and high flow rates, to sustain record-breaking high operating pressures, and to require low control voltages. The active valves operate under a nonlinear electrostatic force that causes them to close when the given a control voltage against a lower pressure, hold closed against high pressures needed for rapid sample loading, and re-open for gas chromatograph loading. The material properties and structural design of the valve ultimately determine how it closes and opens. The valve is designed to operate optimally under prescribed conditions. However, due to practical limitations in batch fabrication, there is variation and uncertainty in critical design parameters for any given valve. Hence, a realizeable valve may not perform well or at all, thus disabling the MGA. Determination of a valve design whose operation and performance is less sensitive to variation and uncertainty is sought. This project aims to understand how the combination of a control voltage and key design parameters determine the valve dynamics, flow rate, leak rate and other measures of valve performance. In our work, we develop a single degree of freedom model of the valve boss dynamics that successfully captures the most significant behavior of the valve boss and outputs an impact velocity of the boss upon closing. Transient data from laser Doppler velocimetry on fabricated valves was provided by Sandia National Laboratories for model calibration and validation. We study steady states of our dynamic model and their stability to gain insight. We quantify flow rate and leak rate of the valve using mathematical models and optimize these rates with respect to key design parameters. We also construct a Bayesian model for quantifying uncertainty within the dynamic model and the design parameters. We are able to quantify the variation in the impact velocity and the flow rate with respect to design parameters, and we provide a rough estimate of uncertainty from our statistical study.

1 Introduction and Background

1.1 Motivation

The ability to rapidly and accurately analyze an air sample to detect trace amounts of a toxin in ambient air is a critical. A Micro-Gas-Analyzer (MGA) is a portable, low-power device for rapidly detecting chemical and biological agents in the surrounding air. Ideally, the MGA would operate with high sensitivity (ppm to ppb) and high selectivity (<0.01% false positive) with an analysis time around 4s [1]. Sandia National Laboratories

[1]Colorado Sate University
[2]University of Maryland
[3]Southern Illinois University
[4]University of California, San Diego
[5]University of California, Berkeley
[6]Pennsylvania Sate University
[7]Utah State University

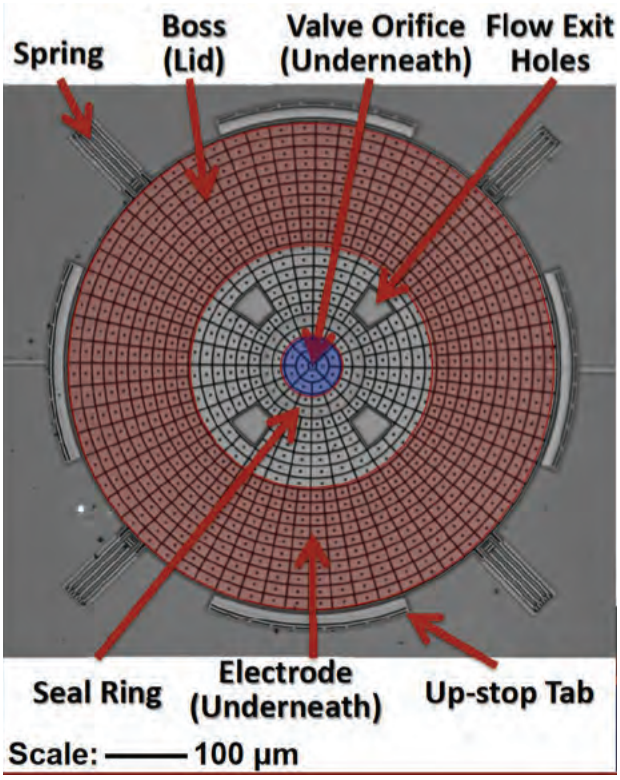


Figure 1: Top view of the valve design.

is developing an MGA [1] in which sample air is passed over a pre-concentrator (PC), which gathers species of possible interest, then directed through a gas chromatograph (GC) column, where species are separated, and finally sent to a detector for species identification. This process is modulated by a series of active and passive valves along the sample flow path. The passive valves are actuated by pressure impulses. The active valves are actuated electrostatically, allowing path selection within the MGA and providing pressure impulses to activate the passive valves [1].

Short analysis times require swift gas chromatographic separation which necessitates rapid sample injection. Decreasing the volume of the sample injector is one way to decrease sample injection time, and minimization of injector volume motivates the development of microfabricated valves [1]. In this project, we focus on the active valves used in sample injection.

1.2 Valve Design and Operation

Figures (1) and (2) show two different views of the active valve design. The valve consists of a boss (lid) suspended by four bifold springs above a base with an annular electrode. The valve base has a center hole, called the Bosch hole, and a seal ring surrounding the Bosch hole. The boss has four flow exit holes to facilitate air flow when open, a tooth ring that fits into the seal ring on the base when closed, and an outer stop ring which prevents the boss from touching the electrode when the valve is closed. Four up-stop tabs around the boss prevent the boss from opening too far upward.

In the absence of pressure and voltage, the springs suspend the valve at an equilibrium point between the valve base and the unstop tabs. We quantify this position as g_u below the up-stop tabs with g_{es} between the boss and the electrode. When a voltage is applied, an electrostatic force acts on the boss in the area above the electrode, bringing the boss down to close. While the valve is closed, sample air is trapped by the seal ring and pressure builds. This air exerts a pressure force on the boss, and the holding pressure of the valve is the maximum value of this pressure under which the valve remains closed. After sample loading, the voltage

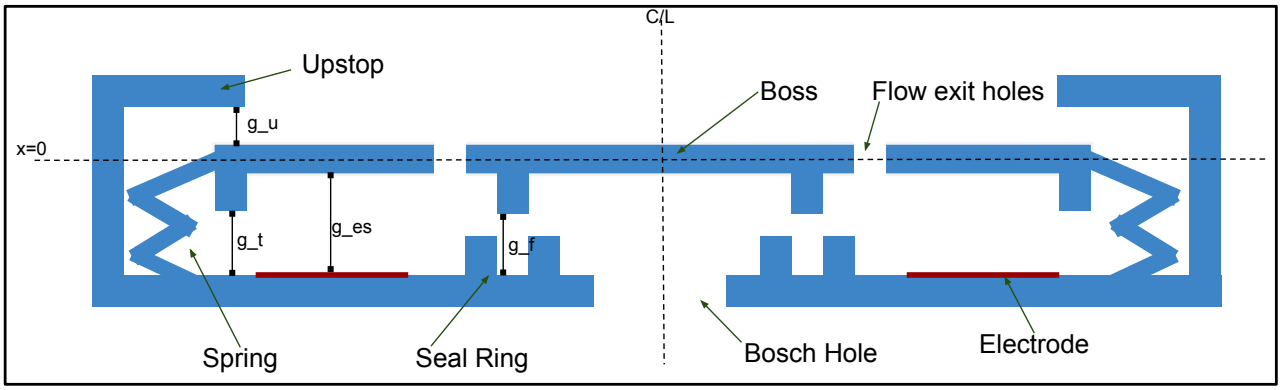


Figure 2: Cross-sectional view of the valve design with design parameters labeled. The Bosch Hole here is the same as the Valve Orifice in Figure (1).

Specifications:	**Values:**	**Unit:**
Min flow rate	1.00e-07	m^3/s
Min hold off pressure	690	kPa
Max leak rate	1.00e-09	m^3/s

Table 1: Design Specifications

is turned off, and the valve opens. In the open position, air flows in through the Bosch hole, through the gap g_f in the open seal ring, and out through the flow exit holes.

1.3 Valve Performance

For optimal MGA performance, we desire active valves with a high flow rate when open, a low leak rate when closed, high holding pressure, and low operating voltage. Table 1 lists the valve specification required by the system [1]. We also desire a low impact velocity upon valve closure to minimize bouncing, thereby minimizing wear on the valve and increasing operational lifespan.

In this project, we construct a mathematical model of the valve's opening and closing dynamics as well as steady state models to calculate flow rate, leak rate, and holding pressure. We validate our models against data provided by Sandia National Laboratories, then use them to examine the sensitivity of valve performance to a range of design parameters.

2 Model Development

2.1 Dynamic Model Equation

The MEMS active valve can be modeled as a forced mass-spring-damper system in Figure 3, where the boss is the mass with the four springs suspending it creating an effective spring force. The system is forced by the introduction of a electrode which creates a electrostatic force (F_{es}) which is used to close the valve as well as a force due to the pressure in the system (F_p) which when the electrostatic force is absent opens the valve. The system can represented by a second-order differential equation,

$$\begin{cases} x(0) = x_0,\ \dot{x}(0) = v_0 \\ m\ddot{x} + c\dot{x} + k_{eff}x = -F_{es} + F_p \end{cases} \tag{2.1}$$

Parameters:	Nominal Values:	Unit:
g_{es}	1.82	μm
g_u	2	μm
g_t	0.5	μm
m_{boss}	4.3e-9	kg
ϵ	8.84e-12	F/m
A_{es}	9.35e-7	m^2
R_{Bosch}	30e-6	m
k_{eff}	112.4	N/m

Table 2: Design parameters.

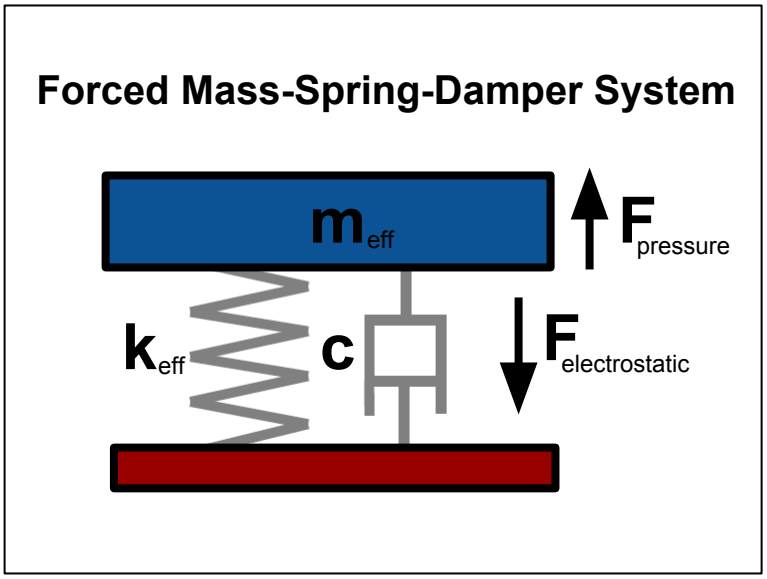


Figure 3: Rigid body diagram of valve boss with effective mass

where m is the effective mass of the boss, c is a viscous damping coefficient, and k_{eff} is the effective spring constant for the four springs.

If we consider 2.1 as a first-order system in displacement x and velocity v we can write

$$\begin{cases} \dot{x} = v; \\ \dot{v} = \frac{1}{m}\left(-cv - k_{eff}x - F_{es} + F_p\right) \end{cases} \tag{2.2}$$

Negative velocity denotes motion towards the substrate. The electrostatic force is determined by treating the boss electrode system as a capacitor which is found to be

$$F_{es} = \frac{\epsilon A_{es} V(t)^2}{2(g_{es} + x)^2} \tag{2.3}$$

where A_{es} is the area of the electrode, V is the applied voltage, g_{es} is the distance between the equilibrium of the unforced system and the electrode, and ϵ is the permitivity of air. The force due to pressure is given by

$$F_p = PA_b \tag{2.4}$$

where A_b is the area of the Bosch opening. The pressure is assumed to be stepwise constant function where $P = P_{close}$ (applied pressure when boss is closed) when opening and $P = P_{open}$ when closing (applied pressure when open). The voltage is assumed to be a step function with a 0.5 μs rise time.

Starting from the open position when a sufficiently large voltage is applied, the boss will drop down to hit seal seat at some time t_1 with velocity v_{impact}. To simulate the loss of kinetic energy and bouncing due to the collision we reduce and reverse the v_{impact} by a coefficient of restitution (COF) α, which represents the ratio of the relative speed after collision to the relative speed before collision [3]. For all simulations we assume that $\alpha = .5$. Bouncing will continue until it displaces less than a tolerance of $1e^{-8}m$. When in the closed position, if the voltage is turned off and P_{close} is applied we simulate the boss hitting the upstop using the same COR. Figure 4 is an example of the dynamics of the boss.

2.2 Equilibrium solutions and their stability

In order to optimize those critical design parameters (such as boss-to-seat gap, boss-to-electrode gap, etc.), we need to understand how the boss behaves as time limits to infinity. In other words, we seek the equilibrium solutions of our model equation. More importantly, one must analyze their stability properties to characterize those that can be realized in practical physical circumstances. An equilibrium is called *asymptotically stable* provided every solution that starts out near the equilibrium converges to it. An equilibrium is called *stable* provided every solution that starts out nearby stays nearby. For zero voltage, the stationary equation of (2.2) is given by $v = 0$, where

$$x = PA_b/k_{eff} \tag{2.5}$$

For nonzero voltage,

$$\begin{cases} 0 &= v; \\ 0 &= \frac{1}{m}\left(-\frac{\epsilon A_{es} V^2}{2}\frac{1}{(g_{es}+x)^2} - k_{sp}x + PA_b - cv\right). \end{cases} \tag{2.6}$$

2.3 Stability criterion

To solve the nonlinear equation (2.6), we let $z = g_{es} + x$ and plug it back into (2.6) to get

$$\frac{\epsilon A_{es} V^2}{2}\frac{1}{z^2} + k_{eff}z - k_{eff}g_{es} - PA_b = 0, \tag{2.7}$$

which can be rewritten as a cubic equation (neglect zero solution)

$$f(z) \equiv z^3 + \left(\frac{-PA_b}{k_{eff}} - g_{es}\right)z^2 + \frac{\epsilon A_{es} V^2}{2k_{eff}} = 0. \tag{2.8}$$

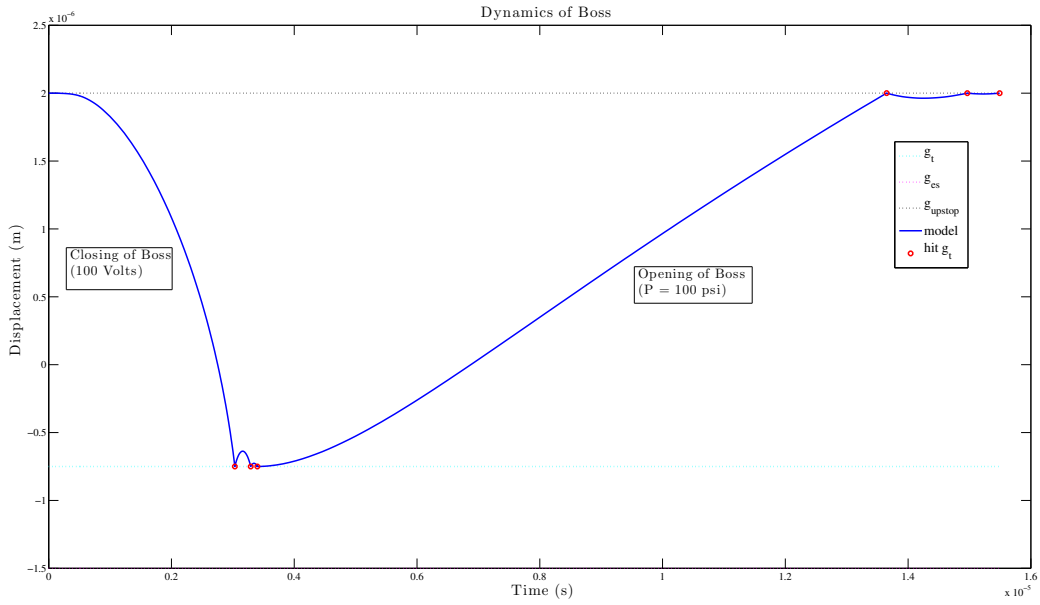


Figure 4: Example of the dynamics of the boss for parameters that results in bouncing both during closing and during opening.

For such a cubic polynomial $f(z)$, we have the analytic formulas for all three cubic roots. In particular, its discriminant, which is given by

$$\Delta_1 = \left[4\left(\frac{PA_b}{k_{eff}} + g_{es}\right)^3 - 27\left(\frac{\epsilon A_{es} V^2}{2k_{eff}}\right)\right]\frac{\epsilon A_{es} V^2}{2k_{eff}}, \tag{2.9}$$

determines the following three possible cases:

- $\Delta_1 > 0$: $f(z) = 0$ has three distinct real roots.
- $\Delta_1 = 0$: $f(z) = 0$ has a multiple root and all its roots are real.
- $\Delta_1 < 0$: $f(z) = 0$ has one real root and two non-real complex conjugate roots.

However, only real roots $z_r = g_{es} + x_r$ are physically meaningful. To study the stability of these equilibrium solutions $x_r = z_r - g_{es}$, we need to check the eigenvalues of the corresponding Jacobian matrix

$$J(x_r) = \frac{1}{m}\begin{bmatrix} 0 & m \\ \frac{\epsilon A_{es} V^2}{(g_{es}+x_r)^3} - k_{eff} & -c \end{bmatrix}. \tag{2.10}$$

Let

$$\rho = 4m\left(\frac{A_{es}\, V^2\, \epsilon}{(g_{es} + x_r)^3} - k_{eff}\right), \quad \Delta_2 = c^2 + \rho. \tag{2.11}$$

Then the explicit formulas for the two eigenvalues of $J(x_r)$ are

$$\lambda_1 = -\frac{c + \sqrt{c^2 + \rho}}{2m}, \quad \lambda_2 = -\frac{c - \sqrt{c^2 + \rho}}{2m}.$$

Based on this, we have the following possible stability conclusion (always assume $c > 0$):

- If $\rho > 0$, then $\lambda_1 < 0$ and $\lambda_2 > 0$, the equilibrium is *unstable*.
- If $\rho < 0$, then the equilibrium is *asymptotically stable* since
  - If $\Delta_2 = 0$, then $\lambda_1 = \lambda_2 = -c/(2m) < 0$;
  - If $\Delta_2 < 0$, then $Re(\lambda_1) = Re(\lambda_2) = -c/(2m) < 0$;
  - If $\Delta_2 > 0$, then $\lambda_1 < 0$ and $\lambda_2 < 0$.

Given a group of given design parameters, constant voltage V and constant pressure P, verifying the stability of each equilibrium x_r requires solving the roots of $f(z)$ and checking the sign of stability indicator ρ.

Applying above conclusion, we could correctly figure out all the equilibria and their stability properties for each of the solutions. However, only stable equilibria are the only steady state solutions that one directly obverses in physical system. Thus, we will only demonstrate those stable equilibria. Table 2 gives the current design parameters for our simulations.

In Figure 5 we plot the locations of stable equilibrium as a function of voltage V (with varying pressures), which shows the pull-in effect, i.e., the voltage when the boss will eventually snap down to the seal ring seat. We call this voltage the *pull-in voltage*. In Figure 6 we plot the locations of stable equilibrium as a function of

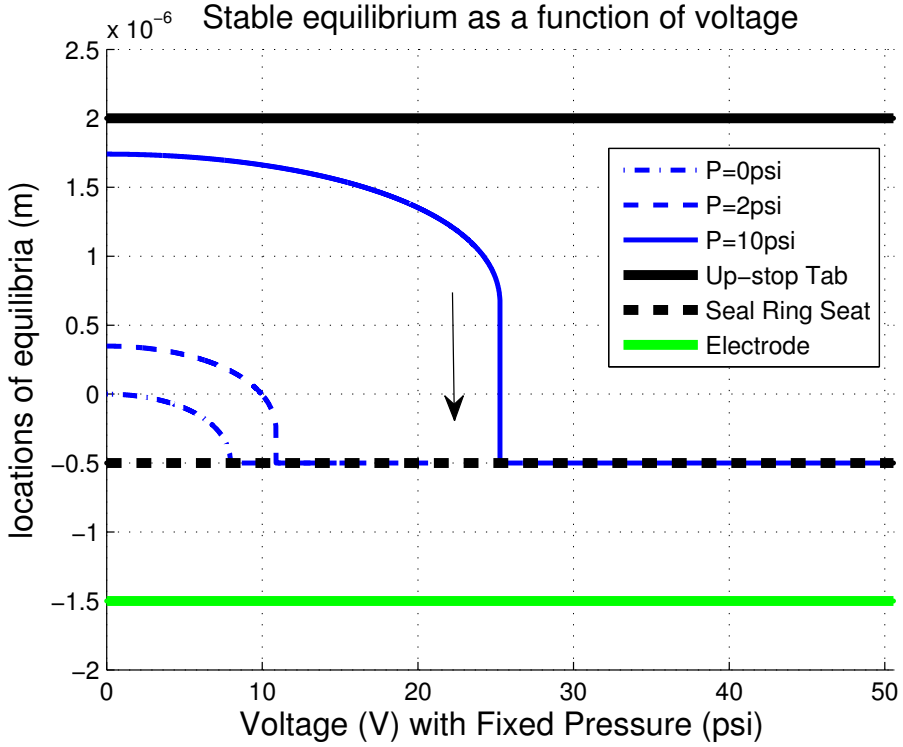


Figure 5: Equilibrium positions of boss as a function of voltage V.

pressure P (with varying voltages), which shows the pop-up effect, i.e., pressure when the boss will eventually open up. We call this pressure the *opening pressure*. Next, we proceed to determine both *pull-in voltage* V_{pi} and *opening pressure* P_{op} by manipulating the steady state equation. Both of them are crucial to the performance of the valve.

2.3.1 Pull-in Voltage and Opening Pressure

Based on the steady state equation

$$\frac{\epsilon A_{es} V^2}{2} \frac{1}{(g_{es} + x)^2} + k_{sp} x - P A_b = 0, \tag{2.12}$$

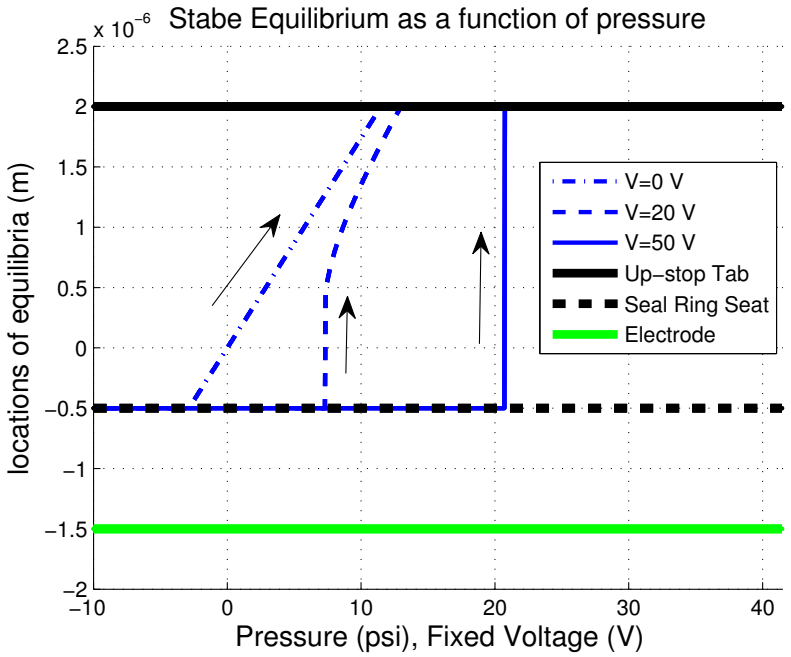


Figure 6: Equilibrium positions of boss as a function of pressure P.

or its equivalent form (assuming $x > -g_{es}$)

$$\frac{\epsilon A_{es} V^2}{2} + (g_{es} + x)^2 (k_{eff} x - P A_b) = 0, \tag{2.13}$$

we can consider V is a function of x. Intuitively the pull-in voltage happens at the location where $\frac{dV}{dx} = 0$. Taking the derivative of the last equation (2.13) with respect to x to setting $\frac{dV}{dx} = 0$,

$$2(g_{es} + x)(k_{eff} x - P A_b) + (g_{es} + x)^2 k_{eff} = 0, \tag{2.14}$$

which gives the equilibrium solution (ignoring the solution $x = -g_{es}$)

$$x_{pi} = \frac{2 P A_b}{3 k_{eff}} - \frac{g_{es}}{3}. \tag{2.15}$$

The *pull-in voltage* (as a function of P) is found by solving 2.13 and plugging in the equilibrium solution

$$V_{pi}(P) = \sqrt{\frac{8}{27} \frac{(P A_b + g_{es} k_{eff})^3}{\epsilon A_{es} k_{eff}^2}}. \tag{2.16}$$

Similarly, we consider P is a function of x and set $\frac{dP}{dx} = 0$ to find the opening pressure. Taking the derivative of steady state equation (2.12) with respect to x gives

$$\frac{-\epsilon A_{es} V^2}{(g_{es} + x)^3} + k_{sp} - \frac{dP}{dx} A_{Bosch} = 0, \tag{2.17}$$

from which we could solve for the equilibrium displacement it opens up to (setting $\frac{dP}{dx} = 0$)

$$x_{op} = \sqrt[3]{\frac{\epsilon A_{es} V^2}{k_{sp}}} - g_{es}. \tag{2.18}$$

The corresponding opening pressure (as a function of V) is given by solving 2.13 and plugging in

$$P_{op}(V) = \frac{3\sqrt[3]{k_{sp}^2 \epsilon A_{es} V^2} - 2k_{sp} g_{es}}{2A_{Bosch}}. \tag{2.19}$$

In summary, we obtained how the *pull-in voltage* V_{pi} determined by the pressure P and how the *opening pressure* P_{op} is affected by the voltage V. Their dependence relations are depicted in Figure 7. It shows the following interesting facts:

- Under a lower pressure (5 psi), a low voltage (16 V) is enough to pull-in the boss down to be closed.
- To achieve higher opening pressure (50 psi), a high voltage (140 V) is necessary to hold the boss closed.

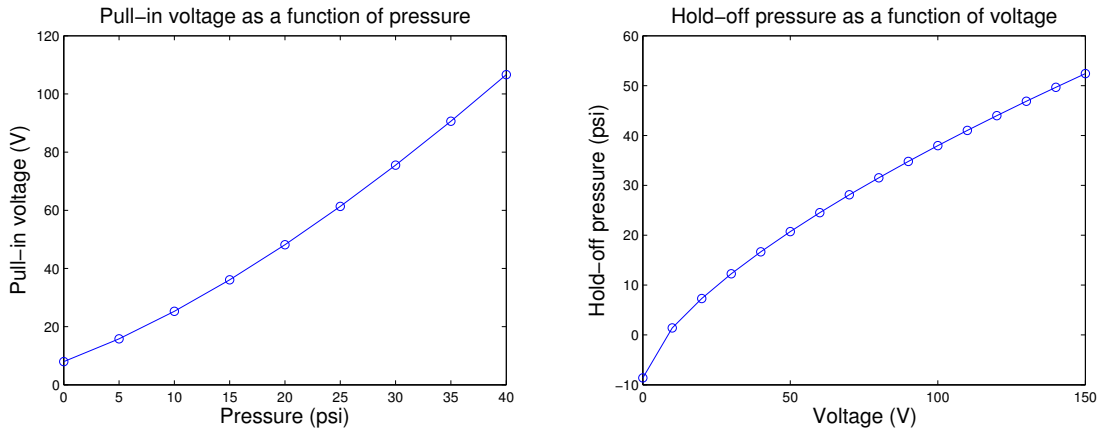


Figure 7: Pull-in voltage $V_{pi}(P)$ and opening pressure $P_{op}(V)$.

2.3.2 Design consideration for parameters

Minimizing the impact velocity and leak rate are two of the design considerations, which require us to carefully choose gaps (g_t, g_{es}, g_u) and working voltage such that:

1. The g_t gap is larger than the pull-in displacement x_{pi} to avoid large impact velocity.
2. The up-stop tab is above the opening displacement x_{op} to avoid large impact velocity. According to the expression of x_{op}, set $V = 0$ to eliminate the possibility of suddenly pop-up (unzipping) since $x_{op} = -g_{es}$ will not happen.
3. The voltage is sufficiently large in order to hold the valve closed when the pressure becomes large and decrease the leak rate.

2.4 Flow rate model

We consider the situation when the valve is opened. Steady state x_s is the neutral displacement of the boss, and it can be derived from

$$x_s k_{ef} = P_{open} A_{boss}\,, \qquad A_{boss} = \frac{R^2 + r^2}{2}\,. \tag{2.20}$$

By (2.20), x_s can be expressed as a function of P_{open}, A_{boss} and k_{ef}, i.e.

$$x_s = \frac{P_{open} A_{boss}}{k_{ef}}\,, \qquad \text{if} \quad x_s < g_u\,, \tag{2.21}$$

where R is outer radius of seal ring, r is the inner radius of seal ring, A_{boss} is the effective area, P_{open} (gage pressure) is the pressure drop across the valve boss when it is opened, and k_{ef} is the stiffness of the spring.

If the boss hit the top (i.e. $x_s \geq g_u$), we need to consider the deformation of the boss, say y, y can be expressed as

$$y = \frac{P_{open} A_{boss}}{k_{ef} + k_{boss}}\,, \tag{2.22}$$

here k_{boss} is the stiffness of the boss. The additional opening y is controlled by the boss stiffness k_{boss} and the area over which the pressure difference acts (A_{boss}).

The flow rate (Q) through the annular flow area is a piecewise function

$$Q = \begin{cases} \frac{\pi(g_{flow} + x_s)^3 P_{open}}{6\mu \ln(\frac{R}{r})}\,, & x_s < g_u \\ \frac{\pi(g_{flow} + g_u + y)^3 P_{open}}{6\mu \ln(\frac{R}{r})}\,, & x_s \geq g_u\,. \end{cases} \tag{2.23}$$

where g_{es} is the gap between boss and electrode, g_u is the gap between boss and upstop, g_{flow} is the gap between teeth and bottom, μ is the absolute viscosity.

Exmaple 1: Pressure Versus Flow
Consider the case $x_s = g_u$, flow rate becomes

$$Q = \frac{\pi(g_{flow} + g_u + y)^3 P_{open}}{6\mu \ln(\frac{R}{r})}\,, \tag{2.24}$$

by (2.22), Q can be written as

$$Q = \frac{\pi\Big(g_{flow} + g_u + \frac{P_{open} A_{boss}}{k_{ef} + k_{boss}}\Big)^3 P_{open}}{6\mu \ln(\frac{R}{r})} \tag{2.25}$$

2.5 Leak rate upper bound

For our leak rate model, we assume the valve closes completely with perfect alignment. For this case, the gap between boss and base is approximately the surface roughness of polysilicon and molecular flow models apply. If, due to slight errors in manufacturing, the valve closes imperfectly, the gap between boss and base exceeds surface roughness, pressure driven flow models apply and leak rates are higher than predicted here [5]. In [2] Galambos et. al. state that experiments indicate a cup/key valve closure reduces the leak rate more than an equivalent path increase in a flat valve closure. Hence, we calculate an upper bound to our valve's leak rate using the leak rate equation for molecular flow through a flat tube of equivalent path length and cross-sectional area. We approximate the cross-sectional area as $r_{av} \times b$, where $r_{av} = (R + r)/2$ and b is the surface roughness of polysilicon. We use a value of $b = 5$nm from [5]. We calculate the path length through the cup/key seal as $l = R - r + 2(g_{es} - g_f)n_{teeth}$, where g_{es} and g_f are as defined above and n_{teeth} is the number of teeth hanging from the boss. As defined, $g_{es} - g_f$ gives the vertical thickness of the teeth hanging from the boss. See figure (9) for clarification.

The flat valve leak rate model, from [5] and [4], is

$$Q_{leak} = C\left(1 - \frac{P_{int}}{P}\right) \tag{2.26}$$

where C is the conductance of the tube, P_{int} is the internal pressure of the MGA, P is the applied pressure and Q_{leak} is given in L/s. We can rewrite this using the gage pressure $P_g = P - P_{int}$ as

$$Q_{leak} = C\frac{P_g}{P}\,. \tag{2.27}$$

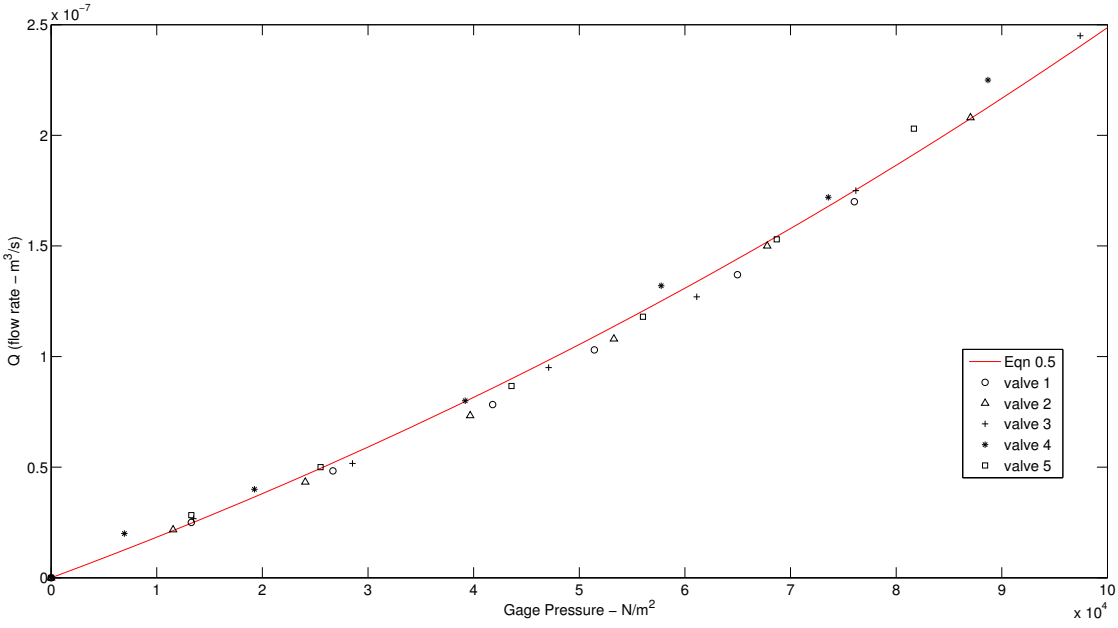


Figure 8: Comparison of measured (five values) and calculated (1.6) flow rates for preferred MB configuration. Parameters used in (1.6) are $R = 50\mu m\,, r = 47.5\mu m\,, \mu = 1.8 \times 10^{-5} Ns/m^2\,, k_{boss} + k_{ef} = 2500N/m\,, g_{flow} + g_u = 2.5\mu m\,.$

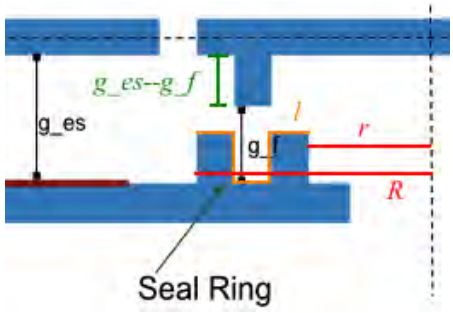


Figure 9: Close up of seal ring structure. Leak path length l is labeled in orange, radii r and R are labeled in red, and tooth width $g_{es} - g_f$ is labeled in green.

From Sandia, we have that P_g is between 13.79 kPa and 34.474 kPa (2 and 5 psi). The conductance C is given by [4]

$$C = \frac{3.44 \times 10^4}{2\sqrt{\pi}} \left(\frac{T}{M}\right)^{1/2} \frac{2\pi r_{av}\alpha^2}{l} K \ . \tag{2.28}$$

The parameters are T = absolute temperature, M = molecular weight of the gas leaking out, and K an empirical correction factor [4]. This conductance equation uses the approximation $2\pi r_{av} >> \alpha \Rightarrow 2\pi r_{av} + \alpha \approx 2\pi r_{av}$. From [5], we conclude

$$C_{H_2} = \frac{3.44 \times 10^4}{2\sqrt{\pi}} \left(\frac{T}{M_{H_2}}\right)^{1/2} = 117 \tag{2.29}$$

and

$$K = 1.444 \ . \tag{2.30}$$

From [4], we have

$$C_{air} = \frac{3.44 \times 10^4}{2\sqrt{\pi}} \left(\frac{T}{M_{air}}\right)^{1/2} = 30.9 \ . \tag{2.31}$$

These constants make our overall leak rate models for hydrogen and air

$$Q_{leak_{H_2}} = 117 \frac{2\pi r_{av}\alpha^2}{l} \frac{P_g}{P} \tag{2.32}$$

and

$$Q_{leak_{air}} = 30.9 \frac{2\pi r_{av}\alpha^2}{l} \frac{P_g}{P} \ . \tag{2.33}$$

For current design parameters in Table 1 $r = 47\mu\text{m}, R = 50\mu\text{m}, g_{es} = 1.8\mu\text{m}, g_f = 0.5\mu\text{m}, n_{teeth} = 1$, and $P = 690\text{kPa}$, these models predict leak rates

$$Q_{leak_{H_2}} = 6.8 \times 10^{-11} \ \text{L/s} = 6.8 \times 10^{-14} \ \text{m}^3/\text{s} \tag{2.34}$$

and

$$Q_{leak_{air}} = 1.8 \times 10^{-11} \ \text{L/s} = 6.8 \times 10^{-14} \ \text{m}^3/\text{s} \tag{2.35}$$

These rates are orders of magnitude lower than the specified maximum leak rate of 10^{-9} m^3/s. Therefore, under the assumption that cup/key configuration has a lower leak rate than a flat valve seat of equivalent path length, it is not necessary to consider further geometric dependence in the leak rate model.

We could possibly refine the model to include gap dependence on applied electrostatic force, i.e. a stronger electrostatic force would pull the boss even closer to the base. This dependence would further decrease the leak rate. Because our model is already out performing the specifications, we do not consider this refinement at this time.

3 Model validation/Calibration

3.1 Boss data description and cleaning process

The data that was considered for analysis in this project consisted of time series data measuring the vertical displacement of the boss as it moves from a fully open state to fully closed. These time series take place over voltages of 50, 75, and 100 volts and pressures of 2 and 50 psi. The measurements taken under 50 psi exhibit extremely erratic behavior leading our group to believe that they never reached a fully closed state. This is to be expected since 50 psi is considered to be a very high pressure value and most likely was being used to test the upper limits of pressure tolerance of the closing motion. For this reason only data sets measures at 2 psi level are being considered in this analysis as shown in Figure **??**. The raw data actually proved to be quite messy even after adjusting them to start from the same level as in Figure **??**. It seems that each set of measurements included varying levels of values taken before the closing event started which is indicate by taking a closer look of the data shown in Figure **??**. Also it appears that the starting point for each set of measurements varies. The final data set that was used in the analysis was cleaned to remove anything

Fit data:	Values:	Unit:
g_{es}	1.50	μm
A_{es}	1.00e-06	m^2
c	0.0728	Ns/m
$mass$	4.59e-09	kg

Table 3: Fit data.

that occurred before a user-chosen start point and shifted onto the same time scale. In the statistical model described later all measurements were shifted to have the same landing position and varying start positions.

In order to see the displacement travel, we select a reference point (mostly the very bottom one) for each data set, make it as the landing level, and plot all the displacement data in one Figure 10. Figure **??** shows how the velocity impacts the valve displacement.

3.2 Validation

For validation of the model we plotted our results against the test data for $2psi$ at three voltages ($50V$, $75V$, and $100V$) with design parameters assumed used for the valve testing found in Table 2. Using a damping coefficients of the same magnitude found in [3] we found similar behavior and closing time to the data sets. Figure 11 is a plot of the dynamics of the closing of the boss and opening of the boss for the design parameters.

3.3 Calibration

Once we were confident that we were capturing the dynamics of the system we then took into account that there is a level of uncertainty in many of the design parameters (due to fabrication) and the estimated damping coefficient. We fit our model to the $2psi$ and $50V$ data using a non-linear least squares fit to the following parameters g_{es}, A_{es}, m_{boss} and c.

Our resulting parameters found by the fit in Table 3 were all within 10% of the given design parameters which is the expected for fabrication variation ($\%\Delta\ g_{es} \sim 0.1$, $\%\Delta\ A_{es} \sim 7$, $\%\Delta\ m_{boss} \sim 7$). With our new parameters we then could see how well our model predicted the other data sets for input voltage $75V$ and $100V$ in Figure 12. From the figure 12 one can see that the data that we are trying to predict has some oscillations not seen in the dynamic model, this is suspected to be interference in the testing of the valve. For completeness we plotted the resulting velocity for the fit parameters versus the velocity data from which the displacement data shown in Figure 13. As one can see the velocity data is noisy and there are significant variations closer to the closing time which may account for some of the differences in behavior between the dynamic model and data.

4 Design Studies

4.1 Introduction to the optimization problem

We now focus on improving the current design of the MGA Valve by developing optimization problems. We first investigate how perturbations of the design parameters affect the air flow rate when the valve is open, as shown in Fig. 14. It can be seen that the flow rate is the most sensitive to the variation of the outer radius of the seal ring R and the inner radius r. Specifically, the flow rate increases rapidly when the overlap of the valve seat $R - r$ (the thickness of the seal ring) decreases. This is also clearly seen in equation (1.5). Comparatively, the flow rate is less sensitive to the variation in the width of the vertical gaps (g_{es},g_{flow} and g_u). Similar conclusions can also be drawn with regard to the sensitivity of leak rate, as depicted in Fig. 15. These observations motivate us to focus on optimizing the dimensions of R and r without affecting the footprint of the device, which we will elaborate in the following subsections.

The optimal design of the valve should also account for the impact velocity of the boss when it lands on the seat and rises to hit the upstop, which reduces the lifespan of the device. By using the equation (5.4),

10.

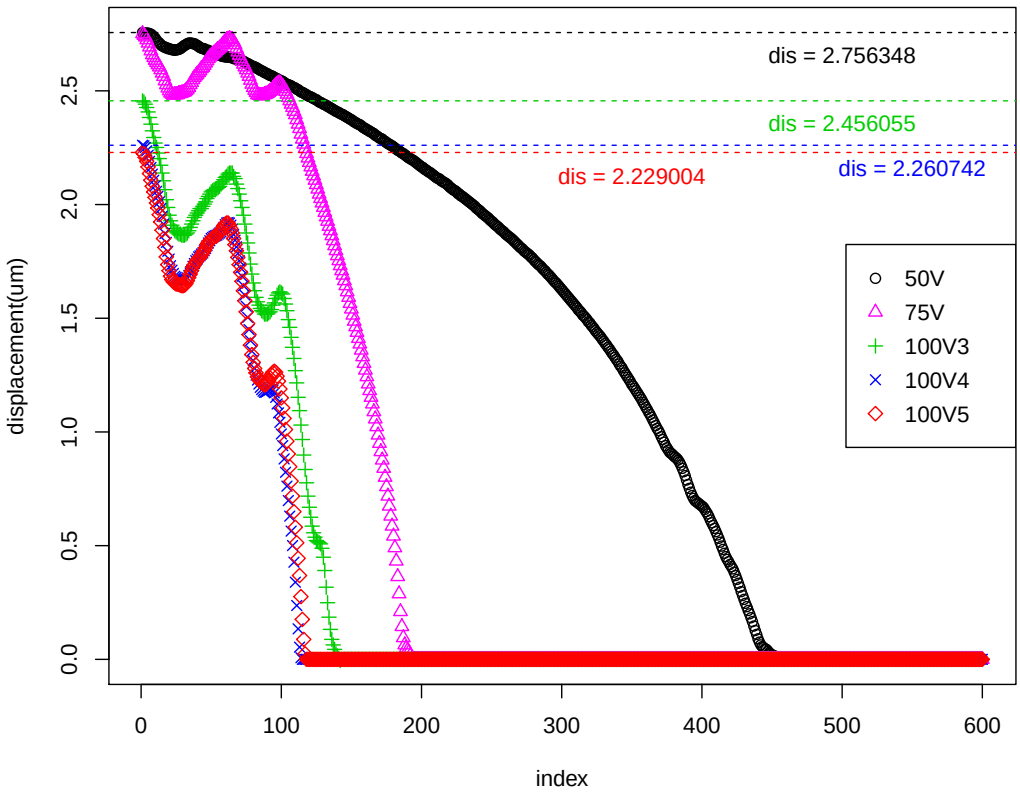


Figure 10: Data scaled to have constant landing position

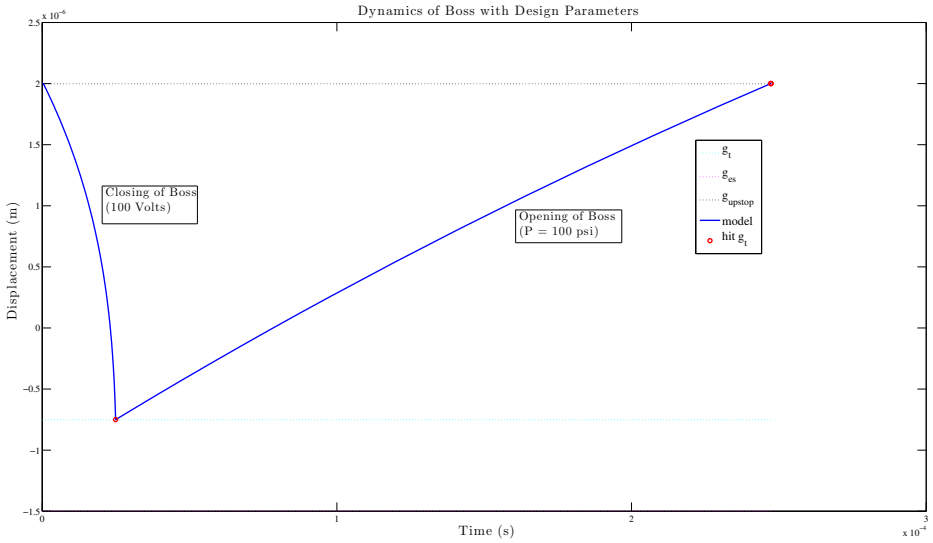


Figure 11: Dynamics of the boss for the given design parameters where bouncing was not observed.

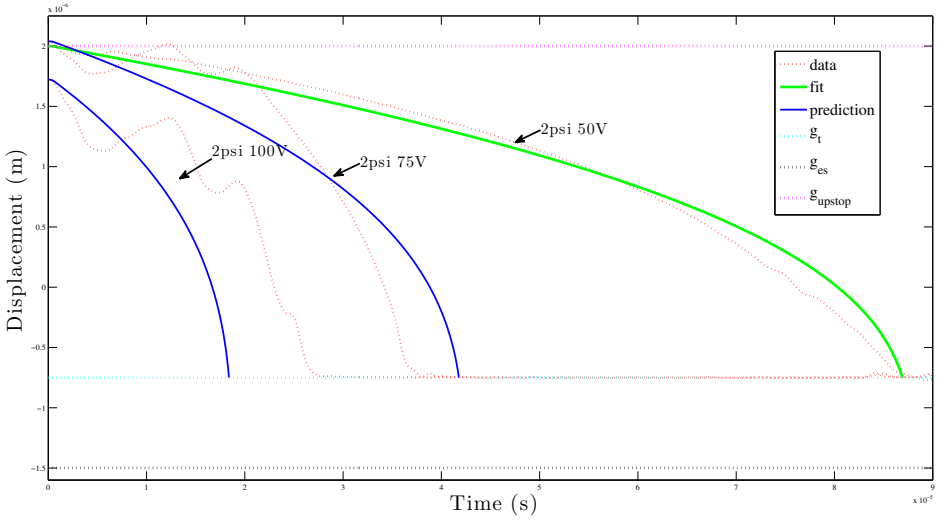


Figure 12: Green line is the displacement generated by the model using the fit parameters to the $2psi$ $50V$ data (red dotted line) and the blue lines represent the predicted results for the other input voltage.

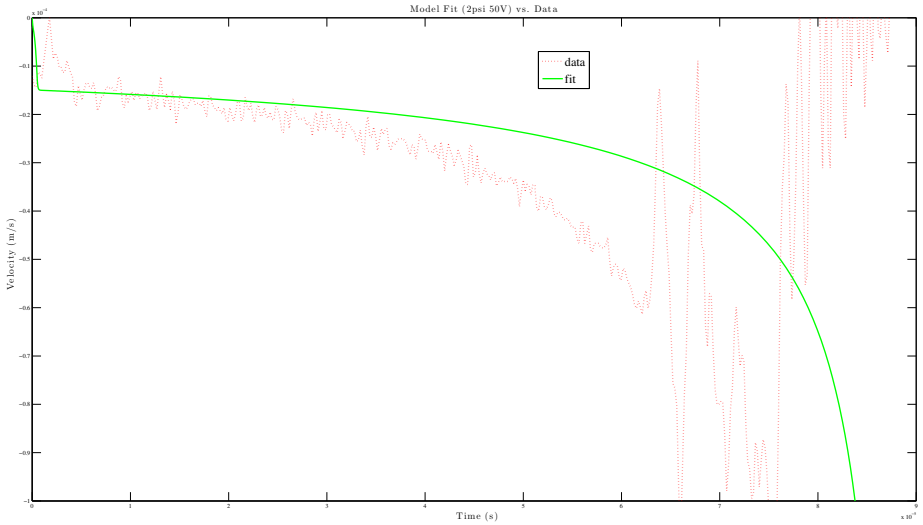


Figure 13: Green line is the displacement generated by the model using the fit parameters to the $2psi$ $50V$ data (red dotted line) and the blue lines represent the predicted results for the other input voltage.

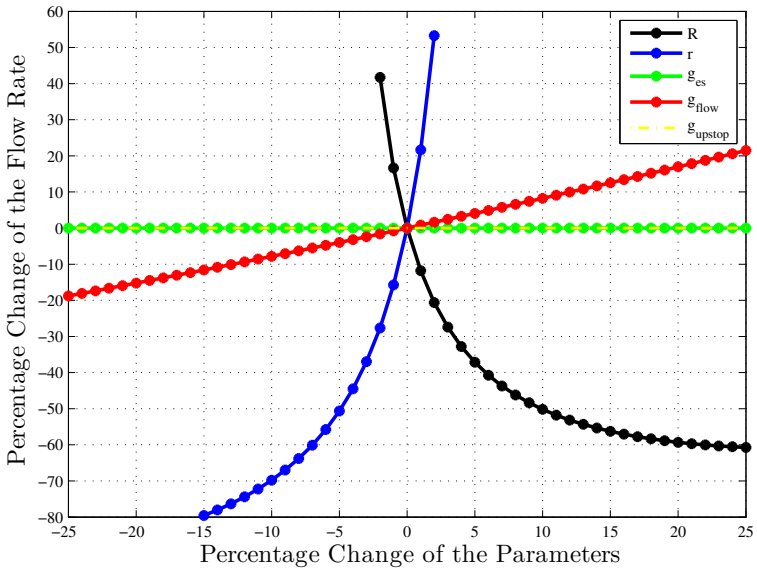


Figure 14: Sensitivity of the Flow Rate of the MGA Valve.

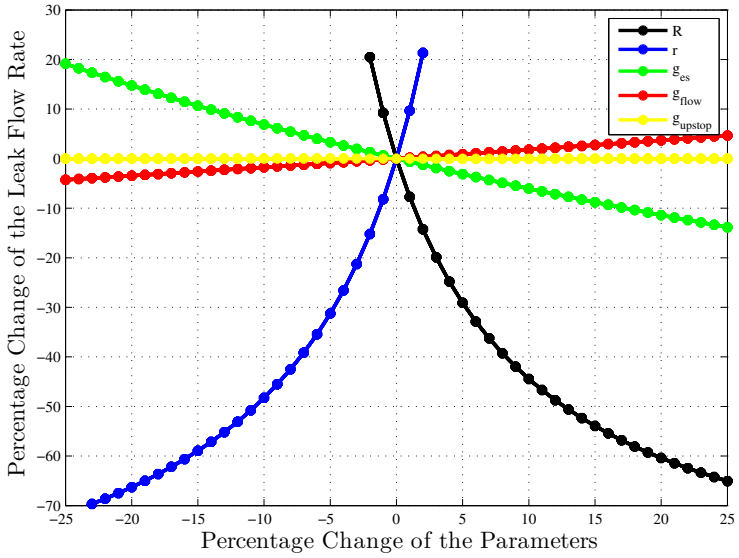


Figure 15: Sensitivity of the Leak Rate of the MGA Valve.

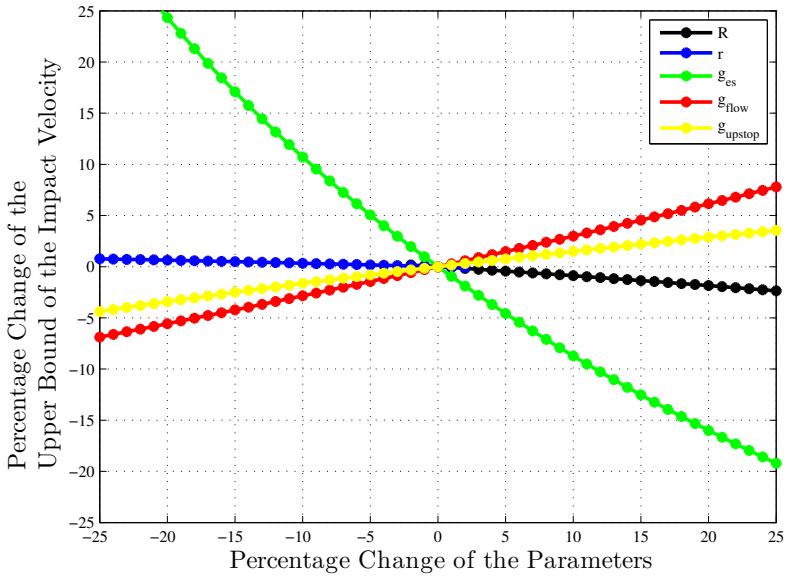


Figure 16: Sensitivity of the Impact Velocity of the MGA Valve.

we calculate the upper bound of the impact velocity and test its sensitivity with respect to the dimension parameters, as shown in Fig. 16. It is shown that a larger gap between the upper tooth and the valve seat is able to significantly reduce the impact velocity. However, the favorable dimensions of the vertical gaps have to be restricted by other practical considerations, which are also discussed later in this section. To find an optimal design we consider the following objective functions:

Objective functions

- Flow rate (Q) (1.4):

$$\text{maximize:} \qquad Q\,. \tag{4.1}$$

- 2). Leak rate (L):

$$\text{minimize:} \qquad L = C\Big(1 - \frac{P_{out}}{P_{in}}\Big) = C\frac{P_{cri}}{P_{in}} \tag{4.2}$$

where

$$P_{cri} = P_{in} - P_{out}\,,$$

P_{in} is the inside (upstream) pressure, P_{out} is the downstream pressure, P_{cri} is the desired holding pressure (maximum possible applied pressure). C is the conductance

$$C = \frac{2\pi C_{air}\, K\, \Gamma_{av}\, b^2}{R + r + 2\big(g_{es} - g_{flow}\big)n}\,,$$

where C_{air} is the slot formula constant for air, K is the fudge factor, $\Gamma_{av} = (R + r)/2$ is the average radius over seal ring, b is the surface roughness for polysilicon, and n is number of teeth.

Other objective functions considered:

- Minimizing impact velocity (may depend on steady state)
- Minimizing closing voltage (may depend on steady state)
- Minimizing footprint $(R\,, r\,, R_{es}\,, \tilde{R})$
- Avoiding circuit shorting $(g_{es}\,, g_{flow})$

The objective functions are constrained by some of the following design constraints:

Constraints

- Holdoff pressure (P_H)

  Minimum holdoff pressure $P_{cri} = 690\,kPa$.

  Given a voltage V, there is a holdoff pressure (P_H) that satisfies

$$\begin{gathered}\frac{1}{2}\epsilon A_{es}\frac{V^2}{\big(g_{es} - g_{flow}\big)^2} - k_{ef} g_{flow} = P_H A_{boss}\,,\\ P_H \geq P_{cri} = 690kPa\,,\end{gathered} \tag{4.3}$$

  here $A_{es} = \pi(R_{es}^2 - r_{es}^2)$, R_{es} is the outer radius of the electrode, r_{es} is the inner radius of the electrode.

- Leak rate (L)

  Maximum desired leak rate is $10^{-9} m^3/s$.

$$L = C\frac{P_{cri}}{P_{in}} = \frac{\pi C_{air}\, K\,(R + r)\, b^2}{R - r + 2\big(g_{es} - g_{flow}\big)n} \cdot \frac{P_{cri}}{P_{in}} \leq 10^{-9}\,.$$

- Flow rate (Q)

  Minimum desired flow rate is $10^{-7}m^3/s$.

- Voltage

  Minimum voltage require to drug the boss back from the neutral position. The electrostatic force plus spring force should be greater than the pressure force, i.e. (assuming the boss hit the top)

$$\frac{1}{2}\epsilon A_{es}\frac{V^2}{(g_{es}+g_u)^2}+k_{ef}g_u \geq P_{open}A_{boss}\,. \tag{4.4}$$

- Others constraints

$$r_{es}-R\geq 13\mu m\,(\text{to allow space for exit holes and insulation for electrode})\,,$$
$$R-r\geq 3\mu m\,(\text{to allow space for teeth})\,,$$
$$R_{es}>r_{es}\,,$$
$$g_{es}>g_{flow}\,,\ etc.$$

4.1.1 Flow rate optimization

Case 1: optimizing over radius R and r

$$\text{max:}\qquad Q(R,r)=\begin{cases}\dfrac{\pi(g_{flow}+x_s)^3P_{open}}{6\mu\ln(\frac{R}{r})}\,, & x_s<g_u\\[2ex] \dfrac{\pi(g_{flow}+g_u+y)^3P_{open}}{6\mu\ln(\frac{R}{r})}\,, & x_s\geq g_u\,.\end{cases} \tag{4.1}$$

with constraints

$$\frac{1}{2}\epsilon\frac{A_{es}}{A_{boss}}\frac{V^2}{\big(g_{es}-g_{flow}\big)^2}-\frac{k_{ef}g_{flow}}{A_{boss}}=P_H\geq P_{cri}=690\,KPa\,, \tag{4.2}$$

$$L=C\frac{P_{cri}}{P_{in}}=\frac{\pi C_{air}\,K\,(R+r)\,b^2}{R-r+2\big(g_{es}-g_{flow}\big)n}\frac{P_{cri}}{P_{in}}\leq 10^{-9}m^3/s\,. \tag{4.3}$$

$$\frac{1}{2}\epsilon A_{es}\ \frac{V^2}{\big(g_{es}+g_u\big)^2}+k_{ef}g_u\geq P_{open}A_{boss}\,, \tag{4.4}$$

constraint (3.4) means that the electric force and spring force are big enough to close the valve against pressure. We also need to consider some simple constraints:

$$R_{es}-R\geq 63\times 10^{-6}m\,, \tag{4.5}$$

$$R-r\geq 2.6\times 10^{-6}m\,, \tag{4.6}$$

The last constraint is for saving space for teeth and seal ring, one can change the difference $2.6\mu m$ based on his need.

When we solve this the optimization problem over R and r, the parameters are found in Table 4

$$g_{es}=1.8\mu m\,,g_u=2\mu m\,,g_{flow}=0.5\mu m\,,R_{es}=280\mu m \tag{4.7}$$

The numerical optimal solution is:

$$R_{opt}=55.225\mu m\,,\quad r_{opt}=52.625\mu m\,,\quad Q_{opt}=1.3072\times 10^{-7}m^3/s\,, \tag{4.8}$$

Fixed Parameters:	Values:	Unit:
k_{eff}	112	N/m
k_{boss}	2500	N/m
P_{cri}	6.9 e -5	Pa
P_{in}	7.91e-5	Pa
C_{air}	30.9e-3	
K	1.44	
P_open	3	psi
b	5e-7	m
ϵ	8.84e-12	F/m
V	100	V
m_{boss}	4.59e-09	kg

Table 4: Fixed parameters used in optimization of flow rate and leak rate.

Where R_{opt} is the optimal outer radius of seal ring, r_{opt} is the inner radius of seal ring, Q_{opt} is the flow rate at $\left(R_{opt}\,, r_{opt}\right)$. The flow rate for current MB design is

$$Q_{MB} = 6.3188 \times 10^{-8}\,,$$

current radius for the seal ring are $R_{MB} = 50\mu m\,, r_{MB} = 47\mu m\,.$

Remark: By increasing the radius around 10% we can increase the flow rate by 100%. In fact, when radius R and r are close to each other, $\ln\left(\frac{R}{r}\right)$ is close to zero. A very little decrease (10%) in R would greatly change the valve $\frac{1}{\ln\left(\frac{R}{r}\right)}$, which in turn affects the flow rate Q (100%).

Sensitivity analysis of Case 1

- Constraint Analysis
  - Flow rate Q versus minimum holdoff pressure P_{cri}

    We want to see how flow rate changes with respect to the constraint minimum holdoff pressure P_{cri} (fig. 6). The sensitivity analysis is going to give us some sense how to improve the current design. As before we fixed the other parameters in (3.7).

    The graph is very flat when P_{cri} is larger than 690kPa. We can increase minimum holdoff pressure P_{cri} without decreasing the flow rate significantly.
  - Flow rate Q versus maximum leak rate L_{max}

    The optimal flow rate Q_{opt} is irrelevant to minimum leak rate (fig. 7). This is because that given the current design gaps $(g_{es}\,, g_{flow}\,, g_u)$, constraint (3.6) is stronger than the constraint (3.3), i.e. $(g_{es} - g_{flow})\times$ number of teeth is large enough to control the leak rate based on our leak rate model.
- Parameters analysis
  - Flow rate Q versus outer radius of seal ring R

    The flow rate is a decreasing function in R, the black vertical line represents the constraint (3.6). In order to maximize the flow rate, R have to hit the constraint.
  - Flow rate Q versus inner radius of seal ring r

    Flow rate function Q is an increasing function in r, the black vertical line is the minimum difference constraints (3.6) as before. In order to minimize the flow rate, optimal inner radius r must hit the constraint (fig. 20), i.e constraint (**??**) is binding.
  - Flow rate Q versus outer radius of electrode R_{es}

    The optimal radius R_{opt} and r_{opt} depends on parameter R_{es}, so it is an function of R_{es} through R_{opt} and r_{opt}. In general, Q_{opt} is an increasing function of R_{es} (fig. 10).

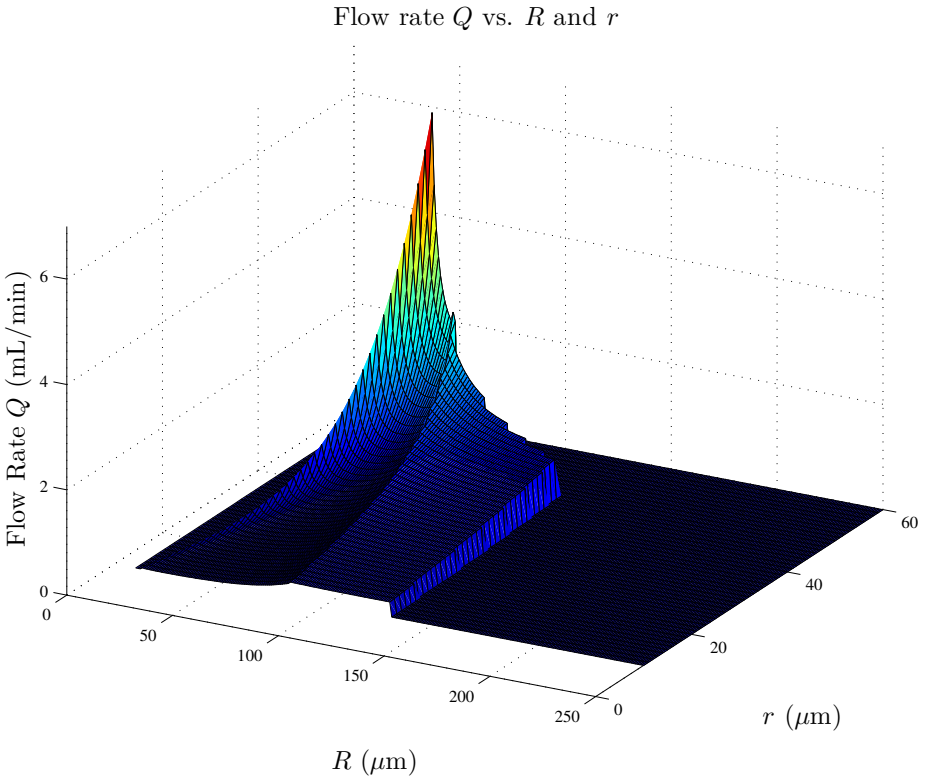


Figure 17: flow rate $Q(R, r)$ with constraints (3.2)-(3.6)

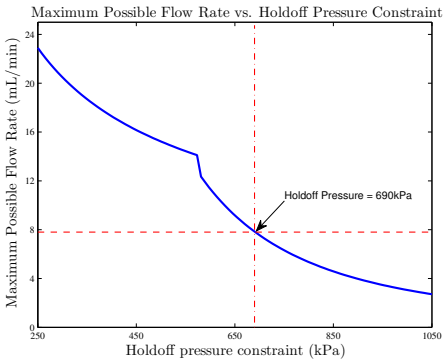


Figure 18: Optimal flow rate Q_{opt} with respect to minimum holdoff pressure

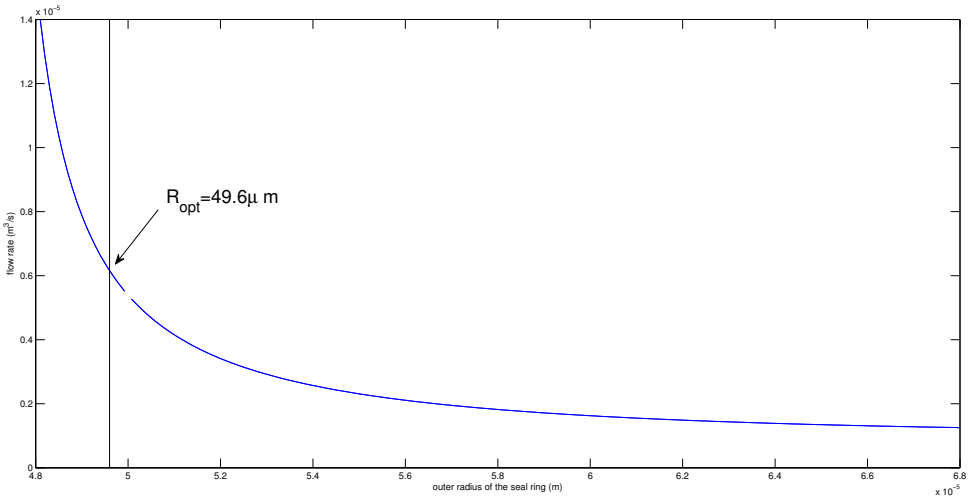


Figure 19: flow rate Q with respect to outer radius R

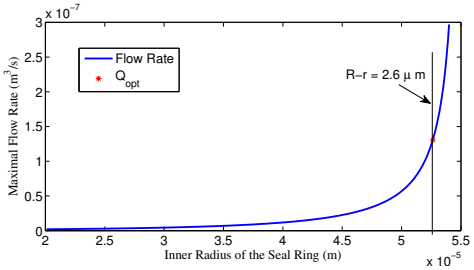


Figure 20: flow rate Q with respect to inner radius r

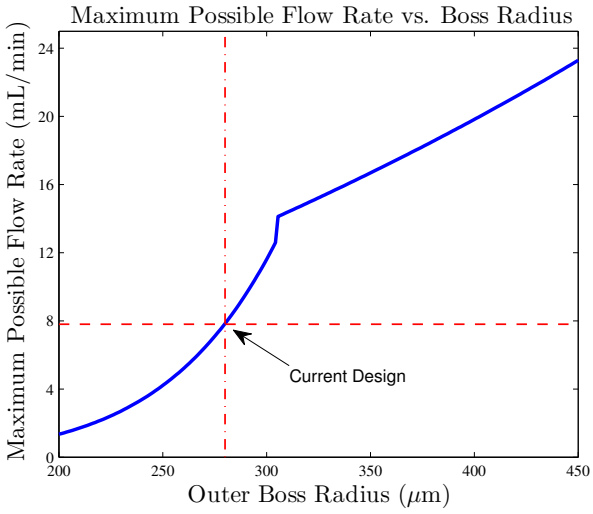


Figure 21: flow rate Q versus R_{es}

– Flow rate Q versus voltage
Similarly, flow rate Q is an increasing function in applied voltage (fig. 11)

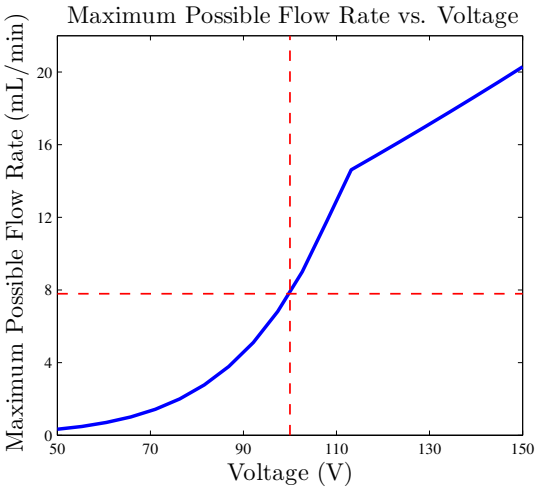


Figure 22: flow rate Q versus R_{es}

Case 2: optimizing over gaps ($g_{es}\,, g_{flow}\,, g_u$)

The objective function is same as (3.1), and we also use the same constraints (3.2)-(3.4). For this case, we fixed parameter R and r(current design), i.e.

$$R = 50\mu m\,, \qquad r = 47\mu m\,,$$

instead, we need some constraints for gap $g_{es}\,, g_{flow}$ and g_u. Since g_{flow} and g_u both contributed for flow rate, the question is that how to allocate the distance between them. Impact velocity is the velocity of the boss when it hits the bottom or top. In practice, we want to restrict (or minimize) the impact velocity to avoid the deformation of the boss. The fixed parameter values are found in Table 4.

$$A_{boss} = \tfrac{\pi}{2}(R^2 + r^2)\,, A_{es} = \pi(R_{es}^2 - r_{es}^2)\,, R_{es} = 280\mu m\,, r_{es} = 63\mu m\,, P_{closed} = \tfrac{1}{2}P_{cri}\,. \tag{4.9}$$

When we start to turn on the voltage (assuming boss remains at the top position), the energy of the system is

$$E_1 = -\frac{1}{2}\epsilon A_{es}\frac{V^2}{g_{es} + g_u} + \frac{1}{2}k_{ef}g_u^2\,. \tag{4.10}$$

When it first hit the bottom seat, the energy becomes

$$E_2 = -\frac{1}{2}\epsilon A_{es}\frac{V^2}{g_{es} - g_{flow}} + \frac{1}{2}k_{ef}g_{flow}^2 + \frac{1}{2}mv^2\,, \tag{4.11}$$

The first term in (3.11) is electric potential energy, the second term is the elastic potential energy, the last term is kinetic energy. The work done by the pressure is

$$W_1 = P_{open}A_{boss}\big(g_{flow} + g_u\big)\,, \tag{4.12}$$

and the work done by the air resistance force is denoted by W_2 which is hard to estimate. By the conservation of energy, we have

$$E_1 - W_1 - W_2 = E_2\,,$$

by (3.10)-(3.12), it yields

$$\begin{aligned} &-\frac{1}{2}\epsilon A_{es}\frac{V^2}{g_{es} + g_u} + \frac{1}{2}k_{ef}g_u^2 - P_{open}A_{boss}\big(g_{flow} + g_u\big) - W_2 \\ &= -\frac{1}{2}\epsilon A_{es}\frac{V^2}{g_{es} - g_{flow}} + \frac{1}{2}k_{ef}g_{flow}^2 + \frac{1}{2}mv^2\,. \end{aligned} \tag{4.13}$$

The kinetic energy can be expressed as

$$\begin{aligned} \tfrac{1}{2}mv^2 &= \frac{1}{2}\epsilon A_{es}\frac{V^2}{g_{es}-g_{flow}} - \frac{1}{2}\epsilon A_{es}\frac{V^2}{g_{es}+g_u} + \frac{1}{2}k_{ef}\left(g_u^2 - g_{flow}^2\right) \\ &\quad - P_{open}A_{boss}\left(g_{flow}+g_u\right) - W_2\,, \end{aligned} \tag{4.14}$$

Assume $W_2 = 0$ and the left hand side of (3.14) can not exceed $\frac{1}{2}mv_{max}^2$, we obtain the following constraint

$$\frac{1}{2}\epsilon A_{es}\frac{V^2}{g_{es}-g_{flow}} - \frac{1}{2}\epsilon A_{es}\frac{V^2}{g_{es}+g_u} + \frac{1}{2}k_{ef}\left(g_u^2 - g_{flow}^2\right) - P_{open}A_{boss}\left(g_{flow}+g_u\right) \le \frac{1}{2}mv_{max}^2\,. \tag{4.15}$$

By estimating the velocity for current design and adjust v_{max}, we can find a very nice optimal solution for $g_{es}\,, g_{flow}$ and g_u.

So far, we only consider the impact velocity at the bottom. If we consider the impact velocity at the top, we obtain another constraint as follows

$$\frac{1}{2}k_{ef}\left(g_{flow}^2 - g_u^2\right) + P_{closed}A_{boss}\left(g_u+g_{flow}\right) \le \frac{1}{2}mv_{max}^2\,. \tag{4.16}$$

After adding the two constraints into our optimization problem, the optimal solution becomes

$$g_{es}^{opt} = 3.0404\mu m\,, \quad g_{flow}^{opt} = 1.8376\mu m\,, \quad g_u^{opt} = 1.366\mu m\,, \quad Q_{opt} = 3.3758\times 10^{-7}\,, \tag{4.17}$$

where $v_{max} = 2\,m/s\,.$

Sensitivity analysis of Case 2

Constraint analysis

- Optimal flow rate Q_{opt} versus maximum impact velocity v_{max}

  v_{max} as an important factor in the constraints (3.16) and (3.17) affects the Q_{opt} positively (fig. 12)

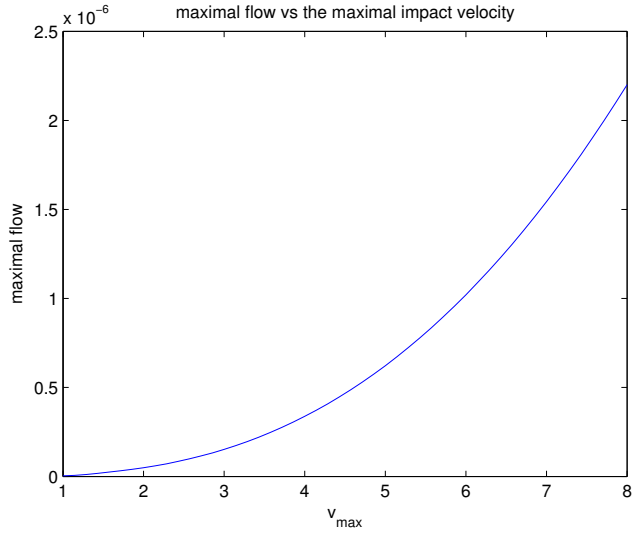


Figure 23: flow rate Q versus v_{max}

- Optimal flow rate Q_{opt} versus minimum holdoff pressure

  As in case 1, Q_{opt} would increase as minimum holdoff pressure increases (fig. 13)

  Remark: our optimal solution is not binding to the other constraints, so the optimal flow rate is insensitive to them.

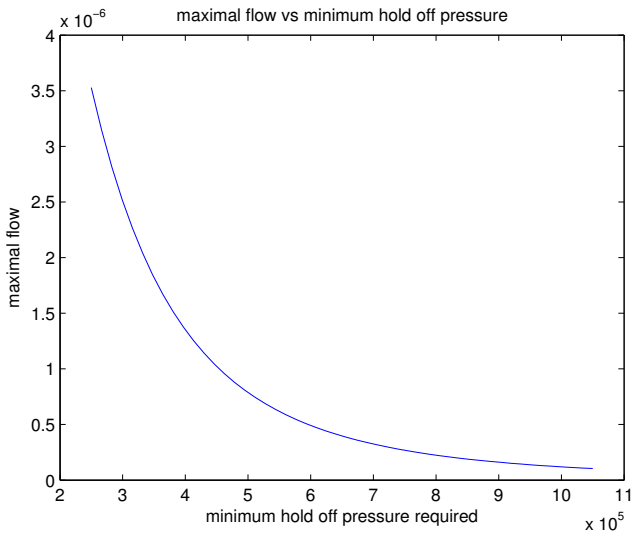


Figure 24: flow rate Q versus v_{max}

Parameter analysis

- Optimal flow rate Q_{opt} versus opening pressure(working pressure) P_{open}

  Optimal flow rate Q_{opt} is an increasing function of P_{open} (fig. 14). P_{open} appears both in the objective function and closing condition and impact velocity constraints. Obviously, opening pressure P_{open} is proportional to flow rate. It also provides an resistance force when the valve is going to be closed, while it help to decrease the impact velocity. Based on our numerical experiments, in general, it affects the optimal flow rate positively.

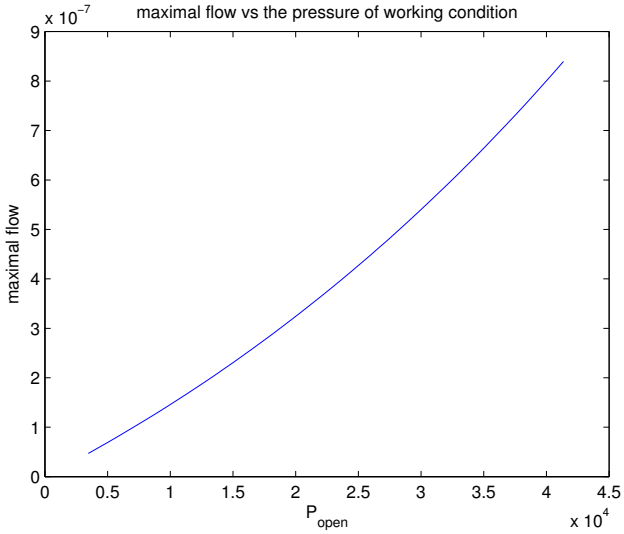


Figure 25: flow rate Q versus P_{open}

- Gap g_{es} versus flow rate Q

  The flow rate formula (1.4) has nothing to do with the gap g_{es}, either do the steady state x_s and additional opening y.

- Gap g_{flow} versus flow rate Q

  Figure 15 shows us how flow rate Q increases in g_{flow}.

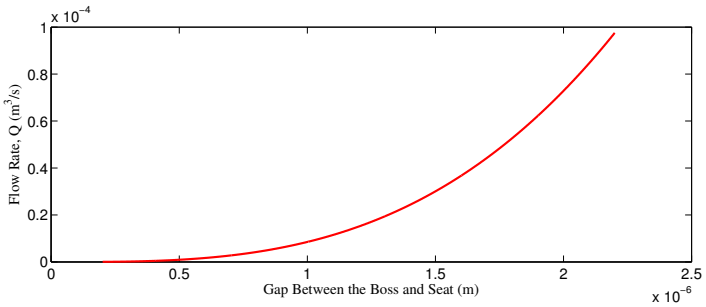


Figure 26: flow rate Q versus g_{flow}

- Flow rate Q versus gap g_u

  In the beginning (fig. 16), flow rate is an increasing function of g_u. At some point x, the flow rate jumps down to some level, x can be expressed as

$$x = \frac{P_{open} A_{boss}}{k_{ef}}\,, \tag{4.18}$$

  which is the steady state. Jump is large when k_{boss} is far greater than k_{ef}. From the design point of

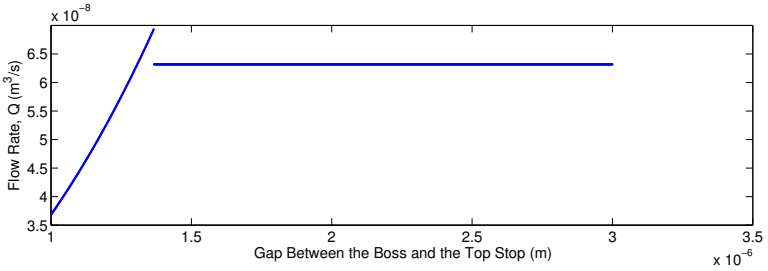


Figure 27: flow rate Q versus g_u

  view, one should enlarge g_u to some point around the steady state in order to have a better flow rate. The jump size is irrelevant to g_u, in fact

$$\text{jump} = \frac{P_{open} A_{boss}}{k_{ef}} - \frac{P_{open} A_{boss}}{k_{ef} + k_{boss}}\,. \tag{4.19}$$

4.1.2 Leak rate optimization

Case 1: Optimizing over radius R and r

$$\text{min:} \qquad L(R,r) = \frac{\pi C_{air}\, K\,(R+r)\, b^2}{R-r+2\big(g_{es}-g_{flow}\big)n} \cdot \frac{P_{cri}}{P_{in}} \tag{4.1}$$

with constraint

$$Q(R,r) = \begin{cases} \dfrac{\pi(g_{flow}+x_s)^3 P_{open}}{6\mu \ln(\frac{R}{r})}\,, & x_s < g_u \\[2ex] \dfrac{\pi(g_{flow}+g_u+y)^3 P_{open}}{6\mu \ln(\frac{R}{r})}\,, & x_s \geq g_u\,. \end{cases}$$

$$Q(R,r) \geq Q_{min} = 5\times 10^{-8}\,, \tag{4.2}$$

$$\begin{cases} \dfrac{1}{2}\epsilon A_{es}\left(\dfrac{V^2}{\big(g_{es}-g_{flow}\big)^2}\right) - k_{ef} g_{flow} = P_H A_{boss}\,, \\[2ex] P_H \geq P_{cri} = 690kPa\,, \end{cases} \tag{4.3}$$

$$\frac{1}{2}\epsilon A_{es}\ \frac{V^2}{\big(g_{es}+g_u+y\big)^2} + k_{ef} g_u \geq P_{open} A_{boss}\,, \tag{4.4}$$

$$R_{es} - R \geq 63\times 10^{-6} m\,, \tag{4.5}$$

$$R - r \geq 2.6\times 10^{-6} m\,. \tag{4.6}$$

In this model, we consider flow rate as a constraint (4.2). It means that the flow rate Q need to be greater than the value 6×10^{-8} which is around to the flow rate of the current design. The maximum leak rate constraint has been removed because we are minimizing the leak rate in this case.

As before the fixed values are found in Table 4

$$g_{es} = 1.8\mu m\,, g_u = 2\mu m\,, g_{flow} = 0.5\mu m\,, R_{es} = 280\mu m \tag{4.7}$$

The numerical optimal solution are

$$R_{opt} = 61.74\mu m\,, \quad r_{opt} = 52.29\mu m\,, \quad L_{opt} = 2.893\times 10^{-13} m^3/s\,, \tag{4.8}$$

Where R_{opt} is the optimal outer radius of seal ring, r_{opt} is the inner radius of seal ring, L_{opt} is the leak rate at $\big(R_{opt}\,, r_{opt}\big)$. The leak rate for current MB design is

$$L_{MB} = 5.295\times 10^{-13}\,,$$

current radius for the seal ring are $R_{MB} = 50\mu m\,, r_{MB} = 47\mu m$

Remark: current design is very close to our optimal solution (R_{opt}, r_{opt}), we don't need to change current design if we only want to minimize leak rate. In fact, by (fig 14), we can see from the picture that the effective region (domain for $R\,, r$ which satisfies the constraints (4.2)-(4.6)) is so small, so we will not be able to decrease the leak rate significantly. Intuitively, flow rate Q is an increasing function of R, while leak rate is a decreasing function in R. The objective function L contradicts the minimum flow rate constraint on Q which make the optimum solution stay around the current design.

Sensitivity analysis of Case 1

Constraint analysis

- Minimum flow rate constraint Q_{min}

  The optimal solution is not sensitive to the minimum flow rate constraint Q_{min} (4.2) when we decrease Q_{min}, because the constraint is not binding at the optimal. However, if we increase the Q_{min} there will be no solution to the optimization problem because of the contradiction between minimum flow rate requirement and minimizing leak rate.

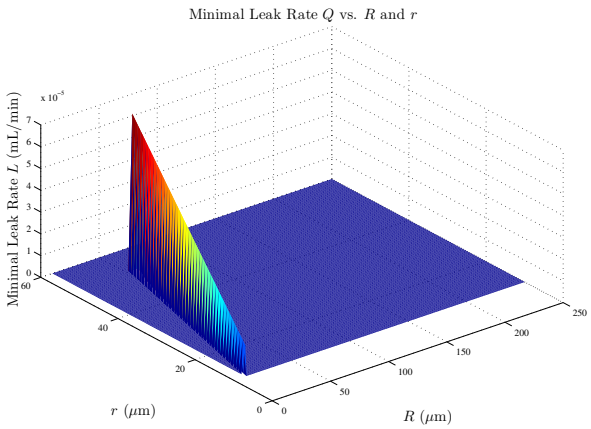


Figure 28: Leak rate L versus R, r

Similarly analysis can be found for the other constraints.

Parameter analysis

- Outer radius of seal ring R versus flow rate (L)

  Leak rate function L is an decreasing function in R (fig. 17), when we fixed the inner radius $r = 47\mu m$. As discussed before, by increasing the radius R to decrease the leak rate is not allowed due to the minimum flow rate requirement.

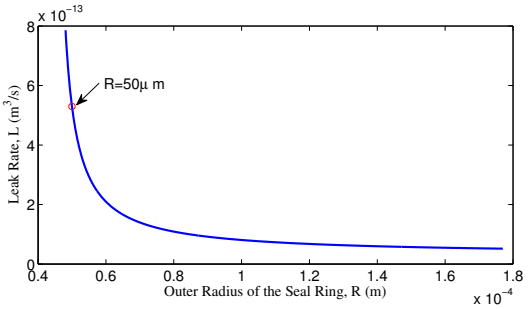


Figure 29: leak rate L versus outer radius R

- Inner radius of seal ring r versus leak rate (L)

  Leak rate function L is an increasing function in r. Again, because of the minimum flow rate constraint, one can not decrease the leak rate by decreasing the inner radius r.

Case 2: optimizing over gap $g_{es}\,, g_{flow}\,, g_u$

Based on current design data, we fixed $R = 50\mu m$ and $r = 47\mu m$ and other parameters as before, we will obtain the optimal solution as follows

$$g_{es}^{opt} = 1.897\mu m\,, \quad g_{flow} = 0.4356\mu m\,, \quad g_u = 2\mu m \quad L_{opt} = 5.006 \times 10^{-13}\,. \tag{4.9}$$

Sensitivity analysis of Case 2

Leak rate L changes as the gap g_{es} and g_{flow} change (fig. 19, 20). Same as case 1, we can not change the current design significantly because strong constraint conditions. Leak rate L is not affected by the gap g_u.

In conclusion, for customers who want to minimize the leak rate, after enforcing the minimum flow rate constraint Q_{min} around current design flow rate, there is no need to improve current design.

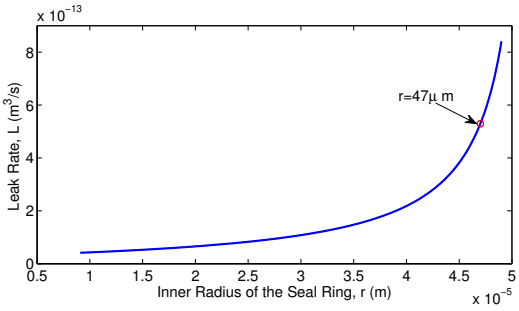


Figure 30: leak rate L versus inner radius r

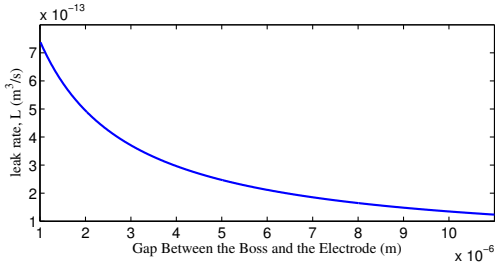


Figure 31: leak rate L versus gap g_{es}

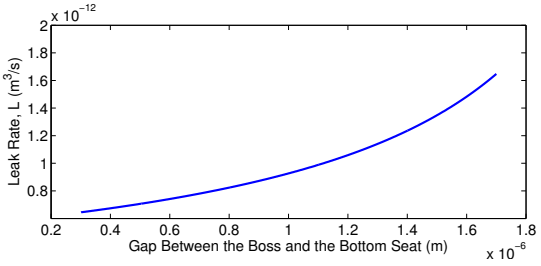


Figure 32: leak rate L versus gap g_{flow}

4.1.3 Impact velocity optimization

It is easy to find a numerical solution (velocity) to the dynamic model, while the analytical solution does not exist because of the nonlinear electrostatic force term. The energy conservation formula (3.14) gives us a way to estimate the upper bound of the impact velocity, which is

$$\frac{1}{2}mu^2 \leq \frac{1}{2}\epsilon A_{es}\frac{V^2}{g_{es}-g_{flow}} - \frac{1}{2}\epsilon A_{es}\frac{V^2}{g_{es}+g_u} + \frac{1}{2}k_{ef}\left(g_u^2 - g_{flow}^2\right) - \ P_{open}A_{boss}\left(g_{flow}+g_u\right), \tag{4.1}$$

Let v_{sup} be the upper bound of the impact velocity defined as follows

$$\frac{1}{2}mv_{sup}^2 = \frac{1}{2}\epsilon A_{es}\frac{V^2}{g_{es}-g_{flow}} - \frac{1}{2}\epsilon A_{es}\frac{V^2}{g_{es}+g_u} + \frac{1}{2}k_{ef}\left(g_u^2 - g_{flow}^2\right) - \ P_{open}A_{boss}\left(g_{flow}+g_u\right), \tag{4.2}$$

We are going to use v_{sup}^2 as the objective function, the optimization problem becomes

$$\text{minimize:} \qquad v_{sup}^2, \tag{4.3}$$

here

$$v_{sup}^2 = \ \frac{2}{m}\Big(\frac{1}{2}\epsilon A_{es}\frac{V^2}{g_{es}-g_{flow}} - \frac{1}{2}\epsilon A_{es}\frac{V^2}{g_{es}+g_u} + \frac{1}{2}k_{ef}\left(g_u^2 - g_{flow}^2\right) - \ P_{open}A_{boss}\left(g_{flow}+g_u\right)\Big), \tag{4.4}$$

with the constraints (4.2)-(4.6) and leak rate constraint (3.3).

Remark: by solving the optimization problem (5.3), we reach the goal that minimizing the real impact velocity. Figure 15 is a simple sensitivity analysis for this impact velocity upper bound (5.4), which is similar to the sensitivity analysis for the real impact velocity. This gives us a evidence that the upper bound of impact velocity v_{sup} does reflect the behavior of the real impact velocity.

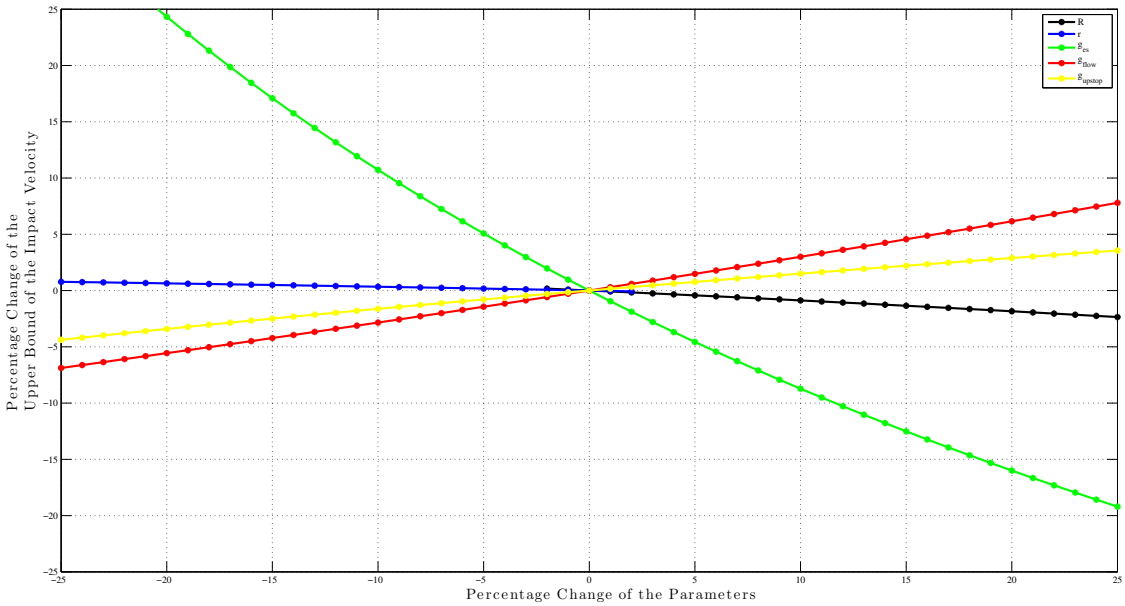


Figure 33: flow rate Q versus g_u

Case 1: Optimizing over radius R and r

Using the same fixed parameter values as before, the numerical optimal solution is

$$R_{opt} = 61.74\,\mu m\,, \quad r_{opt} = 52.29\,\mu m\,, \quad v_{max}^{opt} = 1.524\,m/s\,. \tag{4.5}$$

v_{max}^{opt} is close to the upper bound for the current design $v_{max} = 1.565\, m/s$, while R and r both change a lot, which means the optimal solution is not sensitive to R and r.

Case 2: Optimizing over gap $g_{es}\,, g_{flow}\,, g_u$
The numerical optimal solution is

$$g_{es}^{opt} = 2.617\mu m\,, \quad g_{flow}^{opt} = 1.1674\mu m\,, \quad g_u = 0.5\mu m\,, \quad v_{max}^{opt} = 1.3013 m/s\,. \tag{4.6}$$

For the current design, the upper bound of impact velocity is $v_{max} = 1.565 m/s$.

The upper bound impact velocity is decreasing function of g_{es} but increasing function in g_{flow} and g_u (fig 20). In general, we should choose a large g_u , small g_{es}, and small g_{flow}.

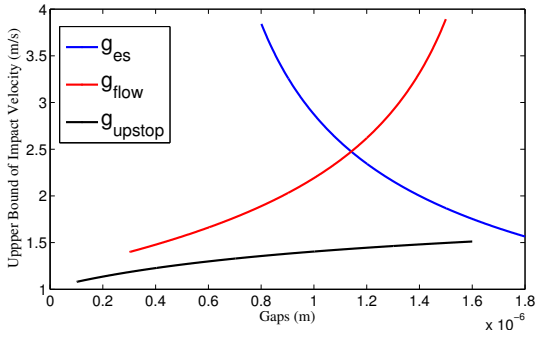


Figure 34: flow rate Q versus $g_{es}\,, g_{flow}\,, g_u$

4.1.4 Multi-objective optimization

We could consider a objective function which is a combination of flow rate Q and leak rate L and impact velocity v_{max}, for example

$$\text{maximize:} \qquad k_1 Q - k_2 L + k_3 v_{max}^2\,, \tag{4.1}$$

$k_1\,, k_2\,, k_3$ are weights for the three functions. The flow rate model is a special case of the multi-objective problem (6.1) ($k_1 = 1\,, k_2 = k_3 = 0$), same for leak rate and impact velocity model.

4.1.5 Sensitivity of Impact Velocity to Parameter Shifts

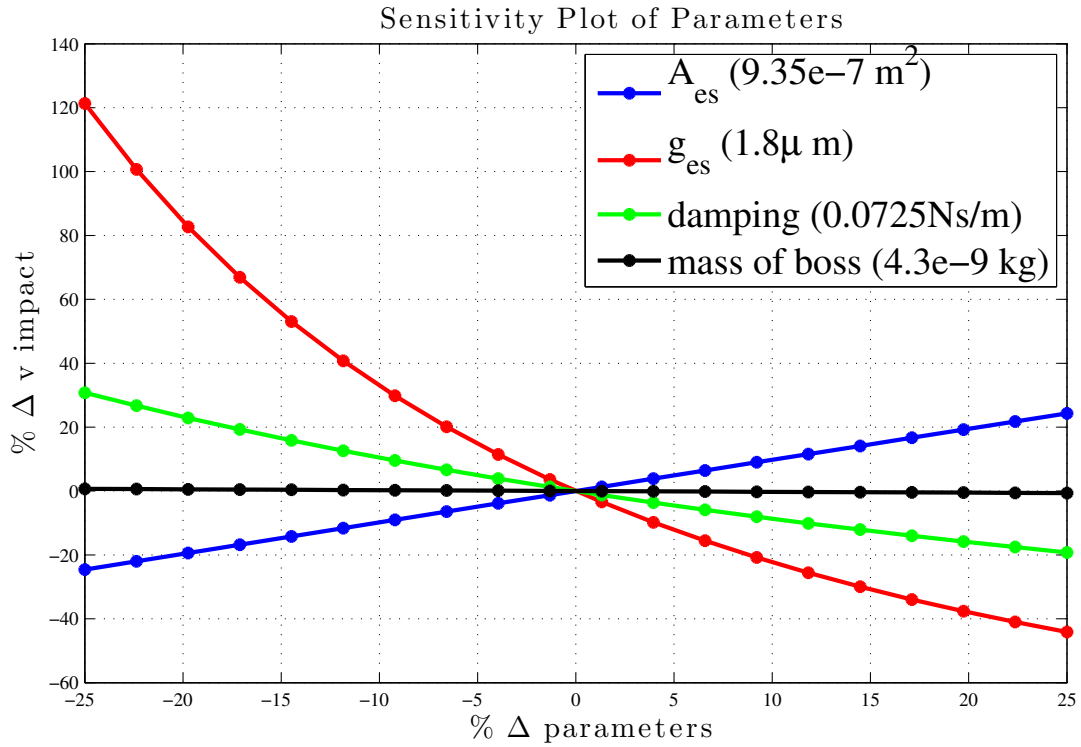
We measured the models sensitivity of the impact velocity (velocity at which the boss hits the substrate) to variations in the design parameters and inputs (Volts, Pressure). In figure X one can see that the g_{es} had the greatest effect on the impact velocity which is expected since the $F_{es} \propto \frac{1}{g_{es}^2}$. In figure X we see that as expected the voltage has a much greater effect on the impact velocity as $F_{es} \propto V^2$.

4.2 Optimization Recommendations to Sandia

We have discussed three models based on three different objective functions. The real design depends on Sandia's demand. The following table are optimal solutions for different models based on the Sandia's need.

Table 5: optimal solution over R and r

objective function	flow rate	leak rate	impact velocity
current design	$6.32 \times 10^{-8} m^3/s$	$5.295 \times 10^{-13} m^3/s$	1.5650 m/s
flow rate	$\mathbf{1.307} \times 10^{-7} m^3/s$	$6.3402 \times 10^{-13} m^3/s$	1.5435 m/s
impact velocity	$5.0006 \times 10^{-8} m^3/s$	$2.893 \times 10^{-13} m^3/s$	**1.5238** m/s


Sensitivity Plot of Parameters
A_es (9.35e−7 m^2)
g_es (1.8μ m)
damping (0.0725Ns/m)
mass of boss (4.3e−9 kg)
% Δ v impact
% Δ parameters

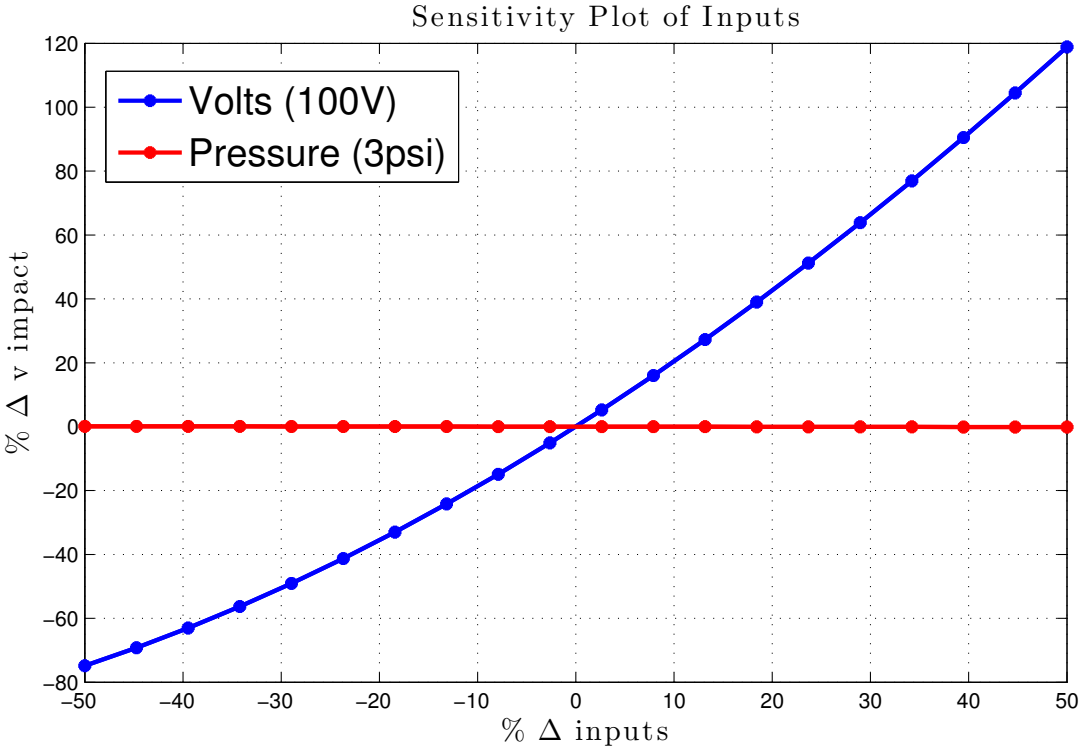

Sensitivity Plot of Inputs
Volts (100V)
Pressure (3psi)
% Δ v impact
% Δ inputs

Figure 35: .

For flow rate optimization, the optimal solution is $R = 55.225\mu m$ and $r = 52.625\mu m$. For impact velocity optimization, the optimal solution is $R = 61.74\mu m$ and $r = 52.29\mu m$. For the current design $R = 50\mu m\,, r = 47\mu m\,, g_{es} = 1.8\mu m\,, g_{flow} = 0.5\mu m\,,$ and $g_u = 2\mu m$. Impact velocity is not significant with respect to *radius*.

Table 6: optimal solution over gap $g_{es}\,, g_{flow}$ and g_u

objective function	flow rate	leak rate	impact velocity
current design	$6.32 \times 10^{-8} m^3/s$	$5.295 \times 10^{-13} m^3/s$	1.5650 m/s
flow rate	$\mathbf{3.376} \times 10^{-7} m^3/s$	$5.4863 \times 10^{-13} m^3/s$	1.6614 m/s
impact velocity	$4.9998 \times 10^{-8} m^3/s$	$5.0267 \times 10^{-13} m^3/s$	**1.3013** m/s

For flow rate optimization, the optimal solution is $g_{es} = 3.04\mu m\,, g_{flow} = 1.8376\mu m$, and $g_u = 1.366\mu m$). For impact velocity optimization the optimal solution is $g_{es} = 2.617\mu m\,, g_{flow} = 1.1674\mu m$, and $g_u = 0.5\mu m$).

Based on current design parameters, if flow rate is the first consideration, one should enlarge both the outer radius and inner radius by around 10% and increases gaps between the boss and electrode and gap between the boss and the seat while decreases gap between the boss and upstop. If impact velocity is much important than flow rate and leak rate, one should enlarge the flow gap and make the upstop gap and gap between boss and electrode small.

5 Statistical Analysis

The usage of data-driven methods to study not only the estimation of parameters but also the effect of parameter variation in complex systems has become popular in recent years. Previously, traditional parametric statistical methods were not of much use for systems that were modeled by sets of complicated differential equations. Often times these equations were highly nonlinear and could only be solved via numerical approximation. This lack of an agreeable structure to the solution was a big problem. However recent advances in nonparameteric, Bayesian, and computational methods have made analyzing these systems relatively doable. This section will detail the application of a delayed rejection adaptive Metropolis (DRAM) method to estimating the parameters for mass (m), electrostatic area (A_{es}), dampening (c), and Bosch radius (R_{Bosch}) found in the numerical solution to the dynamical model described previously.

5.1 Statistical Model

For all $i = 1, \ldots N$, consider the statistical model

$$x_i = f(t_i, V_i, P_i|\theta) + \epsilon_i, \;\; \epsilon_i \overset{iid}{\sim} N(0, \sigma^2) \tag{5.1}$$

Here x_i is the ith positional measurement associated with time t_i and voltage V_i and fit with parameters $\theta = (\theta_1, \theta_2, \ldots, \theta_M)$. N is the overall sample size of the data. The function $f(t, V, P|\theta)$ is the solution to the dynamical model described in Equation 2.2. A complication arrises here since the solution to this model, like many ODE systems, does not have a closed form and only exists via numerical approximations. Due to the lack of a closed form, we pursued a computational Bayesian approach.

The Bayesian model consists of placing a physically appropriate prior on θ (which is the same as placing prior on each θ_j in θ since we assume prior independence) and using a multivariate Gaussian likelihood. Explicitly the model is

$$\begin{aligned} \theta &\sim [\Theta] \\ \mathbf{X}|\theta &\sim \mathcal{N}\left(f\left(\mathbf{t}, \mathbf{V}|\theta\right), \sigma^2 I_{N\times N}\right) \end{aligned} \tag{5.2}$$

where $\mathbf{X}, \mathbf{t}$ and $\mathbf{V}$ represent the vectors of all responses, times, and voltages and $f\left(\mathbf{t}, \mathbf{V}|\theta\right)$ is a vector evaluated element-wise. The goal is to estimate the posterior distribution of the parameters $(\Theta|\mathbf{X})$.

The lack of an explicit closed form makes the usage of parameter partitioning methods like Gibbs sampling impossible. The traditional Metropolis-Hastings (MH) algorithm is capable of handling this problem via Markov Chain Monte Carlo estimation but suffers from many well known short comings when it comes to multidimensional parameter spaces. For an overview of MCMC methods and the MH algorithm see [6]. Chief among them is a lack of adequate exploration of the parameter space (mixing). To overcome this we chose to implement a version of a delayed rejection adaptive Metropolis (DRAM) algorithm.

5.2 Statistical Parameters Versus Model Parameters

Special care is required to alleviate potential confusion between the idea of statistical parameters and model parameters. The model parameters describe all the variables found in the dynamic model. A large of amount of uncertainty in this problem lies in the fact that we don't have adequate values for these variables for the specific valve that the data describes. Some of this is due to the inability to measure the variable due to the extremely small nature of the valve. Also there are values that could be measured or estimated via physical properties well but were not included or available. The statistical parameters are a subset of these parameters that we wish to estimate from the data while fixing the remaining model parameters to the values that we have for this valve. The statistical parameters for this model are mass (m), electrostatic area (A_{es}), damping coefficient (c), electrostatic gap (g_{es}), and unstop gap (g_u). In the notation of the model from above this means $\theta = (m, A_{es}, c, g_{es}, g_u)$.

5.3 Delayed Rejection Adaptive Metropolis

The DRAM algorithm that is used in this analysis is taken from the MATLAB implementation provided on Marko Laine's website [7]. This algorithm is a modified version of the standard MH algorithm via the addition of delayed rejection (DR) and adaptive Metropolis (AM) properties. The basic idea of a DR algorithm is that upon rejection of a proposed parameter value via standard MH, chain progression is frozen and a second and possibly more stage of parameter proposal is entered. This allows to local adaptation of proposal distributions to increase acceptance rates of proposed parameters. AM algorithms allow for global adaptation of the proposal distribution, notably the covariance matrix, based on information gained from the entire chain run up to the current iteration. This method has strong implications since destroys the Markovian and reversible nature of parameter chain. However, it has been shown that given appropriate conditions on how the algorithm adapts globally that this chain will still converge to the stationary distribution that describes the posterior parameter distribution [9]. The paper by [8] describes the DRAM method in great detail with [10] and [9] providing foundations for the DR and AM methods individually.

The only prior information we have about the distribution of the parameters found in θ is in the from reasonable physical bounds arising from the physical meaning of these parameters ($m > 0$, etc.). Operating on the assumption of prior independence, we specify the marginal priors as being uniform over their prescribed intervals. The bounds for g_u were calculated directly from the data and are the observed minimum and maximum starting positions across the 5 datasets. Table 7 contains these bounds as well as the initial values for each parameter. These initial values reflect likely values that were supported by our model validation as well as outside information. As with its' bounds, the initial value of g_u is calculated directly from the data and set to the mean of all the observed starting positions.

Parameter	**Initial Guess**	**Bounds**
m	4.3e-9 kg	[0, 10e-6]
A_{es}	9.35e-7 m^2	[0, 10e-6]
c	0.08 Ns/m	[0, 1]
g_{es}	2.5e-6 m	[0, 3e-6]
g_u	2.4907e-6 m	[2.229e-6, 2.7563e-6]

Table 7: Initial values and prior bounds for DRAM algorithm

5.3.1 Fixed versus Varying σ^2

The functionally of the *mcmcrun* code from [7] allows the user the option to keep the variance parameter of the noise term, σ^2, fixed or allow it to vary along with the other parameters. If it is specified as fixed, an estimator based off of the residuals from fitting model under the initial parameter values is used for $\hat{\sigma}^2$. This method would only be appropriate in cases where you feel that your initial parameter selection are very close to the estimates either via inspection or previous knowledge. This is seldom the case however. The alternative is to have the σ^2 a diffuse conjugate Gamma prior and consider it as part of your parameter space. Due to confusion as to the function default, the fixed case was initially considered and a disproportionate amount of time was spent on those results. Once the mistake was realized the DRAM analysis was reran using a variable σ^2. The theoretically correct approach is the variable σ^2 method so that will be the focus of the results presented here.

5.4 Results

A 5000 iteration run of the DRAM algorithm with variable σ^2 was performed using all 5 available 2 psi datasets (denoted 50V, 75V, 100V-1, 100V-2, and 100V-3). A burn in of 500 was used on all estimates. Figure 36 shows a plot of the accepted value chains for each parameter. The striking feature of this plot is just how poor the DRAM algorithm performs at traversing the parameter space, which is the exact opposite of what is known for. Delayed rejection, adaptive Metropolis, and their ilk are known for their abilities to increase the mixing properties of these parameter chains. The acceptance rate of this run is 2.3%, much too low for any statements regarding the distribution or variational properties of the parameters to be statistically valid. It is possible that a longer run would remedy this situation. A 20000 iteration DRAM run was done over an subset of the data (50V, 100V-1, and 100V-2) and still exhibited this poor mixing behavior. Ideally a much longer run encompassing the full dataset would be done to definitively answer this question but due to time constraints of the program and time constraints on virtual sessions from the NCSU Virtual Computing Lab where this is being run this is not feasible at this time. If this approach also fails to alleviate the problem, we would have to reevaluate the dynamic model. There could possibly be some identifiability issues with the statistical parameters or an unaccounted force term acting on the system.

While the mixing problems exhibited here makes any distributional statements about the parameters invalid, the algorithm itself does still provide point estimates of the parameters that provide a very good fit to the data. This is a property that arrises in traditional MH algorithms that exhibit mixing problems as well. This case here is one of the few where it is appropriate to report a post estimate without a measure of variability since we have already decided that any variability measure would be invalid. Figure 37 shows the model fit versus the data for each voltage level. The point estimates are available in Table 8

Parameter	**Estimate**
m	3.5806e-9 kg
A_{es}	1.0135e-6 m^2
c	0.1055 Ns/m
g_{es}	5.2483e-7 m
g_u	2.6285e-6 m
σ^2	1.2723e-014 m^2

Table 8: Parameter estimates of MB valve over 5000 iterations of DRAM

6 Conclusions and General Recommendations

6.1 Conclusions and Accomplishments

Our group took a multidisciplinary approach to investigating the nature of MEMS valves with regards to uncertainty and variation of design parameters. A brief list of our accomplishments during this workshop is:

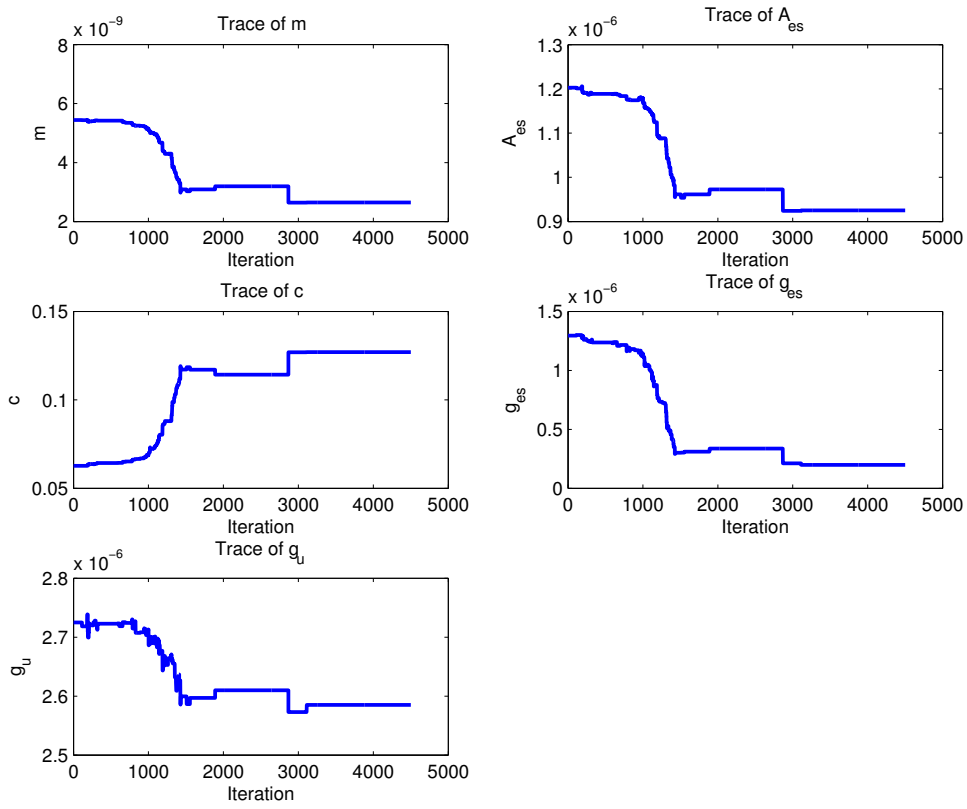


Figure 36: Trace plot of parameter chains for 5000 iterations with a burn-in of 500

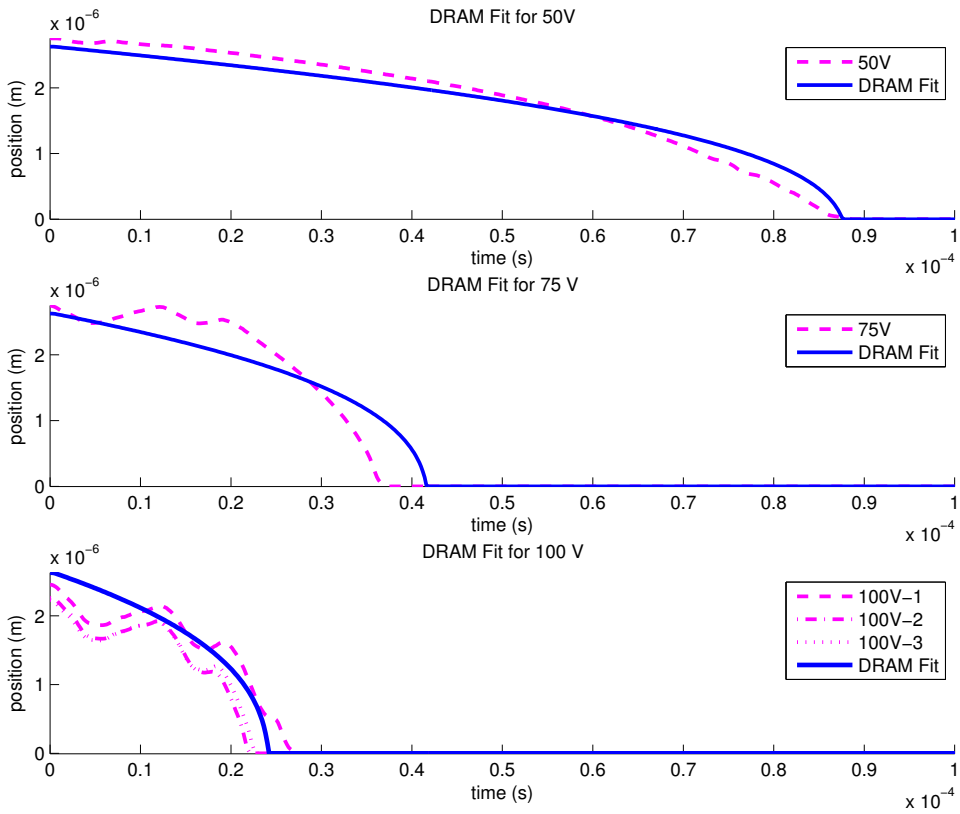


Figure 37: Fits for varying voltage levels using DRAM

- We identified relevant design parameters and quantified valve performance metrics.
- We constructed a single degree of freedom model that describes the dynamic behavior of the boss under opening and closing actions.
- Design optimization was performed using quantified metrics such as flow rate, leak rate, and impact velocity.
- Parameter sensitivity analyses were performed with regards to flow rate, leak rate, and impact velocity.
- A statistical model was developed in an effort to quantify distributional properties of parameters in order to gain greater insight.

6.2 Recommendations

A brief list of our recommendations for SANDIA is:

- Minimizing variation in electrostatic gap will reduce significant variation in valve performance
- If impact velocity is of paramount importance a low closing voltage system with a high holding voltage is preferred.
- If leak rate is the driving factor, a high voltage system will provide better performance.
- Relaxing the hold off pressure requirement while simultaneously increasing voltage, working pressure, or boss radius will most effectively increase flow rate.
- Improved data collection methodology and control would help create a richer dataset from which inference could be made for quantifying uncertainty.
-

Our work is best viewed as a comprehensive first step in understand the complex system that governs MEMS valves. Future work can uses these findings as a starting point to hopefully provide conclusions that lead to better MEMS valve designs. We would like to thank all the staff at SAMSI, NCSU, and Sandia that made our work on this project possible.

A Table of parameters

References

[1] P. Galambos, J. Lantz, M. S. Baker, J. McClain, G. R. Bogart, and R. J. Simonson, "Active MEMS valves for flow control in a high-pressure Micro-Gas-Analyzer," *J. Microelectromech. Syst.*, vol. 20, no. 5, pp. 1150-1661, Oct. 2011.

[2] P. Galambos, J. W. Lantz, C. D. James, J. L. McClain, M. S. Baker, R. Anderson, and R. J. Simonson, "Low leak rate MEMS valves for Micro-Gas-Analyzer flow control," in *Proc. IEEE TRANSDUCERS*, 2009, pp.1658-1661.

[3] M. Allen, J. Massad, R. Field, and C. Dyck. "Input and Design Optimization Under Uncertainty to Minimize the Impact Velocity of an Elecotrostatically Actuated MEMS Switch" in *Journal of Vibration and Acoustics*, vol.130, April 2008

[4] A. Roth, *Vacuum Technology*. New York: North Holland, 1982.

Parameters:	**Description:**	**Unit:**
R_b	Radius of Bosch hole	μm
r_{es}	Inner radius of electrode	μm
R_{es}	Outer radius of electrode	μm
A_{es}	Area of electrode	m^2
g_{es}	Gap between boss and electrode for new design	μm
g_{upstop}	Gap between boss and upstop	μm
g_{flow}	Gap between bottom of boss teeth and substrate	μm
g_{travel}	Gap between boss teeth and bottom	μm
t_{boss}	Thickness of boss	μm
D_{boss}	Diameter of boss	μm
m_{boss}	Mass of boss (estimate)	kg
k_{eff}	Effective stiffness of all four springs	N/m
k_{boss}	Effective stiffness of boss	N/m
l_{spring}	Length of spring	μm
w_{spring}	Width of spring	μm
t_{spring}	Thickness of spring	μm
μ	Density of polysilicon	kg/m^3
E	Elastic modulus of polysilicon	GPa
t_{teeth}	Thickness of teeth	μm
w_{teeth}	Width of teeth	μm
R	Outer radius of seal ring	μm
r	Inner radius of seal ring	μm
b	surface roughness of polysilicon	μnm
x	displacement of equilibrium	μm
y	center displacement of boss beyond upstop gap	μm
k_{boss}	flexural stiffness of boss	μm
V	Voltage	$volts$
ϵ	dielectric constant of air	F/m
c	damping	Ns/m

Table 9: Parameters.

[5] P. Galambos, C. D. James, J. Lantz, R. Givler, J. McClain, and R. J. Simonson, "Passive MEMS valves with pre-set operating pressures for Micro-Gas-Analyzer," *J. Microelectromech. Syst.*, vol. 18, no. 1, pp. 14-27, Feb. 2009.

[6] A. Gelman, J.B. Carlin, H.S. Stern, D.B. Rubin *Bayesian Data Analysis*. Chapman & Hall/CRC, 2004

[7] Laine, M. *MCMC Toolbox for MATLAB*. Accessed July 23, 2012 from http://helios.fmi.fi/ lainema/mcmc/

[8] H. Haario, M. Laine, A. Mira and E. Saksman, 2006. *DRAM: Efficient adaptive MCMC*, Statistics and Computing 16, pp. 339-354. (doi:10.1007/s11222-006-9438-0)

[9] H. Haario, E. Saksman and J. Tamminen, 2001. *An adaptive Metropolis algorithm* Bernoulli 7, pp. 223-242.

[10] A. Mira, 2001a. *On Metropolis-Hastings algorithm with delayed rejection* Metron, Vol. LIX, 3-4, pp. 231-241.

Multi-Constraint Path Planning for a Network of Autonomous Robots

Abirbhab Bandyopadhyay[1], Rana Haber[2], Sunil Thulasidasan[3], Jia Zhao[4]

Problem Presenter:
John Peach, MIT Lincoln Laboratory

Faculty Mentors:
Andreas Aristotelous, SAMSI and Duke University
Cammey Cole Manning, Meredith College

Abstract

A solution to the path planning problem for a network of autonomous robots operating under multiple constraints is presented, where the robots operate in an environment in which the knowledge of the terrain is obtained through aerial imagery. The motivating problem here is one where teams of unmanned robots compete against each other in navigating around obstacles, visiting a series of pick-up and drop-off points. Robot teams get awarded points for visiting the pick-up and drop-off points and get penalized for time spent in the arena. Thus, the goal here is to execute the task as quickly as possible, while accumulating as many points as possible. In addition, both the layout of the environment and the values of the pick-up and drop-off points can vary with time.

A complete solution requires two main components – image processing and itinerary calculation. The image processing part consists of deducing the structure of the arena from the aerial imagery, at the end of which the master ground robot has global information regarding the positions of the obstacles and points to visit. Using this, the ground robot computes a visibility graph that provides a set of feasible navigation trajectories which avoids obstacles. The second component calculates optimal itineraries for the ground robot(s) in the team, based on the path costs calculated from the visibility graph such that our constraints are satisfied while optimizing for accumulated points. Three approaches are presented for calculating the itinerary: a Brute-Force algorithm, a Greedy Algorithm, and one approach based on Linear Programming.

Simulation results are provided in terms of the actual path taken, computational time, and optimality of the solutions. The various algorithmic issues encountered and potential avenues for future work are also discussed.

1 Introduction

Navigation of unknown areas by unmanned autonomous robots is challenging, particularly for search-and-rescue missions where they might be deployed to rapidly navigate through a dynamic and hazardous terrain. In a real life search-and-rescue environment, a ground-based vehicle can be used for maneuvering through narrow passageways with the assistance of one or more air platforms; these air platforms can communicate with the ground vehicle by mapping out a path to the missing or injured person. The overhead vehicle is able to take images of the area, and the ground vehicle must

[1]Indian Statistical Institute, Calcutta
[2]Florida Institute of Technology
[3]University of Washington, Seattle and Los Alamos National Laboratory
[4]University of South Carolina, Columbia

be able to utilize the information to successfully navigate to the person in need. The previous scenario highlights the importance of the robot motion planning and justifies why its study has become incredibly broad and theoretically deep at the same time.

Path planning and navigation for mobile robots, in particular in the case where the environment is known, is a well-studied problem; see, for example, the book by Latombe [12] and the references therein. There are a number of existing maze-searching and robot motion planning algorithms, many of which only focus on successfully navigating the arena but not on performing additional specialized tasks. Controlling mobile robots in uncertain environments is of central importance to many military applications as described in [18]. Also, the automated robots were used successfully as the first responders in the World Trade Center disaster [14]. A detailed description and application of automated robots for space exploration and research and also their efficient use can potentially reduce cost and risk to humans [8]. Operating automated robots within uncertain environments has proved to be challenging; this encouraged the development of a new research field of computational intelligence [19]. Clark et al. [3] used two approaches, "environmental mapping evolution algorithm" and "stochastic learning automation approach." Both methods explored the maze for path planning and ensuring complete coverage. Kazerouni et al. [9] proposed methods which improved the solution time; their main concern was the dead ends in the maze. Ersson et al. [7] considered arcs instead of straight line paths or block-to-block traversing; they also explored the idea of re-planning the paths based on updated information of the maze and the cost constraints. Their method was shown to be suboptimal, but they did consider non-uniform obstacles. Mishra et al. [13] used Djikstra's algorithm and the Flood Fill algorithm for path planning; a drawback in their methods was the consideration of only "Manhattan" type paths.

Herein we explore the issue of determining optimal path planning according to a cost matrix that can be time dependent. In our proposed setup we allow the incorporation of multiple ground and aerial automated vehicles which are working in unison to perform certain prescribed tasks. The proposed setup allows the incorporation of features like the dynamical change of the arena layout by random movement of obstacles.

Our report is arranged as follows. First we give a detailed description of the problem at hand. Then we describe each of the methods used in planning the optimal path according to prescribed constraints. Methods considered are Brute-Force algorithm, Greedy Algorithm, and Linear Programming. We provide simulation results that support the effectiveness of the proposed methods. Finally, we conclude our report by discussing some future directions that our work might follow.

2 Problem Description

In our setup we want to see how fast robot vehicles can traverse a certain course keeping in mind that there are obstacles, rules, and points to acquire. More than one ground vehicle may enter the arena, but only one is required to navigate from the entrance to the exit and back. Scoring will be based on both the minimum distance traveled and the time to reach the goal.

The arena is 80-feet by 80-feet and is divided into (hypothetical) square grids of 4-feet by 4-feet with a wall height of 4 feet and a 20-foot high available air space. The 400 square grid has various obstacles that should be avoided by the ground vehicle; obstacles must be driven around. The obstacles in the field can be of any polygon shape and might be dynamic, e.g. a wall might open up or close at any given time. An example of an arena is described in Figure 1.

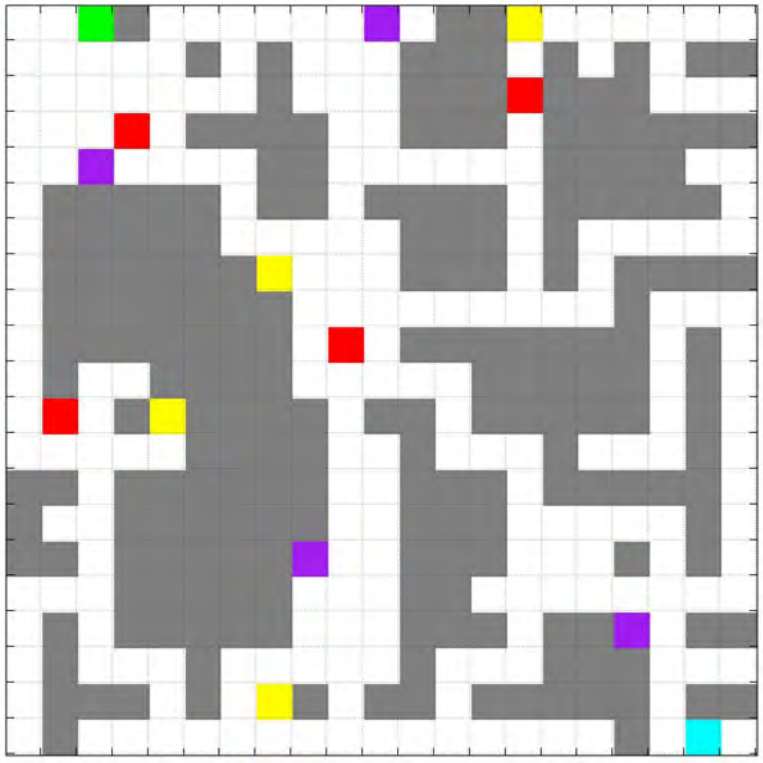


Figure 1: Example of the Arena. Walls are marked with grey squares. Yellow squares mark the drop-off points, purple squares mark the pick-up points, red squares mark the hazard points, green and cyan squares mark the entrance and exit, respectively.

A team can have several unmanned vehicles, air vehicles (UAV) and ground vehicles (UGV). The assumption is that two UAVs and two UGVs are used. The primary UAV is the *Eye in the Sky* (UAV-EITS) which has a fish eye lens with a 180-degree view and the secondary one is the *Surveillance* (UAV-Surv) which has a more focused camera with a 40-degree angle view. The primary UGV is the UGV-Prime which is a more powerful processor than the other vehicles and will maintain overall air and ground traffic control. UGV-Prime has the ability to turn on its axis and make sharp turns but it isn't as fast as the UGV-1 which is the other UGV. The UGV-1 is a smaller vehicle that is much faster but cannot make sharp turns as it does not rotate on its axis. The challenge requires that the UGVs traverse a path to several *Pick-Up Points* (PUP) and *Drop-Off Points* (DOP), which are designated in Figure 1 by purple and yellow squares respectively. In order to gain points the UGV must always visit a purple square prior to visiting a yellow one. At the purple square it should stay for 30 seconds before it can move again. Once a UGV reaches the *Exit*, it needs to return to the *Entrance* to consider the course complete. The PUP and DOP have a unique scoring system that might dynamically decrease with time and once passed through they become just another path point. Other blocks that need to be identified are red squares, which are considered as hazards. The vehicles may cross these red squares to reduce travel time, but they will have a large negative effect on the scoring. The air vehicle(s) are expected to take multiple snapshots of the maze that can be used to find the optimal path and to search for obstacles and hazards.

Our strategy for navigating successfully the arena is as follows. First, we are considering having the UAV-EITS move to the center of the field to take an initial picture of the whole obstacle which will be sent to the main computer. Then the image will be processed to trace the optimal path. The UAV-EITS will remain at the center of the field taking pictures of the course periodically to make sure that no changes have been made and that the path is still accessible. The UAV-SURV would initially take pictures at four different corners of the field, as to avoid any angles distortion taken by the UAV-EITS. After the initial pictures, the UAV-SURV will follow the UGVs at a closer range, 8 feet to 10 feet, making sure it detects better images than the UAV-EITS. It is suggested that the UAV-SURV flies from the start position to the first corner so that it computes the exact position of the start point.

3 Implementation

As mentioned before, one or more autonomous UAVs periodically photograph the terrain such that both an *Eye in the Sky* view and detailed terrain information is available to the UGVs. The initial parts of the image processing consists of removing noise and distortion from the aerial imagery and converting the 3-D aerial imagery to a flattened 2-D representation. The 2-D image is then used to deduce the locations of the obstacles, the pick-up and drop-of points and the locations of exits, and hazards, if any. The entire processing of converting a real-world 3-D image to a maze-like 2-D representation in and of itself is an algorithmically challenging task, which largely lay outside the scope of this project. Due to this and project time constraints, we restricted our task to begin at the stage where we already have the 2-D arena map. To simulate this process, we generated our own test mazes, which we describe below.

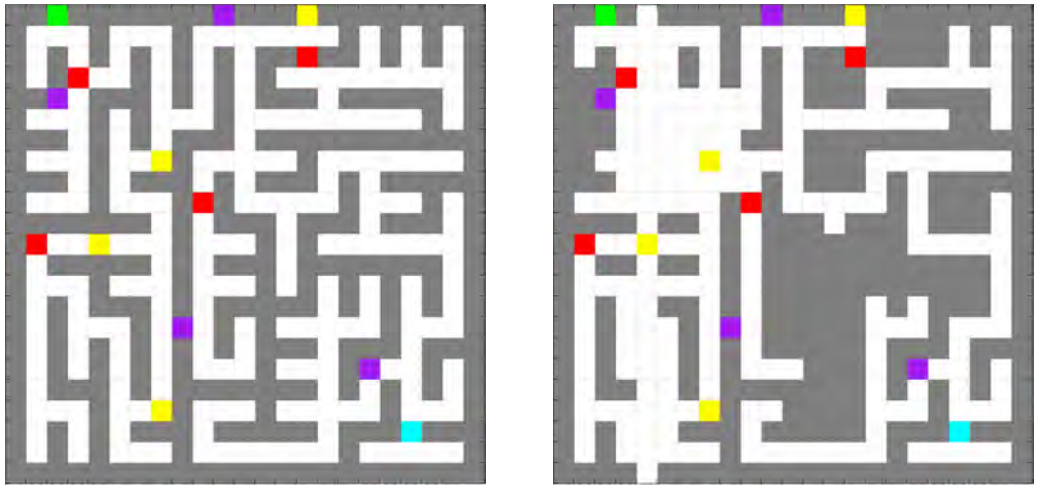


Figure 2: Example of a Maze (left) and a modified Arena (right)

3.1 Arena Simulation

Sample maps of the arena were generated by using standard maze generation algorithms such as Depth-First Search [10], Randomized Kruskal [11], and Randomized Prim's algorithms [17], and then modifying the result by removing some interior walls and filling some dead ends until the result more resembles a top view of a city rather than a traditional maze [1]. The walls and pathways are 4-feet wide, which was discretized into 4-foot by 4-foot squares, with every individual square representing a wall or a path area.

3.2 Image Processing

Dead-End Removal Even with some random dead-end filling, the arena still contains dead end paths that do not contain pick-up or drop-off points. For the purpose of planning a path, these dead ends contribute nothing, and in fact, add corners that affect the visibility graph (described below). To reduce running time of finding the shortest path, we use an image smoothing operation which fills these dead end paths. Figure 3 shows the difference between the original maze and the modified arena with fewer valid paths.

3.3 Corner Detection

Before the task of path planning can be executed, a representation of the environment in terms of accessible and non-accessible areas has to be created. For a ground vehicle, this is a 2-D plane where obstacle regions are represented as closed polygons that are completely contained inside a larger boundary. The vertices of the polygons represent the corners of the obstacles. Then all regions that lie entirely outside the polygons are available for ground locomotion, and for navigating around obstacles, the robot follows the straight line path that connects two corners that are mutually visible from each other. We describe this *Visbility Graph* in greater detail in Section 3.4; here, we describe the process of detecting corners from an image on which the pose estimation and flattening has already been applied.

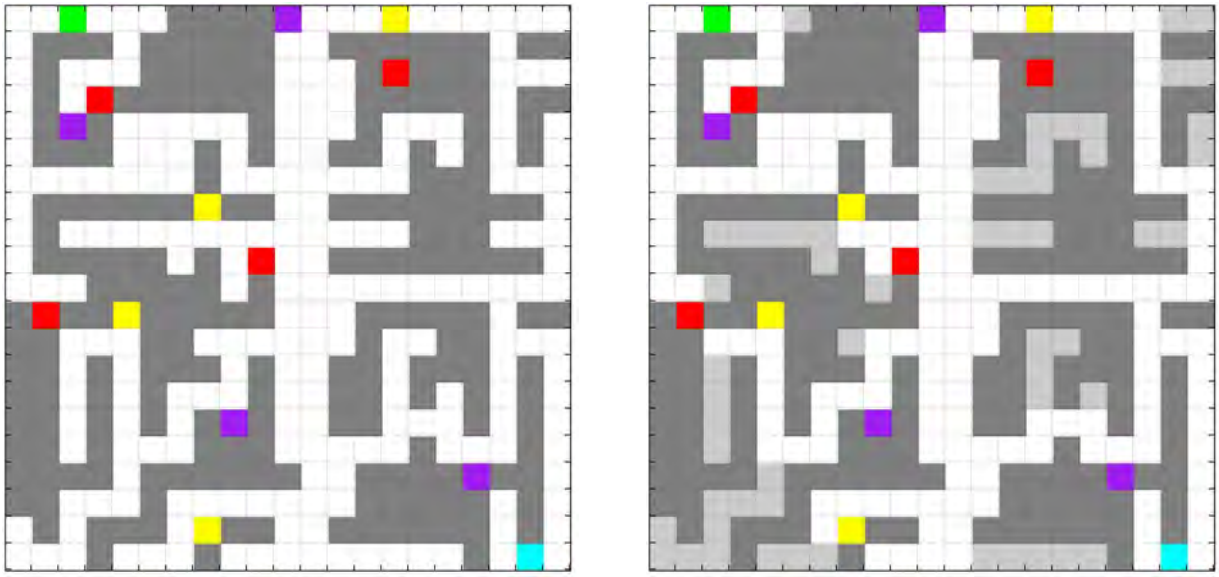


Figure 3: A Comparison Between Former Map with the Smoothed Map

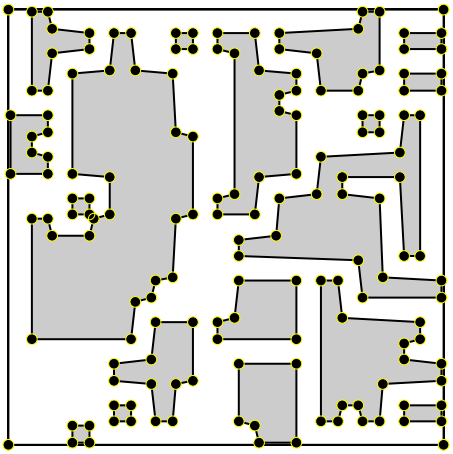


Figure 4: Corners of the Environment Delineate the Boundary of the Obstacles

The input to the corner detection algorithm is a binary matrix, where a 1 entry in the i, j-th position indicates that the corresponding spatial location in the flattened image is occupied by an obstacle, and a 0 entry implies free space. We use MATLAB's `bwlabel` function to group and label connected objects. Below we show an example of this procedure, starting out with binary matrix A which contains two obstacles (denoted by 1s).

```
A =

     1     1     0     1
     1     1     0     1
     0     0     1     1
     0     0     1     1

>> bwlabel(A,4)

ans =

     1     1     0     2
     1     1     0     2
     0     0     2     2
     0     0     2     2
```

The output of `bwlabel` is a matrix where entries belonging to the same object are given the same label. For a given object, corner detection then reduces to traversing the edges of an object and detecting where the transitions occur. The output of the corner detection algorithm is a list of polygonal vertices (Figure 4), which can then be used to construct the visibility graph

3.4 Visibility Graph

In computational geometry, a visibility graph [6] is a graph constructed from a set of locations and obstacles in a plane (or higher dimensional space) such that the set of vertices in the graph are the vertices of the obstacles and an edge exists between any two mutually visible locations.

More formally, if $G = (V, E)$ is an undirected graph, then for any $v_i, v_j \in V, e_{ij} \in E$ if and only if the line joining v_i and v_j does not pass through an obstacle. Intuitively, paths in the visibility graph represent the set of physical trajectories that a moving object can take without passing through an obstacle.

Visibility graphs are a highly useful construct in robot motion planning when the terrain knowledge is available in advance. Obstacles in the terrain are represented as polygonal objects, whose vertices define the set of vertices V in our visibility graph G_v. Once the visibility graph is constructed, finding an optimal path in the visibility graph reduces to finding the shortest-cost path between any two vertices in G_v; for this, any standard shortest-path algorithm can be used. In this project, we use the classic Dijkstra's shortest path algorithm [5].

Since edges in the visibility graph are straight line segments, the shortest path in the visibility graph follows the straight line segments except at the vertices of the obstacles, where the path assumes a new direction. Since robots have non-zero size, to prevent the robot from actually colliding with the corner of the obstacle, object sizes are expanded to include a buffer region around the actual object;

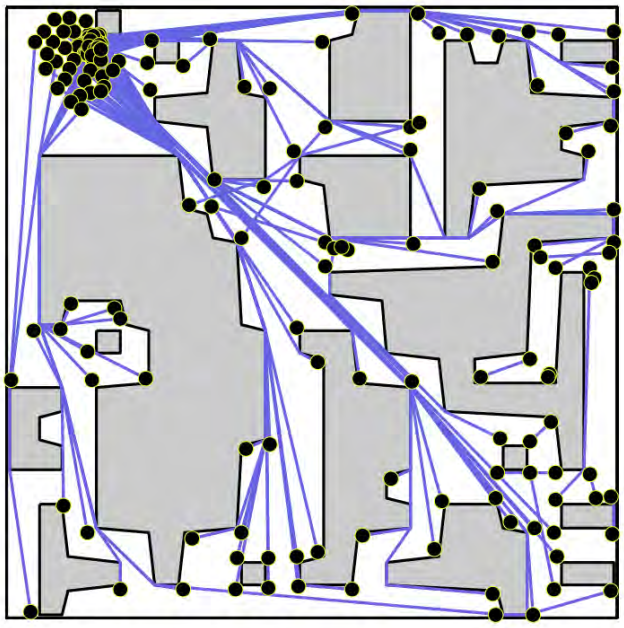


Figure 5: A subset of the visibility graph in a multi-polygon obstacle environment. The cluster of points in the top right corner are a set of starting locations that are also vertices in the visibility graph.

the corners of the expanded obstacles will then become the vertices of the visibility graph.

An $O(n^2)$ algorithm for constructing the visibility graph in a plane of n non-intersecting line segments is described in [20]. For this project, we used the Visilibity library [16], a free and open source computational geometry library written in C++, that computes the visibility graph of a given polygonal environment, and can then be used to find the optimal path between any two locations. A subset of the visibility graph for one of the test environments in our project is shown in Figure 5.

3.5 Algorithms for Finding Optimal Routes

Based on our literature search and experimental comparison, we tested several route planning methods which were the Brute-Force method, the Greedy Algorithm and Linear Programming. Performance of each of them depend on the numbers of PUPs and DOPs. From our experimental simulation, we showed that the Brute-Force method and Linear Programming could be effectively used for small numbers of nodes and both will provide optimal solutions. As the number of nodes increases, a Greedy Algorithm proved to be much faster, but at a cost of returning sub-optimal solutions.

The following notation will be used in the next few sections: **P** denotes the set of pick-ups and **Y** the set of drop-offs, the **S** represents the start point and **E** is the exit point.

3.5.1 Brute-Force Search

Brute-Force search, although trivial, is still widely-used for its simple implementation and effectiveness. Brute Force will always find a solution when it exists. The algorithm searches through every possible route while calculating and comparing the cost until it finds an optimal solution. The following pseudo-code uses Brute Force to solve our problem:

```
function [best_route, least_cost]=brute_force_search(cost_matrix,S,E,Y,P)
        possible_route=find_all_possible_route(S,E,Y,P);
        costs=cost_for_the_relative_route(possible_route);
        [best_route,least_cost]=the_optimal_route(possible_route, costs);
end
```

Brute Force is not computationally efficient when the problem size becomes too large, since the computational complexity grows combinatorially, hence the need for more efficient methods.

3.5.2 Greedy Algorithm

One of the choices of a medium sized problem is the Greedy Algorithm, which is widely used in data-mining and machine-learning field and gives good results that are often nearly optimal.

The Greedy Algorithm returns the locally optimal solution at each step, which in general leads to a good feasible solution, if not optimal [4]. In our implementation, each robot starts from the Entrance and then moves to the next unvisited least cost node, with the constraint that the vehicles maintain a non-negative purple-to-yellow square count. This count increases by +1 when the vehicle stops at a purple square and decreases by -1 at a yellow square. A valid path may visited several purple squares in succession, but the count must never be less than zero. The Greedy Algorithm based pseudocode for route search is given below:

```
function [best_route, least_cost]= greedy_search(cost_matrix,S,E,Y,P)
        visited_marker = [];
        while ( have_some_unvisited_points)
                next_stop = next_least_cost(cost_matrix,visited_marker);

                if( is_legal(next_stop))
                visited_marker = [visited_marker, next_stop];
                else
                        marker_it_unreachable(next_stop);
                end
        end
```

3.5.3 Multi-Robot Based Greedy Algorithm

With the possibility of vehicle failure, a multi-vehicle solution seemed prudent. The multiple traveling salesman problem is discussed extensively in the literature, e.g. [2] as has the multiple vehicle routing problem [15], but the problem of visiting nodes in a particular order is not well covered.

For this project, we applied a multi-vehicle based Greedy Algorithm, which guides several vehicles to diversely spaced nodes simultaneously. The Greedy Algorithm is well suited to a dynamic maze in which walls move during the course of the vehicle traversing the path. Should a planned path close, or a previously walled off area open, the Greedy Algorithm may be quickly re-run using the robot(s) current positions as the new starting points.

3.5.4 Linear Programming

Determining the optimal path sequence through the nodes (Entrance, PUP, DOP, and Exit) may be formulated as a Linear Programming (LP) problem. The general Linear Programming problem is expressed as

$$\text{minimize } c^T x$$

$$\text{subject to } Ax \geq b$$

$$\text{and } x \geq 0$$

where x is a vector of variables to be determined, c is a cost vector, A is a coefficient matrix, and b is a vector of constraints. It can be shown that the solution region is bounded by a 2-dimensional polytope and the optimal solution exists at one of the vertices. For this problem, x represents the transition matrix from node to node, or may be viewed as the edge graph for the optimal solution. The problem of finding an optimal path is equivalent to solving a travelling salesman problem which is known to be NP complete meaning that a solution may be verified in polynomial time, but there is no fast method to generate a solution. As such, finding an LP solution is expected to grow exponentially with the number of nodes in the problem.

The rules of the problem add further complexity due to the fact that at least one more pick-up point must be visited than drop-off points. That is, a ground vehicle must have an object on board obtained at a PUP before arriving at a drop-off location.

1. The ground vehicle must begin at the Entrance.
2. It must stop at at least one PUP prior to visiting a DOP.
3. All ground vehicles should stop at as many PUPs and DOPs as possible to accumulate the maximum number of points.
4. Once the vehicle stops at a PUP or DOP, that point becomes a regular grid and no further points may be obtained if it is revisited.
5. At least one ground vehicle must stop at the Exit, and then return to the Entrance.
6. Hazards may be crossed, but at a penalty.

In the solution, the entries of the transition matrix x are either 0 or 1 where $x_{ij} = 0$ means that a path from node i to node j is not possible, while a path exists if $x_{ij} = 1$. Note that a constraint is that both x_{ij} and x_{ji} cannot both be 1 because this would imply that it is possible to travel from one node to another and then back. This constraint may be sharpened by requiring that

$$x_{ij} + x_{ji} \leq 1$$

since the optimal solution will not have a path between most pairs of nodes (just the ones constituting the actual path). The full set of constraints are the following:

Number of PUPs	Sequences Without Exit	Sequences with Exit
1	1	3
2	8	40
3	180	1260
4	8064	72576
5	604800	6652800

Table 1: Sequence Numbers with Growing PUPs and DOPs

1. $x_{ii} = 0$ for all i. A path cannot loop from a node back to the same node.
2. $x_{ij} + x_{ji} \leq 1$ for all $i \neq j$. At most one path may lead from one node to another.
3. $\sum_{i=1}^{n} x_{ij} = 1$ for all j. Exactly one path arrives at a node.
4. $\sum_{j=1}^{n} x_{ij} = 1$ for all i. Exactly one path leads away from a node.
5. The solution may not be permutable into a bipartite graph. In other words, solutions with more than one complete circuit are not permitted.

Generating the matrix A for this last constraint required a rapidly increasing number of equations as the number of nodes increased. For 6 nodes this resulted in 55 constraints, 8 nodes required 500 and for 10 the number reached 8865. Finally, the output may not satisfy the condition that the number of PUPs visited must exceed the number of DOPs visited so far on the path. If such a solution is obtained, the path cost for a set of nodes violating this condition is increased and the program rerun. Many routines are available for solving Linear Programming problems, and the one chosen for this project was Matlab's bintprog function.

3.6 Results

From Table 1, we can easily see that when the numbers of PUP, which is equal to the number of DOP, is no more than 4, it's more efficient to use the Brute-Force approach, since it is more efficient in the computation efficiency.

The comparison between computational times of those three methods is given in the Figure 6.

For our one agent based greedy algorithm, Figure 7 shows the path, which the single robot follows during its work. Figure 8 shows the separate path of each robots, when robots are working simultaneously. From Figure 9, we could see our multi-agent based greedy algorithm reduces the paths cost efficiently with the number of robots growing.

4 Future Work

The optimal path solution for the ground vehicle traversing the arena from entrance, through the pick-up points, drop-off points and exit requires processing one or more aerial images of the environment, converting these images into a "bird's-eye" view, then applying edge and corner detection to identify the non-navigable areas, and color detection to identify the PUPs and DOPs as well as hazards. The

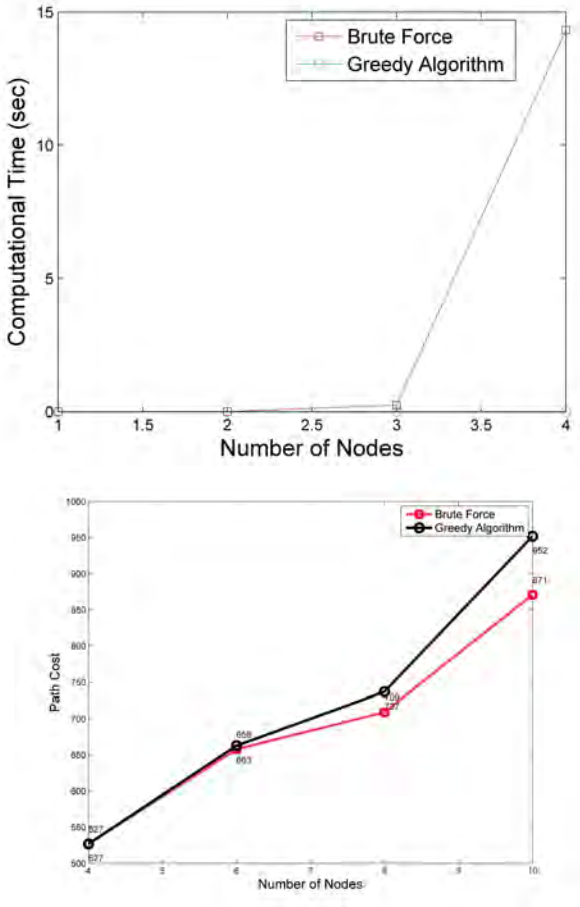


Figure 6: The Time Difference (left) and Cost Difference(right)

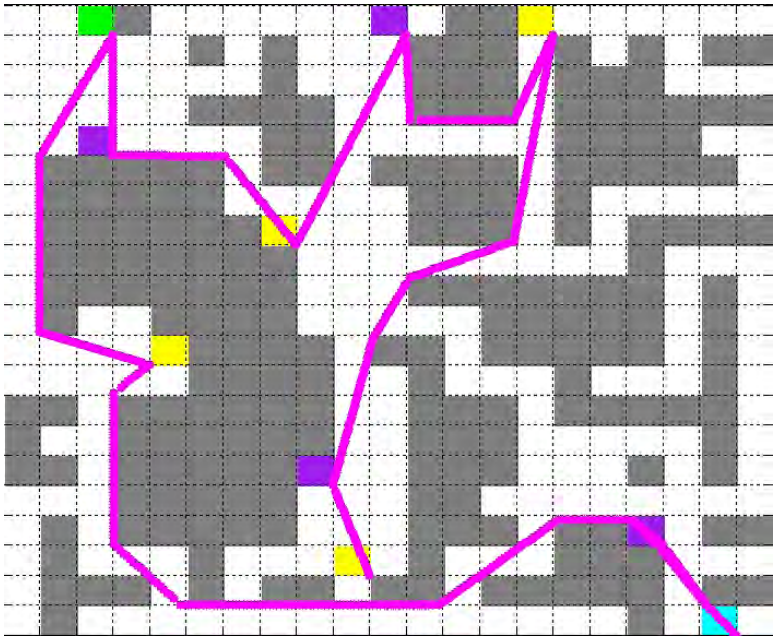


Figure 7: Path of One Robot

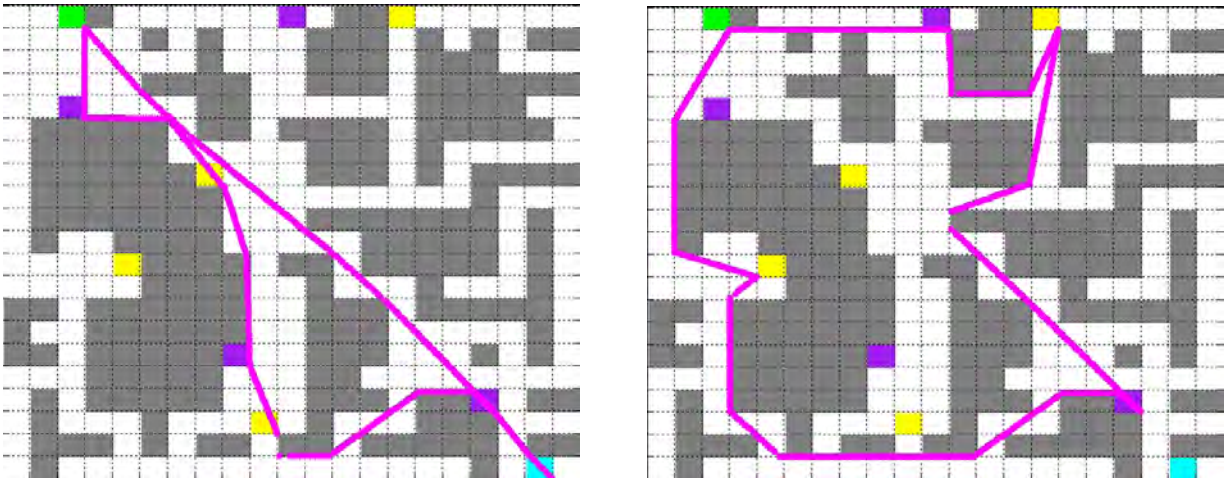


Figure 8: Paths of Two Robots

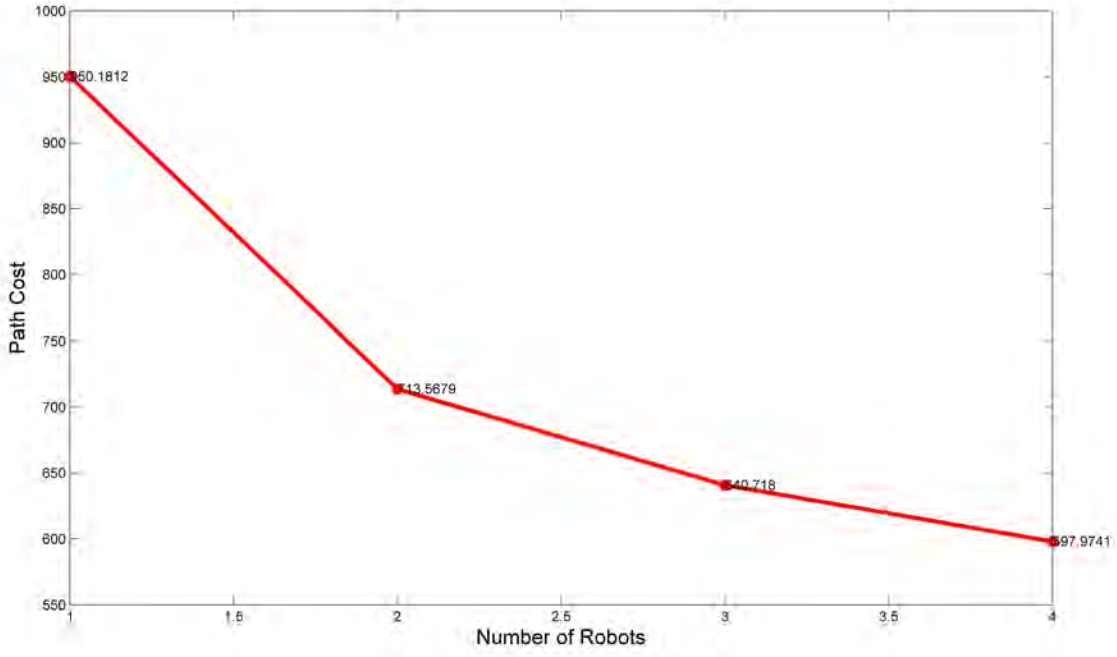


Figure 9: Path Costs with Two Robots

image processing portion of the problem has not been investigated in this study and will require further work. For the path planning work done here, the Greedy and Brute-force algorithms as well as the Linear Programming method all have deficiencies. Brute-Force and LP are computationally expensive but provide complete solutions for small order problems. The Greedy Algorithm is fast, but is known to return sub-optimal solutions. Other methods not investigated here are modifications to the Travelling Salesman Problem to include the PUP and DOP constraints, or application of Genetic Algorithm variants.

The corner detection method assumes that each observed block is contained in the interior of the arena. When an edge of a block coincides with the arena boundary, corner points must be adjoined to the list of boundary corners, and the orientation must be corrected. The visibility graph function requires that interior points be ordered in clockwise fashion while the exterior boundary must be counter-clockwise, following the well known right-hand rule to maintain correct orientation.

Many algorithms have been developed to calculate distances between nodes. The visibility graph method was chosen for this project due to ease of implementation and the fact that it provided an L2 norm on the space. Other methods considered used the L1 norm (taxicab geometry) which resulted in ambiguous path choices in an open arena. The L1 norm is more suitable for a maze or inner city driving.

5 Conclusion

After modification of standard maze generation techniques, we were able to generate arenas that closely resemble the arena expected for the autonomous robot challenge. These simulated environments provided the means to test corner detection, visibility graph algorithms and path planning techniques. We showed that for small numbers of intermediary nodes, the Brute-Force and Linear Programming methods will give globally optimal solutions, and we developed a novel multi-agent Greedy Algorithm that adapts well to dynamically changing environments.

References

[1] http://www.cs.cmu.edu/ jkubica/code/matlabmaze.html.

[2] Tolga Bektas. The multiple traveling salesman problem-an overview of formulations and solution procedures. 34(3), June 2006.

[3] R. Clark, Wedeward K. El-Osery, A., and S. Bruder. International test and evaluation association workshop on modelling and simulation, las cruces, nm,. In *A Navigation and Obstacle Avoidance Algorithm for Mobile Robots Operating in Unknown, Maze Type Environments*, December 2004.

[4] Leiserson Cormen and Rivest. *Introduction to Algorithms*. MIT Press.

[5] Thomas H. Cormen, Charles E. Leiserson, Ronald L. Rivest, and Clifford Stein. *Introduction to Algorithms*. The MIT Press, 2 edition, 2001.

[6] Mark de Berg, Marc van Kreveld, Mark Overmars, and Otfried Schwarzkopf. *Computational geometry: algorithms and applications*. Springer-Verlag New York, Inc., Secaucus, NJ, USA, 1997.

[7] T. Ersson and Xiaoming Hu. Path planning and navigation of mobile robots in unknown environments. In *Intelligent Robots and Systems, 2001. Proceedings. 2001 IEEE/RSJ International Conference on*, volume 2, pages 858 –864 vol.2, 2001.

[8] D.S. Katz and R.R. Some. Nasa advances robotic space exploration. *Computer*, 36(1):52 – 61, jan 2003.

[9] B.H. Kazerouni, M.B. Moradi, and P.H. Kazerouni. Variable priorities in maze-solving algorithms for robot's movement. In *Industrial Informatics, 2003. INDIN 2003. Proceedings. IEEE International Conference on*, pages 181 – 186, aug. 2003.

[10] Donald E. Knuth. *The art of computer programming, volume 2 (3rd ed.): seminumerical algorithms*. Addison-Wesley Longman Publishing Co., Inc., Boston, MA, USA, 1997.

[11] Joseph. B. Kruskal. On the shortest spanning subtree of a graph and the traveling salesman problem, Febrary 1956.

[12] Jean-Claude Latombe. *Robot Motion Planning*. Kluwer Academic Publishers, Norwell, MA, USA, 1991.

[13] S. Mishra and P. Bande. Maze solving algorithms for micro mouse. In *Signal Image Technology and Internet Based Systems, 2008. SITIS '08. IEEE International Conference on*, pages 86 –93, 30 2008-dec. 3 2008.

[14] R.R. Murphy. Trial by fire [rescue robots]. *Robotics Automation Magazine, IEEE*, 11(3):50 – 61, sept. 2004.

[15] R. Nallusamy. optimization of multiple vehicle routing problems using approximation algorithms. 1(3), 2009.

[16] K. J. Obermeyer and Contributors. The VisiLibity library. `http://www.VisiLibity.org`, 2008. R-1.

[17] R. C. Prime. Shortest connection networks and some generalizations. 1957.

[18] D. Voth. A new generation of military robots. *Intelligent Systems, IEEE*, 19(4):2 – 3, jul-aug 2004.

[19] L. Wang. Computational intelligence in autonomous mobile robotics-a review. In *Micromechatronics and Human Science, 2002. MHS 2002. Proceedings of 2002 International Symposium on*, pages 227 – 235, 2002.

[20] Emo Welzl. Constructing the visibility graph for n-line segments in o(n2) time. *Information Processing Letters*, 20(4):167 – 171, 1985.

Saltwater Intrusion and Freshwater Supply in Coastal Aquifers

Quanling Deng[1], Will Henry[2], Mona Karimi[3], Mikhail Smilovic[4], Jillian Trask[5], Liang Wu[6],

Problem Presenter:
Matthew Farthing
US Army Corps of Engineers

Faculty Mentor:
Lea Jenkins
Clemson University

Abstract

An optimization model with the economic objective of maximizing pumped groundwater and minimizing saltwater intrusion is developed. The proposed optimization model–S-Well–is based on the semi-anlaytical solution by Strack [12] with the assumptions of the sharp interface and vertically integrated flow. The saltwater intrusion problem is reduced to solving potential flow with a search for two-dimenional space for the saltwater toe location. A simple genetic algorithm (GA) is used for the optimzation. Aquifer parameters such as hydraulic conductivity and aquifer thickness, freshwater outflow, and recharge rate are either uncertain or random. To calibrate the S-Well Optimization model under these uncertain conditions, the Monte Carlo sensitivity analysis is presenented. The proposed optimization model–S-Well–is then applied to a hypothetical unconfined coastal aquifer in Pajaro Valley, California as a test problem. Finally, we discuss the optimal location of wells given a virgin land and solve the Pareto curve for the cost versus output dual-objective optimization problem.

1 Introduction

Groundwater represents about 30 percent of all fresh water on the earth, polar ice represents 69.7 percent, while other freshwater in rivers and lakes represent about 0.3 percent [14]. Groundwater is a vital resource for sustaining the communities and economies of the coastal regions in the world. One-third of the world's drinking water is provided by groundwater. In the United States, it is the source of drinking water for about half the total population and nearly all of the rural population, and it provides over 50 billion gallons per day for agricultural needs [15].

As major sources for freshwater supply in many countries around the world, groundwater aquifers play an important role in water management. In many coastal areas, high urbanization rates and agricultural growth have increased the groundwater demand. While the available water resources are nearly constant, to secure such demand is a great challenge. Wells have been drilled excessively to satisfy this demand.

The proximity of coastal aquifers to saltwater creates unique issues with respect to groundwater sustainability in coastal regions. Overdraft from coastal aquifers and thus saltwater intrusion (SWI) toward aquifers cause 'unacceptable' draw-down and deterioration of water quality. High salinity of groundwater limits its usage for irrigation and drinking purposes unless desalinated or mixed with lower salinity water. Hence, the protection of groundwater resources becomes an essential matter under the conditions of increasing demands and decreasing available resources. To ensure a longterm sustainable water supply, withdrawal and management policies are necessary. The objective here is to maximize the net benefit of water use, subject to the constraints of no-saltwater-intrusion into the wells, installment constraints, and restrictions on the placement of pumping wells.

Initial efforts to improve the fresh water supply in coastal aquifers via optimization techniques started in the early 1970s. Previous reviews of SWI models can be found in Cummings [5], Shamir [11], El Harrouni

[1]University of Wyoming
[2]San Diego State University
[3]Texas A&M University
[4]McGill University
[5]University of Tennessee, Knoxville
[6]University of Notre Dame

et al [6], and Cheng et al [4]. Most of the previous approaches have been involved with various management objectives. The uncertainty in modeling issues is presented by some researchers such as Benhachmi et al [2]. In the present work, the optimization problem formulated is solved by using the genetic algorithm technique by Aral and Guan [1]. The method developed is applied to the previous work by Cheng et al [3] and a specification is provided to demonstrate the use of the model.

The objectives of this paper are twofold: first to provide an optimization model for SWI and, second to present a preliminary predictive uncertainty analysis in modeling issues. For the optimization model, we develope a multi-objective approach to determine pumping rates and well placement while satisfying desired extraction rates in coastal aquifers and avoiding SWI. The objective is formulated based on the analytical solution for the steady state SWI model (see equation1).

$$\phi = \frac{q}{K}x + \frac{Q}{4\pi K}\ln\left[\frac{(x-x_w)^2+y^2}{(x+x_w)^2+y^2}\right] \tag{1}$$

2 Model and Methods

2.1 Optimization Model

As with many saltwater intrusion problems, our first objective is to maximize the sum of the pumping rates of the wells. In turn, these wells have pumping capabilites which must be taken into account. Unlike many other models, another objective, in this model, is to place the wells as close as possible to the coastline in order to avoid the increased cost of well installation further from the coast. This, of course, restricts the first objective. In addition, we assign a penalty to wells which encounter encroachment so as to limit saltwater intrusion. Per Park and Cheng [9, 3], we combine these objectives and constraints into one objective function by rewriting and normalizing. The optimization model is as follows:

$$\begin{aligned} &\max \sum_{i=1}^{n} \alpha\left(\frac{Q_i}{{Q_i}^{max}}\right) + \beta\left(\frac{x_{ref}}{{x_s}^i} - 1\right) - \sum_{i=1}^{n} \gamma\left({x_{diff}}^i\right)^2 \\ &\text{where} \quad {x_{toe}}^i + {x_{diff}}^i = {x_s}^i \\ &\text{and} \qquad \gamma = \begin{cases} c_1 & {x_{diff}}^i < 0 \\ 0 & \text{elsewhere} \end{cases} \end{aligned} \tag{2}$$

Here α and β are weighting constants. See Table 1 for an explanation of the symbols in equation 2.

Symbol	Explanation
c_1	any large constant
Q_i	pump rate for well i
x_{ref}	reference coastline location
x_i	x coordinate of well i
${x_s}^i$	stagnation point for well i
${x_{diff}}^i$	slack variable for well i
${x_{toe}}^i$	toe location for well i

Table 1: Explanation of symbols for the optimization model, equation 2.

Note that this model assumes a sharp-interface, meaning that the region separating the saltwater and freshwater is thin in comparison to the depth of the groundwater aquifer. Additionally, we assume steady state conditions and that the aquifer is homogeneous. Furthermore, the model adopts the Dupuit assumption, allowing us to consider a two-dimensional domain as opposed to a three-dimensional domain. Moreover, the Badon-Ghijben-Herzberg (BGH) principle, which assumes that stagnant water is used to interpret the interface location, is supposed. Finally, the single-potential theory approach is assumed, giving continuity over the freshwater and saltwater zones [9].

After determining the basic model, we applied a genetic algorithm in order to calibrate the model. Specifically, we ran simulations with regard to Pajaro Valley. Pajaro Valley is located in Santa Cruz County of California; at approximately 120 mi^2, it is the largest farming area in the county. As such, its economy is heavily dependent on its water supply for agricultural and domestic uses, among others. Dry farming was prevalent in the valley in the mid-to-late 1800s, but the valley has gradually transitioned to crops which require more water. Its current water demand is 55,000 acre-feet per year, two-thirds of which is met by groundwater. Among the valley's main sources of recharge to groundwater aquifers are rainfall, irrigation return, and infiltration of surface water [8].

2.2 S-Well Algorithm

The S-Well algorithm is our "Simple Well Genetic Algorithm". This algorithm was created by us in MATLAB. First, the genetic algorithm (GA) is set up for specific parameter values: q, K, upper and lower bounds on Q, and the x and y domain. If we are fixing the location of the wells (i.e., not allowing the algorithm to move them) then the x and y coordinates of each well is also provided. Other parameters, such as the population number, and number of generations, survivors, breeds, and mutants, are also defined. The S-Well algorithm then maximizes the sum of all pumping rates while avoiding encroachment. Encroachment is defined as the point when the toe line has coordinate values that are within $\pm$ 1 grid point of a well's y location and greater than (beyond) a well's x location. The objective functional used in the S-Well algorithm is:

$$\max \sum_{i=1}^{n} \alpha Q_i - Q_{max} \cdot (\text{number of toe points that encroach}) \tag{3}$$

The first generation is generated by slightly mutating a seed solution. For our work, a seed solution provides the locations of four wells and their respective pumping rates. Because we have fixed locations in the S-Well algorithm, the mutation is only applied to the pumping rates. In the process of generating the seed mutations, each pumping rate has a fifty percent chance of being mutated and the mutation factors range randomly from 0.75 to 1.25. In our model we generate 36 mutations for our first generation. The objective function is calculated for each of the 36 solutions; the solutions are ranked by the objective, from greatest to least and the top 13 are kept as survivors for the next generation. We then create 18 breeds via matings between the 36 solutions of the first generation. The mating pairs are chosen randomly for each of the 18 breeds, however, the probability of being selected to breed is proportional to the solution's performance as determined by the objective functional. In order to find the probability of mating, we first raise each objective score of the 36 solutions to the power of 6, thereby augmenting the advantage of the better perfoming solutions. Next, the new scores are normalized against the sum of the new scores to give the probability of each solution breeding. Finally, we create five additional solutions by mutating only the best solution of the first generation. This mutation process is the exact same as that which was used to mutate the seed solution. We now have 36 new solutions (13 survivors, 18 breeds, and 5 mutants) that make up our second generation. The objective function is then evaluated for each of these solutions and the process of survival, breeding, and mutating is repeated. This comprehensive process is repeated until the termination criteria is achieved; in this case, the termination criteria is simply completing 500 generations. See Figure 3 for a flowchart of the GA algorithm.

2.3 Well Thought-Out Well Placement

2.3.1 Get Well (but get them away from the coast)

We have thus far dealt with optimizing output and minimizing salt-water intrusion given a fixed number of wells and fixed locations. From this, we observe that, the optimal solution can result in sacrificing a given number of wells to achieve maximum output. This leads to the question, given virgin land, where are the ideal well locations? Adding a few more constraints and running the problem through the optimizer more than provides a maximal output, but at the expense of highlighting the flaws embedded into our idealized model; the pumps are located at the edges of the domain. It must, then, be addressed that our coast is infinitely long. The first advantage to increasing maximal output is, then, to place the wells as far apart from one another as the domain allows. The other approach is to place the wells as far away from the coast as possible; wells at the edge can pump more before the toe-line encroaches.

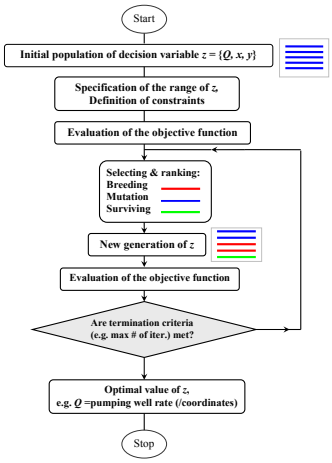


Figure 1: Genetic Algorithm Flowchart

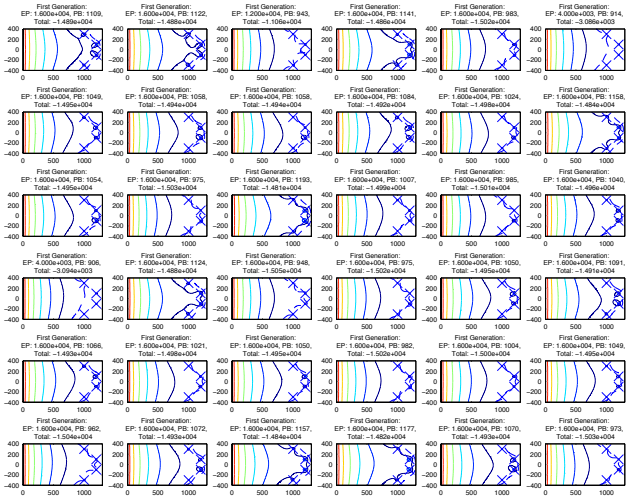
Parameter Symbol	Parameter Explanation	Parameter Value
K	hydraulic conductivity	100
q	freshwater outflow rate	0.6
d	depth of aquifer	15
Q_{min}	minimum pump rate	0
Q_{max}	maximum pump rate	1500
x_{min}	minimum x-value of well location	0
x_{max}	maximum x-value of well location	1300
y_{min}	minimum y-value of well location	-400
y_{min}	maximum y-value of well location	400
population number		36
number of generations		500
number of breeds		13
number of mutants		18
		5

Table 2: Explanation of symbols for the optimization model and potential equation, equations 1 and 2 respectively.

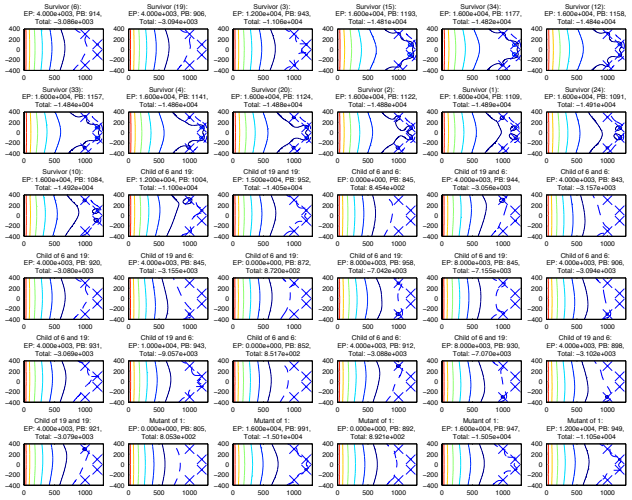
It appears the ideal well locations are simply stated: first, place them as far away from the coast as possible and, second, as far apart from one another as possible. However, this seems far from the ideal (realistic) solution and we need to take some other factors into account. First, wells may be necessary in certain regions of the land mass to avoid transportation of water from other wells. Second, pumps are not infinitely powerful and are subdued to maximum pumping rates. Lastly, we must acknowledge the financials associated with the base cost of each well, the pumping rates of wells, and those associated with building further away from the coast.

2.3.2 The Price, the Pumping, and the Pareto

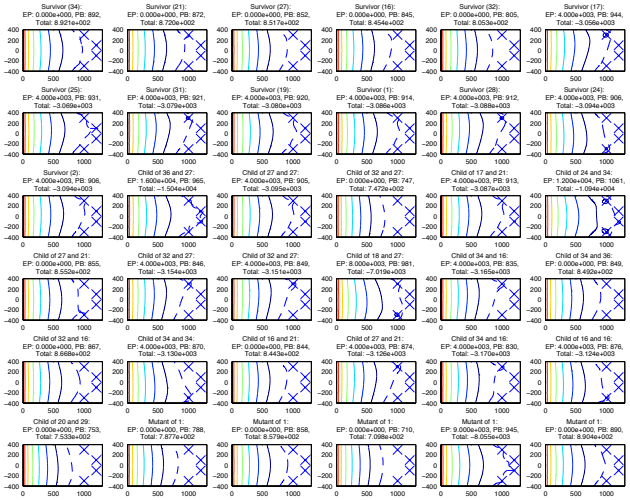
We are presented with a dual-objective optimization problem; optimize the output and minimize cost. Before we can begin to solve this, we need to develop a rough estimate of cost. We will take three things into consideration: a base cost, pumping rate, and costs associated with the distance a well is placed from the coast. The latter perhaps deserves some justification. We assume that wells farther from the coast will have greater installation costs associated with the need for deeper holes. We can also assume that the majority of one's water needs are evenly distributed throughout the landmass, and placing all the wells along the boundary



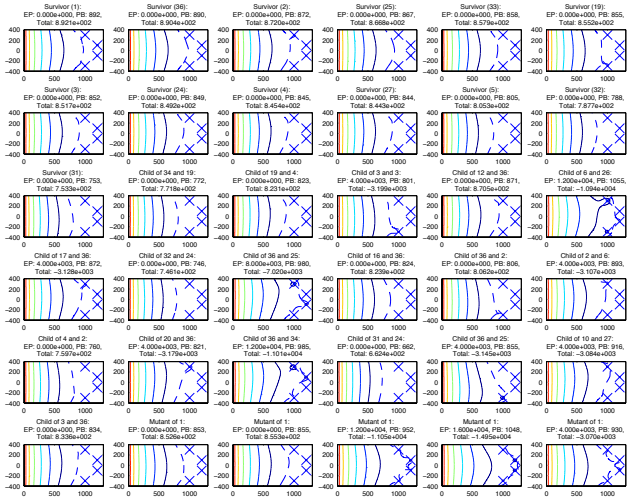
(a) First Generation Solutions



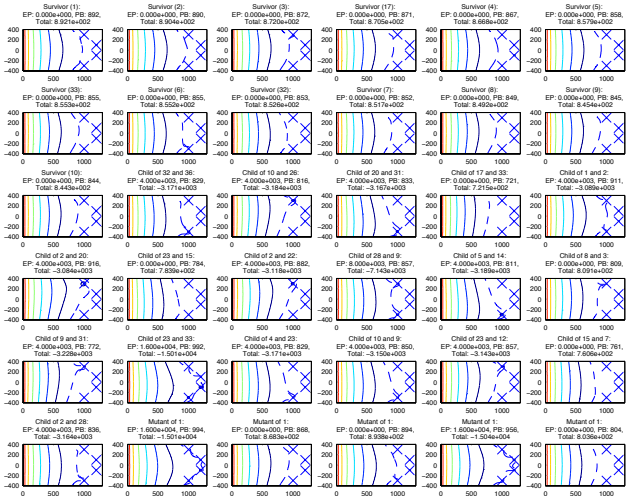
(b) Second Generation Solutions



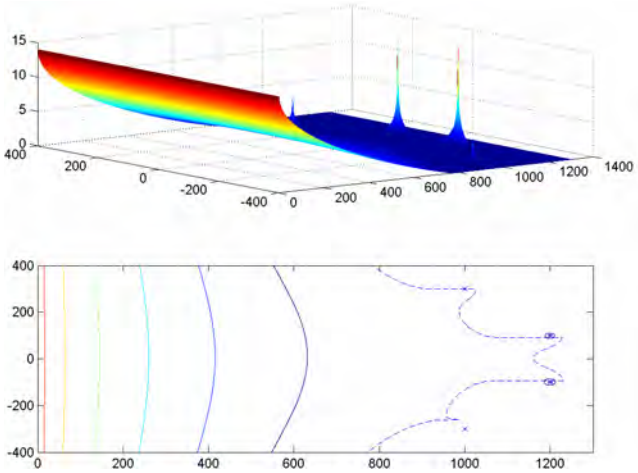
(c) Third Generation Solutions



(d) Fourth Generation Solutions



(e) Fifth Generation Solutions



(f) Best of 500^{th} Generation Solutions

Figure 2: Output of S-Well Algorithm. Figure (f) shows the optimal solution after 500 generations with well 1 at $(1000, 300)$, $Q = 168.14$; well 2 at $(1200, 100)$, $Q = 344.63$; well 3 at $(1200, -100)$, $Q = 371.63$; well 4 at $(1000, -300)$, $Q = 145.84$

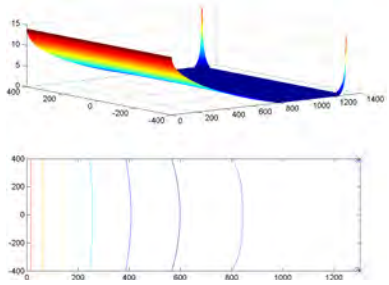


Figure 3: Optimal location of wells.

	Cost (£)
Base cost	$20,000$
Pumping rate	$Q \times 10$
Distance from coast	$x \times 30$

Table 3: Cost of well located at (x, y) with pumping rate Q.

will have subsequent costs associated with water transportation. Naturally all costs are in Well-sh currency, the pound.

The optimization problem and resulting Pareto curve were solved using two different methods. We begin with the first method, the Sweeping Method [7]. Given objectives f_1 and f_2, we create a single-objective optimization problem with f_1 as the objective function and f_2 as a constraint. We then run this optimizer n times, with n values for f_2 evenly distributed between it's minimum and maximum, a sweep. Depending on the nature of the Pareto curve, it may be desirable to repeat this process exchanging the roles of f_1 and f_2. This problem, however, yields a relatively tame Pareto curve and one sweep is satisfactory. For the single-objective optimization, we used the NSGA2-algorithm. The second method was to again use the NSGA2-algorithm, this time with it's multi-objective capabilities resulting in the same Perato curve.

2.4 Sensitivity Analysis

2.4.1 Introduction

The usefulness of our model here depends in part on the accuracy and reliability of the outputs. Yet, because our models are always imperfect simplifications of real-world systems, and because input data cannot be perfectly accurate, and may vary from time to time, the model outputs are subject to imprecision. Sensitivity analysis (SA) is the study of how the uncertainty in the output of a model can be apportioned to different sources of uncertainty in the model input. In this model, there are some uncertainty in several parameters, such as the hydraulic conductivity K and the uniform freshwater outflow rate q. In Cheng's paper, these values are assumed to be constants. Since there is always some uncertainty in the data, it is useful to know over what range and under what conditions the components of a particular solution remain unchanged. Further, the sensitivity of a solution to changes in the data gives us insight into possible technological improvements in the process being modeled. Sensitivity analysis provides an invaluable tool for addressing such issues. Its procedures explore and quantify the impact of possible errors in input data on predicted model outputs. Simple sensitivity analysis procedures can be used to illustrate either graphically or numerically the consequences of alternative assumptions about the future.

There are a number of questions that could be asked concerning the sensitivity of an solution. In our models here, for example, the geological parameter K, hydraulic conductivity, varies for aquifers in different areas, and even varies in different layers in an aquifer. Also model parameter q, uniform freshwater outflow rate, is highly related to rainfall, and historical records of rainfall data are used as a basis for our model inputs.

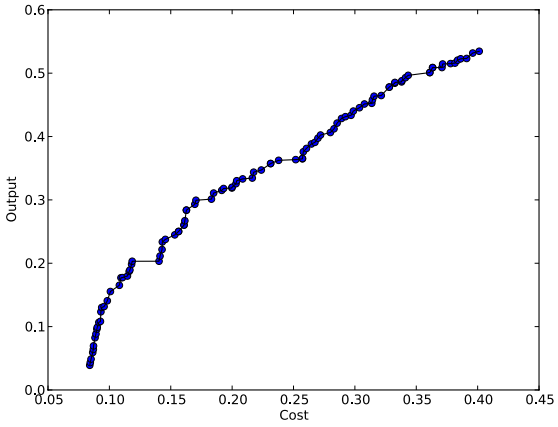


Figure 4: Perato Curve

Yet such precipitation conditions in the future may change. That gives the necessity of performing sensitivity analysis.

Regarding the methodologies, there are several possible procedures to perform uncertainty and sensitivity analysis. Commonly used methodologies include automated differentiation, methods based on emulators, screening methods, variance based methods, methods based on Monte Carlo filtering; the list goes on. Here we use Monte Carlo method to perform sensitivity analysis with respect to our model parameters K and q. This Monte Carlo method first selects a random set of input data values drawn from their individual probability distributions. These values are then used in the simulation model to obtain some model outputs. This process is repeated many times, each time making sure the model calibration is valid for the input data values chosen. The ending results will show how model outputs behave when inputs vary.

2.4.2 Analysis for hydraulic conductivity K

The Hydraulic conductivity, K, is one of the most important parameters for optimizing pumping rate and well locations with constraints, i.e. that no saltwater intrusion. Different values of hydraulic conductivity were applied to investigate the sensitivity analysis of this parameter in the optimization of pumping rate and well locations.

In Cheng's paper, the hydraulic conductivity K is a constant, which was assumed to be 100 m/d (meter per day). To do the sensitivity analysis for K, it is necessary to estimate some representation of the variances of the parameter K with some consistent procedure. For a normal distribution, the distance between the 5^{th} and 95^{th} percentiles is 1.645 standard deviations on each side of the mean, that is approximately 3.3 standard deviations. Here we use real data with a mean value $m = 15.4$ meter per day, and a standard deviation $std = 1.25$ meter per day for K. Thus, if the high or low range is thought of as approximately a $5 - 95$ percentile range for a normally distributed variate, a reasonable range for K might be $[m - 3 \times std, m + 3 \times std]$. Here we consider lognormal distributed values for K.

By Matlab code, we then get the toe-tracking graph (in Figure 5) for a single well. In Figure 1, the star stands for the well location and the black, red, and blue lines refer to the the tracking toes. If the x axis value of a point on the line is greater than the x axis value of well location (which is 1000 here), then it means that it has intrusion of the saltwater front into the well. Figure 1 shows us that the most of the values for K we used in our model is reasonable. We use the data about 1000 lognormal distributed values for K, and find that for 18.2% K values, we get intrusion and 81.8% of all these K values subject to the constraint of no intrusion of the saltwater front into the well.

The abnormal red tip on the red curve means that the toe movement is sensitive to the hydraulic conductivity K near the well. Near the well, the toe line changes relatively significant when the input value K changes a little. Therefore, to optimize the pumping rate for a single well, we need a more accurate value for hydraulic conductivity K near the well.

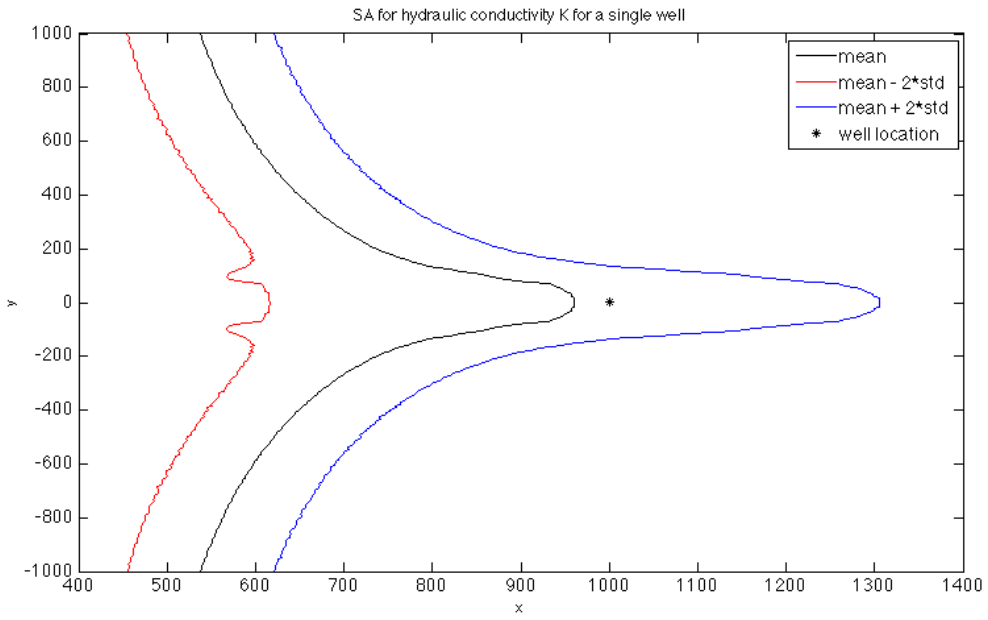


Figure 5: SA_K: Saltwater intrusion front for a single pumping well

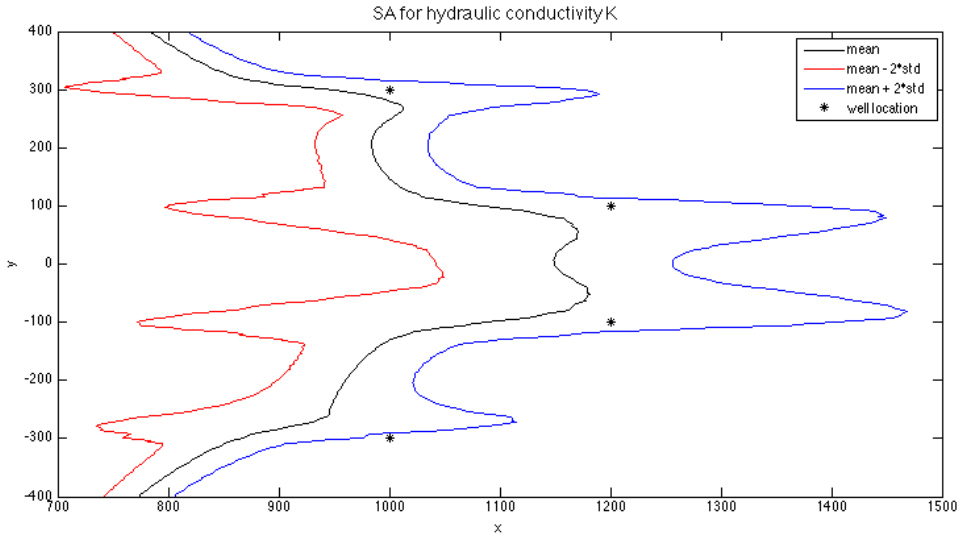


Figure 6: SA_K: Saltwater intrusion front for 4 pumping rate optimized wells

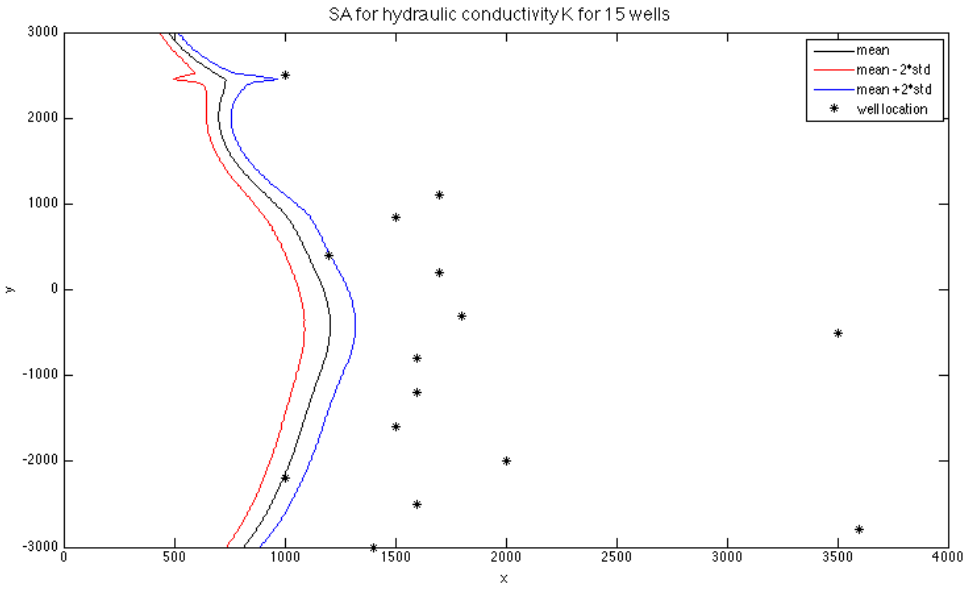


Figure 7: SA_K: Saltwater intrusion front for 15 pumping rate optimized wells

We also considered multiple wells with optimized pumping rates, and found that the toe movements are locally sensitive to the hydraulic conductivity K near all the wells. In Figure 2 and Figure 3, 4 and 15 wells with optimized pumping rates are considered respectively. Near the wells, the toe movements change significantly when the input K values changes a little. Similar to the single well case, to optimize the pumping rate for multiple wells, we need more accurate values for hydraulic conductivity K near all these wells.

2.4.3 Analysis for uniform freshwater outflow rate q

Besides the hydraulic conductivity K, there are some uncertainty in the uniform freshwater outflow rate q. Different values of uniform freshwater outflow rates were applied to investigate the sensitivity analysis of this parameter in the optimization of pumping rate and well locations. The uniform freshwater outflow rate q has a great influence on the optimization problem. Generally speaking, the higher the uniform freshwater outflow rate q is, the higher the optimized pumping rate will be. The higher the uniform freshwater outflow rate q is, the harder wells can pump subject to the constraint of no intrusion of the saltwater front into the wells. On the other hand, uniform freshwater outflow rate q is influenced by rainfall data. In general, the more rain falls, the higher the uniform freshwater outflow rate q will be.

Similar to what we did for the hydraulic conductivity K, we considered lognormal distributed values for q. We considered multiple pumping rate optimized wells, and found that the toe movements are locally sensitive to the uniform freshwater outflow rate q near all the wells. The curves in Figure 4 shows the movements of toes when we vary the values for q. Figures 4 and 5 represent the influence of uniform freshwater outflow rate on the corresponding 4 and 15 pumping rate optimized wells. The movements of toes vary a lot when we the values for q vary a little. That's to say that the toe movements are sensitive to the parameter q. In summary, to optimize the pumping rate for multiple wells, we need more accurate values for uniform freshwater outflow rate q near all these wells.

2.4.4 Summary

Determining the appropriate values to assign to the numerical items (the input variables) in a model is a critical and challenging part of the model building process in decision analysis. But finding numerical values for real problems requires gathering relevant data, which can sometimes be difficult. As a result, we often use rough estimates. Because of the uncertainty about the true value of a numerical item, it is important to

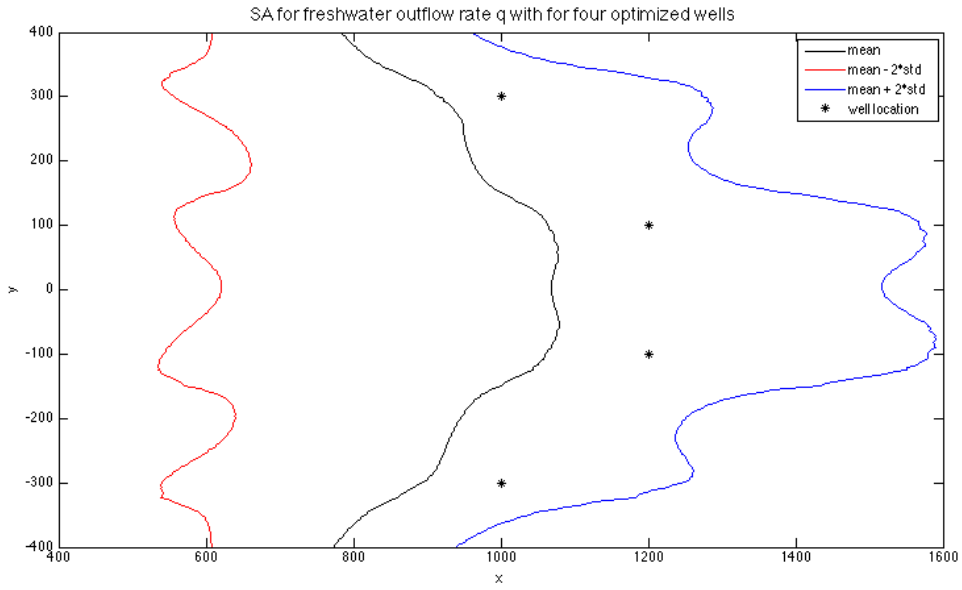


Figure 8: SA_q: Saltwater intrusion front for 4 pumping rate optimized wells

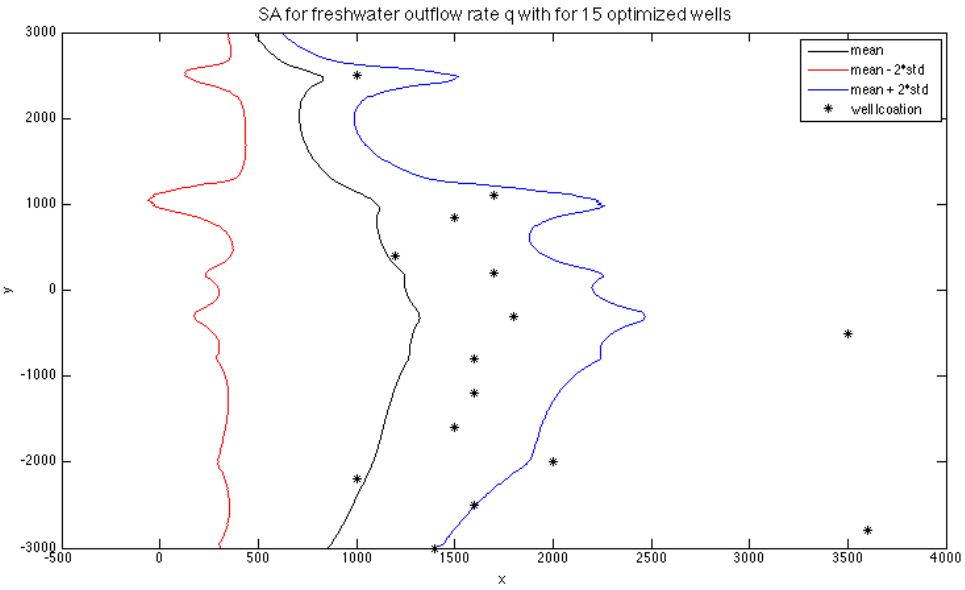


Figure 9: SA_q: Saltwater intrusion front for 15 pumping rate optimized wells

determine how the solution derived from the model would change (if at all) if the numerical value assigned were changed to other plausible values.

3 Discussion

There are several approaches to multi-objective problems, such as fitness sharing, the niche approach [9], and even gradient methods [3]. However, because we do not know our solution space a priori, we have chosen the GA method as it does not require that it be continuous. Unfortunately, the GA can be less efficient, requiring more time to run its evaluations.

Our model and simulations are preliminary; they are an idealization of reality. Obtaining data, particularly recent data, on Pajaro Valley remains challenging; Pajaro Valley has been sorely understudied. Should more recent and accurate data pertaining to Pajaro Valley become available, it would be worthwhile to revisit the simulations to revise our parameter estimates. While parameters such as Q, d, and the domain in our model are scaled versions of the true (albeit outdated) figures for Pajaro Valley, others such as K and q are adopted from Cheng [3] in order to have successful runs of the S-Well algorithm. Note, however, that K and q are able to be calculated and scaled given data on Pajaro Valley, but the scaled calculations fail to allow the algorithm to run. In addition, our model neglects time, flow, and mixing, all of which have the potential to improve our model by making it more realistic. Finally, once we are able to refine the model, we are interested in applying numerics to obtain numerical solutions.

This group has a particular interest in incorporating the monetary costs of manufacturing, placing, and maintaining wells into the model as this could have a considerable impact on the direction of the decision makers when judging the course of action to address declining resources.

References

[1] Aral, M. M., and J. Guan. Optimal groundwater remediation system design with well locations selected as decision variables. Multimedia Environmental Simulations Laboratory No. MESL-01-97:Georgia Tech, Atlanta, Georgia, 1997.

[2] Benhachmi, M.K., Ouazar, D., Naji, A., Cheng, A.H.-D. and EL Harrouni, K. Chance constrained pumping optimization in saltwater intruded aquifers by simple genetic algorithmStochastic model. Proc. 2nd Saltwater Intrusion and Coastal Aquifers. Edited by L. Marin, Merida, Mexico, 2003.

[3] Cheng, A.H.-D., D. Halhal, A. Naji, and D. Ouazar. Pumping optimization in saltwater-intruded coastal aquifers. Water Resources Research, 36 (8): 2155 – 2165, 2000.

[4] Cheng, A.H.-D., and D. Ouazar. Analytical solutions in Saltwater Intrusion in Coastal Aquifers - Concepts, Methods, and Practices. Edited by J. Bear et al. Kluwer Acad.:Norwell, Mass, chap. 6, pp. 163-191, 1999.

[5] Cummings, R.G., Optimum exploitation of groundwater reserves with saltwater intrusion. Water Resources Research 7: 1415-1424, 1971.

[6] El Harrouni, K., Ouazar, D. and Cheng, A.H.-D. Salt/fresh-water interface model and GAs for parameter estimation. Proc. SWIM 15, Ghent, 1998.

[7] Hosseini, Liu, Puelz, Smilovic, Tracht. Visualizing the Perato Surface. IMA pre-prints, 2012.

[8] Johnson, M. J., Londquist, C. J., Laudon, J., and Mitten, H. T. Geohydrology and Mathematical Simulation of the Pajaro Valley Aquifer System, Santa Cruz and Monterey Counties, California. U.S. Geological Survey. Water Resources Investigations Report 87-4281, 1988.

[9] Park, C.-H. A Thesis Presented to the Academic Faculty: Saltwater Intrusion in Coastal Aquifers. Georgia Institute of Technology, 2004.

[10] Parsinejad, M., Y. Feng and B. Ghanbarian, Sensitivity Analysis of Hydraulic Parameters in the Simulation of Unsaturated Soil Water Dynamics, Iran Agricultural Research, Vol. 24, No.2 and Vol. 25, No.1, 2006.

[11] Shamir, U., J. Bear, and A. Gamliel. Optimal Annual Operation of a Coastal Aquifer. Water Resources Research 20: 435-444, 1984.

[12] Strack, O. D. L. Single-Potential Solution for Regional Interface Problems in Coastal Aquifers. Water Resources Research 12:1165-1174, 1976.

[13] Szilagy, J. Sensitivity analysis of aquifer parameter estimations based on the Laplace equation with linearized boundary conditions. WATER RESOURCES RESEARCH, 39 (6): 1156, 2003.

[14] UN Atlast of the Oceans, Human Settlements On the Coast. http://www.oceansatlas.org/servlet/CDSServlet?status=ND0xODc3JjY9ZW4mMzM9KiYzNz1rb3M , Accessed July 2012.

[15] US Geological Survey, Groundwater depletion. http://ga.water.usgs.gov/edu/gwdepletion.html, Accessed July 2012.

[16] WATER RESOURCES SYSTEMS PLANNING AND MANAGEMENT ISBN 92-3-103998-9 UNESCO 2005, Chapter 9.

A S-Well Algorithm Code

```
%% multiPumpingWellFixedLocGa


%% %%%%%% Initilization %%%%%%
clearvars



%% %%%%%% Options & Variables %%%%%%

% Physical Options
K = 40         ;   % hydrolic conductivity m/d
q = 0.4015         ;   % uniform freshwater outflow rate m^2/d
d = 14          ;   % elevation of mean sea level above the datum m
rho_f = 1       ;   % freshwater density g/cm^3
rho_s = 1.025   ;   % saltwater density g/cm^3
s = rho_s/rho_f ;   % saltwater to freshwater density ratio

% Domain Definition
xMin = 0 ;
xMax = 4000 ;
nx = 1000;

yMin = -4000 ;
yMax = 4000 ;
ny = 2000 ;


x = linspace(xMin,xMax,nx);
y = linspace(yMin,yMax,ny);
[X,Y] = meshgrid(x,y) ;

% Genetic Algorithm Options
genNumber = 200;
popNumber = 25;
surviveNumber = 10;
breedNumber = 14;
mutationNumber = 1;
minMutationFactor = 0.5 ;
maxMutationFactor = 1.5 ;
probAugmentFactor = 5 ;

% Optimization Constraints
optimizeToeLine = false;
lbQ = 0 ; % Q lower bound
ubQ = 1500 ; % Q upper bound
```

B Pareto Code

```
from pyOpt import NSGA2
from pyOpt import Optimization
import numpy as np
from objfunctionex import objfunc
import os
from math import pi, sqrt, log
import re
from matplotlib import pyplot, pylab

#Constants

Const={'K':100.,
       'q':0.6,
       'd':14,
       'num_wells_M':4,
       'rho_s':1.025,
       'rho_f':1.}

Const['s']=Const['rho_s']/Const['rho_f']

#Domain

D={'x_m':0., #minimum x for bounds
   'x_M':4100., #maximum x for bounds
   'y_m':-1600., #minimum y for bounds 1600
   'y_M':1600., #maximum y for bounds 1600
   'x_m_well':1000.} #minimum x for well location

D['x_M_well']= D['x_M']-100. #maximum x for well location 100
D['y_m_well']= D['y_m']+100. #minimum y for well location 100
D['y_M_well']= D['y_M']-100. #maximum y for well location 100

X=np.arange(D['x_m'], D['x_M'], 20.)

def optimizer_setup():

    #Setup the optimization problem

    optprob=Optimization('Optimal well arrangement given cost', objfunc)

    #Objective functions
    optprob.addObj('f1')
    optprob.addObj('f2')
```

```
import numpy as np
from math import cos, atan, sqrt, pi, log, floor

def multiWellPotential(q,K,Q_list,X,Y,x_w_list,y_w_list):
    """
    Multiple well potential from Cheng
    Assumes q,K are scalars
    """
    #should/could check that arrays have the same shape
    phi = np.zeros(X.shape,'d'); tmp = np.zeros(X.shape,'d');
    phi_i= np.zeros(X.shape,'d')
    for Q,x_w,y_w in zip(Q_list,x_w_list,y_w_list):
        dist = np.power(X-x_w,2); dist += np.power(Y-y_w,2)
        tmp = np.power(X + x_w,2); tmp += np.power(Y-y_w,2)
        tmp = np.divide(dist,tmp);
        tmp = np.where(tmp > 0,tmp,1.0e-8)
        tmp = np.log(tmp);
        tmp *= Q/(4.0*np.pi*K)
        phi += tmp
    #
    tmp[:] = X; tmp *= (q/K)
    phi += tmp

    phi = np.where(phi >= 0,phi,0.0)
    return phi

def depth_unconfined(phi,s):
    """
    compute the interface depth for an unconfined aquifer
    """
    xi = np.copy(phi)
    xi *= 2.0/(s*(s-1))
    xi = np.sqrt(xi)
    return xi

def compute_interface(xi,d):
    """
    compute the interface depth for an unconfined aquifer
    """
    interface = np.copy(xi)
    interface -= d
    interface *= -1.0
    interface = np.where(interface >= 0, interface,0.0)
    return interface

def objfunc(x, Const={}, D={}, X=[]):
```

Inspection of Composite Assemblies Using a Non-destructive Approach

Marylesa Howard[1], Fernando Camacho[2], Zackary Kenz[3], Ruiqiang Song[4], Wei Yijun[5], Xiaoqing Zhu[6]

Problem Presenter: Haris Halilovic, LORD Corporation
Faculty Mentors: Drs. H.T. Banks[3] and Shuhua Hu[3]

Abstract

Acoustic emissions (AE) occur naturally from materials subjected to stress. For example, a material may emit acoustic emissions caused by of redistribution of internal stresses in the material upon being compressed. The presence of AE can be used to indicate the creation and growth of micro cracks among other defects [9].

AE is classified as a non-destructive test. No external stimulus is applied; one only measures AE signatures the material naturally emits. One of the attractive features of AE analysis is that it is a portable test that can give real time information about the status of working systems like airframes, pressure vessels and pipelines [17].

In this report we analyze the AE experimental results of a destructive test applied to some composite material samples. Each composite sample is either from a control group, a group with an unspecified defect 1, or an unspecified defect 2. Parameter subset selection analysis is completed, then classification methods are applied to delineate between a control piece and a defective piece. Finally, we examine methods to predict the plastic deformation and breaking loads from the acoustic emission signatures.

[1]University of Montana
[2]University of Kentucky
[3]North Carolina State University
[4]Michigan Technological University
[5]University of Michigan
[6]Michigan State University

Contents

1 Introduction and Motivation

LORD Corporation is a leading global company specializing in adhesives, coatings, vibration and motion control, and magnetically responsive technologies. They develop new materials that are used in critical applications such as military and aerospace systems. The composite materials can have defects which need to be detected during manufacturing. The goal set forth by LORD for this project was to determine a model from the acoustic emission data that could allow them to perform non-destructive inspection for defects of composite products manufactured.

This problem was presented as part of the 2012 Industrial Math/Stat Modeling Workshop for Graduate Students administered by the Statistical and Applied Mathematical Sciences Institute (SAMSI) and the Center for Research in Scientific Computation (CRSC) at North Carolina State University. Graduate students from around the world apply for the workshop, and are then presented a problem and given one week to work on solutions. Students come from mathematics, statistics, and engineering backgrounds and as a result approach the problems in different ways; together, the hope is that different backgrounds result in novel approaches to the problems. In this document, we demonstrate how combining different approaches and tests has resulted in the underpinnings of a novel methodology for using AE to analyze composite materials non-destructively.

2 Description of the Data

Understanding the data is the first step of analysis. The AE data obtained from LORD is separated into three groups: control, defect 1 and defect 2. Each group contains twenty tests Each test was performed on a separate composite assembly; the composites manufactured to be as identical as possible within each group. For each test, a load was applied and removed three times to evoke acoustic emissions, with the peak load increasing each time the load was re-applied. Finally, a fourth load was applied to break the composite material. The first two loads applied do not yield much emission data compared to the third load applied. Upon consultation with LORD, we largely did not use data from the fourth load because this load cycle is continued until the composite material breaks; we want to predict the performance of the part using only the acoustic emissions in the first three nondestructive loads. Once the composite is broken, it is useless (except, of course, for our testing purposes). We will examine the first three load blocks in prediction modeling. In particular, we will also examine the prediction power of the third block by itself since the first two blocks contain relatively fewer acoustic emissions than the third block.

For each of the three groups we were provided measurements of the acoustic signature for thirteen different AE parameters during the tests of each composite material, some of which are suspected to be relevant in a prediction model. We were also given the load applied to the sample at certain time steps. Figure 1 shows plots of load applied versus time for the control, defect 1, and defect 2 groups. We can view each plot as having four load cycles, which we also refer to as blocks; the composite breaks in the fourth block. For composites with defects, the plastic deformation occurs during the third load block at about half the load compared to the control data, which has deformation occurring on the fourth load around 3000 units. For data analysis we segment the data into blocks, using the times corresponding with lowest loads between each loading cycle as the separation points.

3 Exploratory Data Analysis

The general approach to statistical data analysis is to understand the data on a deeper level, in this case without the use of an underlying mathematical/physical model. For our problem, LORD has studied the physical properties of the composite material and wished for us to focus on statistical analysis of the AE data. Here we explore the plots of parameters versus time in comparison to the load versus time plots, as well as box plots and density plots of statistics on the parameters.

3.1 Visual representation of data samples

We first represent samples of data visually. Figures 2, 3, and 4 show plots of acoustic emissions over time for parameter 5 above the corresponding plots of time versus load for each of the first three blocks, for the control,

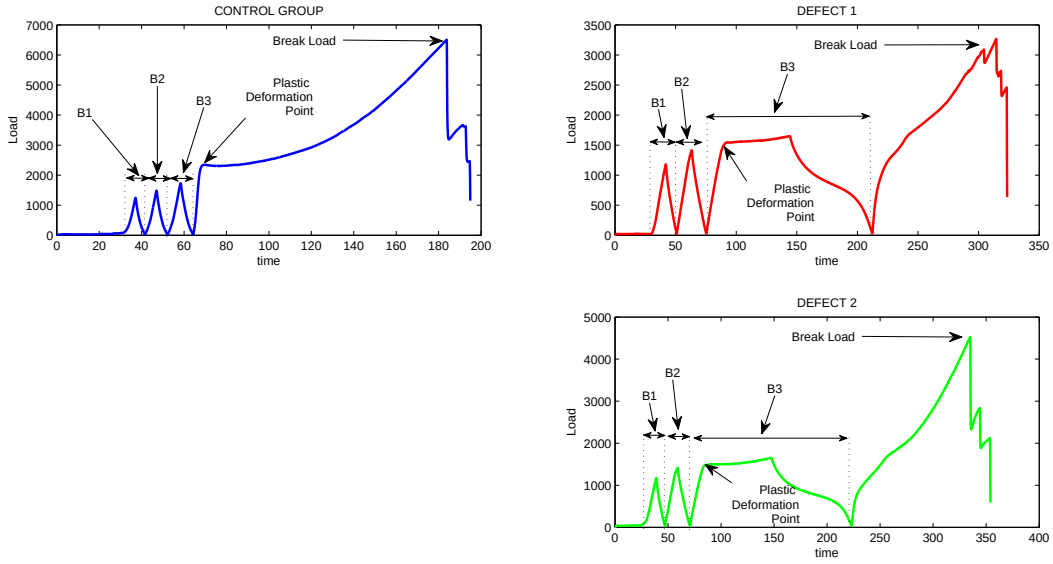


Figure 1: The load versus time (seconds) data for a test set of each group: control, defect 1 and 2. Blocks are labeled by B1: block 1, B2: block 2, B3: block 3

defect 1 and defect 2 groups, respectively. We observe that the majority of the emission data collected occurs near the peak of the load applied for each load block. This is most likely due to the Kaiser effect, which states that when a material is stressed to a certain load and then removed, the material will not emit again until the previous maximum stress level is exceeded [3].

3.2 Description of combined data

Having acoustic emission data for each group is useful, but individual emissions do not necessarily have much structure. Instead, we will use various methods for combining data in order to tie individual acoustic emissions together into groups like control, defect 1, and defect 2. One of these methods is simply combining all the acoustic emissions across all 20 data sets. To do this, we combine the AE data for each parameter of the 20 tests together for control, defect 1 and defect 2. This pools all the AE data together into classes, providing some additional structure for our analysis in the next two sections.

3.3 Box plots

In order to visualize the data and examine some basic statistics, we will create box plots [13]. A byproduct of this will be some statistical knowledge of each parameter, which will later be of use in creating density plots in the next section. To form a box plot, we complete the steps which follow.

1. The vertical axis displays the response variable while the horizontal axis displays the factor of interest.
2. Calculate the median, the 25^{th} percentile and the 75^{th} percentile quartiles.
3. Draw a line at the median and draw a box between the 25^{th} percentile and the 75^{th} percentile quartiles.
4. Calculate the interquartile range (IQR):

$$IQR = 75^{th} \text{ percentile value} - 25^{th} \text{ percentile value} \tag{1}$$

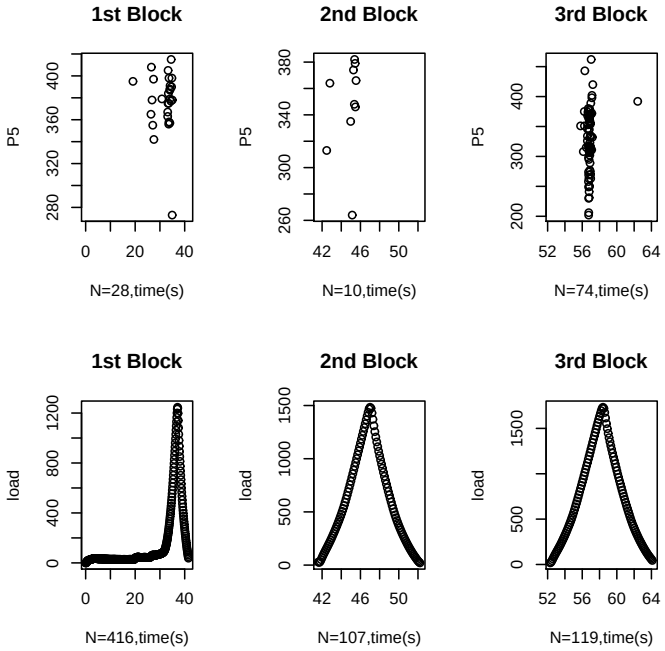


Figure 2: Scatter plot for Control group, data set 34178, for all three data blocks in this data set (top) Time (seconds) versus acoustic emission parameter 5 (P5). (bottom) Time (seconds) versus load. N is the number of data points in each sub-figure.

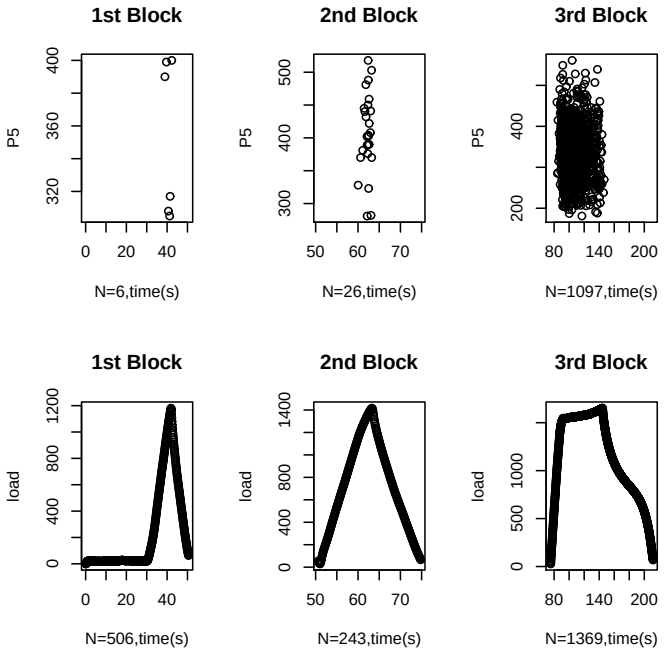


Figure 3: Scatter plot for defect 1 group, data set 34182, for all three data blocks in this data set (top) Time (seconds) versus acoustic emission parameter 5 (P5). (bottom) Time (seconds) versus load. N is the number of data points in each sub-figure.

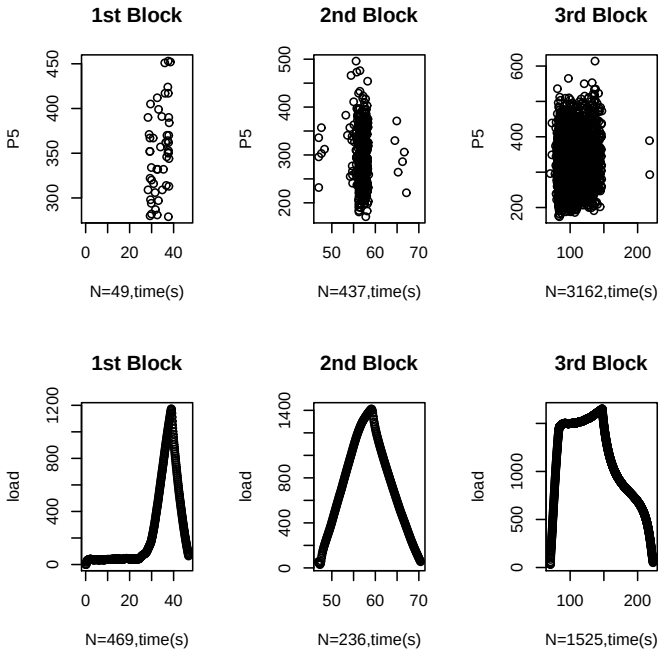


Figure 4: Scatter plot for defect 2 group, data set 34186, for all three data blocks in this data set (top) Time (seconds) versus acoustic emission parameter 5 (P5). (bottom) Time (seconds) versus load. N is the number of data points in each sub-figure.

5. Calculate the lower bound (L1) and upper bound (U1) as follows, and draw a line at each bound.

$$L1 = 25^{th}\text{percentile quartiles} - 1.5IQR \tag{2}$$

$$U1 = 75^{th}\text{percentile quartiles} + 1.5IQR \tag{3}$$

6. Draw a line through lower bound to upper bound.

7. Draw all the data outside the lower bound and upper bound, which is seen as the outliers.

Using this method, we can produce a box plot for each parameter. From these plots of control data (e.g., see Figure 5), we consistently found outliers above the "upper bound" data line. As these are far from the median, in our futher analysis using density plots we will exempt these points from the analysis (note that in later methods, we will utilize all the data). Similarly for the defect groups, there were upper bound outliers.

3.4 Density plot

Before creating density plots, we take the upper bound U1 as defined in Equation (3) as a data threshold for all the blocks and all the control, defect 1 and defect 2 groups; thus, only data below U1 is used. Then we created the density plot for the data less than the bound for each parameter. Here we use the Kernel Density Estimation [20] which is a method that uses a locally weighted averaging distribution. We assume the data is an independent, identically distributed sample from a normal distribution, and the estimation of the data density is a weighted average of the density at all the points. For the kernel K, we will use a Gaussian density. The formula for the estimation is

$$\hat{f}_h(x) = \frac{1}{n}\sum_{i=1}^{n} K_h(x - x_i) = \frac{1}{nh}\sum_{i=1}^{n} K(\frac{x - x_i}{h}) \tag{4}$$

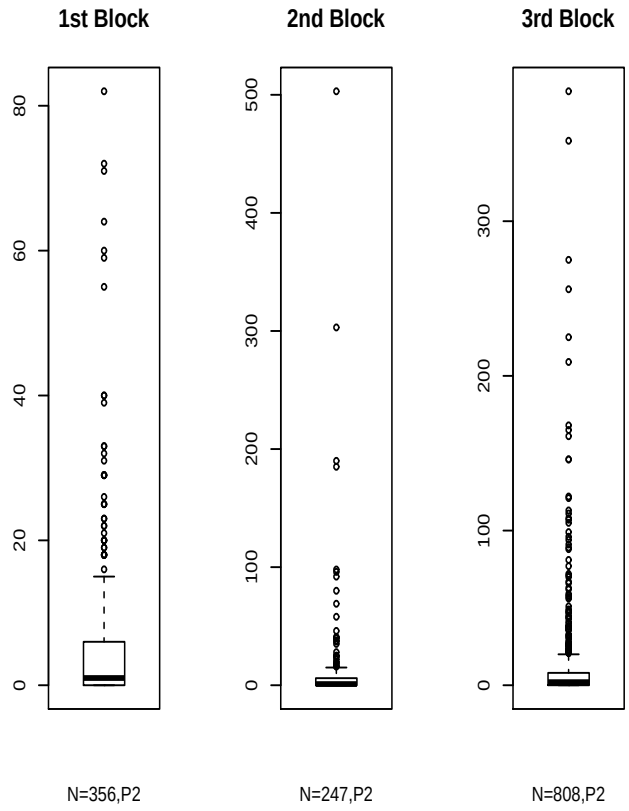


Figure 5: Box plot for parameter 2, combined data, control group, for each block. N is the number of data points in each sub-figure.

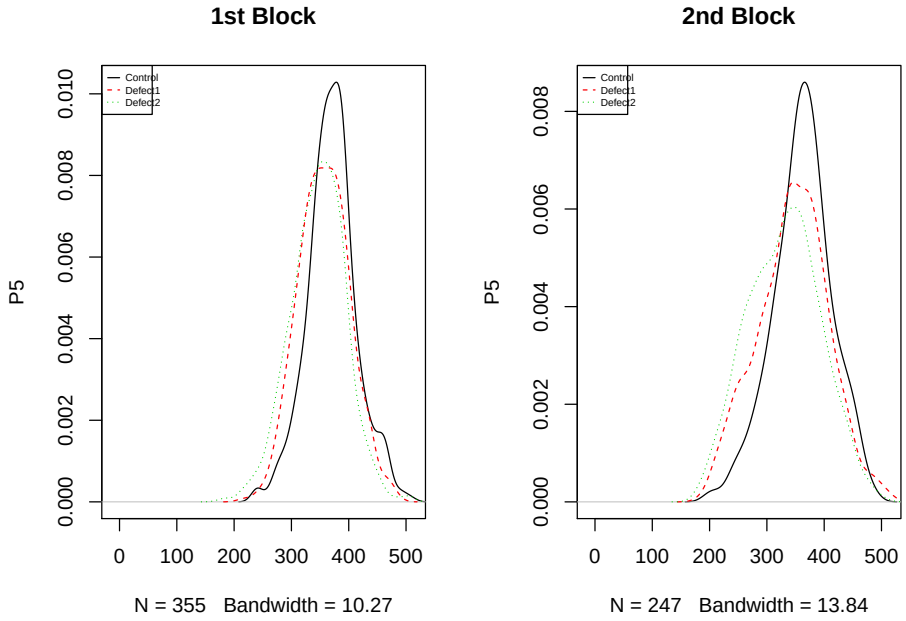


Figure 6: Density plot for parameter 5 of combined data in 1st block and 2nd block in control, defect 1 and defect 2 groups. N is the number of data points in each sub-figure.

We use Silverman's "rule of thumb" [19] to choose the bandwidth

$$h = \frac{0.9A}{n^{1/5}}, \tag{5}$$

where $A = \min\{\sigma, IQR\}$, σ is standard deviation of the points being estimated, IQR is the interquartile range as we calculated in (1), and the n is the number of data observations. So the estimation for the density is

$$\hat{f}_h(x) = \frac{1}{2\pi nh} \sum_{i=1}^{n} \exp\left[-\frac{(x - x_i)^2}{2h}\right]. \tag{6}$$

In Figures 6 and 7, we see that the density plots for the first two blocks seem to have similar means across the control and defective composites. However, in the third block the means of the defective composites are offset from the mean of the control group. This may indicate that the third block is important in classifying composites as controls or defects. Since the defective composites have similar means, we may have difficulty determining if a part belongs to the defect 1 or defect 2 group. Additionally, the number of data points in the third block is significantly higher for all tests than in the first two blocks. This is an indication of the importance of third block data.

4 Parameter Set Reduction

The first step of our analysis on this project was to examine ways to reduce the parameter set if possible. For each test performed, thirteen variables were recorded. However, LORD suggested that we begin by examining the importance of each parameter for its usefulness in predicting material type from acoustic emission data.

4.1 Correlation

The Pearson correlation coefficient, r, is a statistic that indicates the strength of the linear relationship between two quantitative variables [4]. The value of r is contained between the interval $[-1, 1]$, where an r

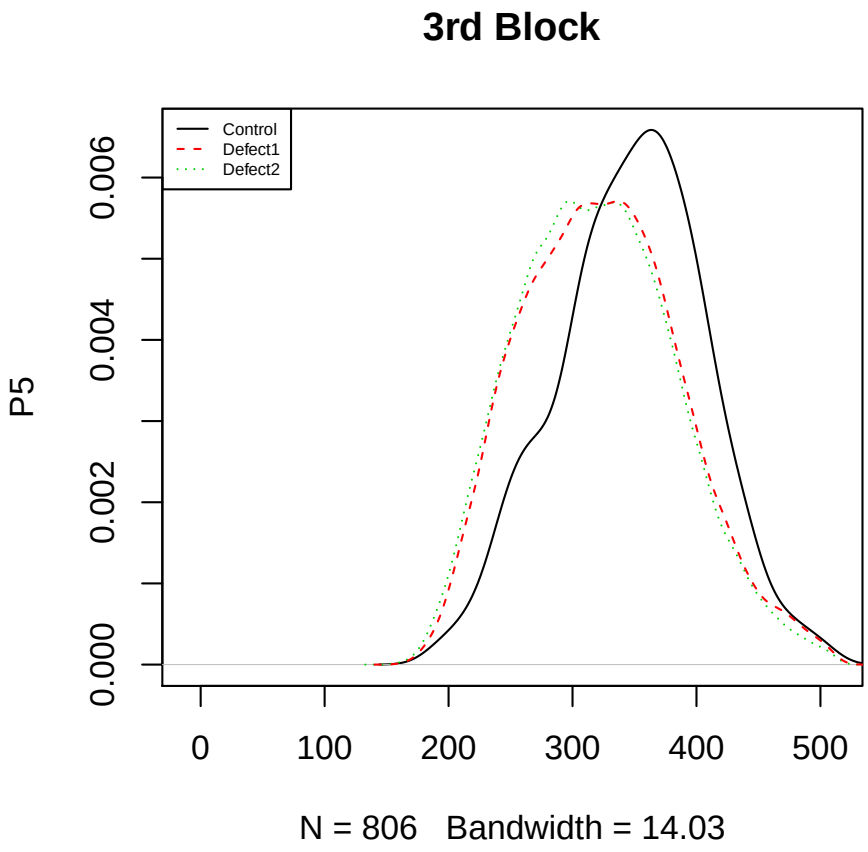


Figure 7: Density plot for parameter 5 of combined data in 3rd block in control, defect 1 and defect 2 groups. N is the number of data points in the figure.

Parameter Number	Parameter number												
	1	2	3	4	5	6	7	8	9	10	11	12	13
1	1.00	0.81	0.42	0.00	-0.25	0.56	0.58	-0.05	0.19	-0.11	0.04	0.13	0.81
2	0.81	1.00	0.67	0.01	-0.43	0.91	0.91	-0.11	0.52	-0.15	0.09	0.39	1.00
3	0.42	0.67	1.00	0.03	-0.70	0.74	0.78	-0.32	0.37	-0.18	0.20	0.24	0.68
4	0.00	0.01	0.03	1.00	0.06	0.03	-0.03	0.13	0.01	0.06	0.28	-0.03	0.01
5	-0.25	-0.43	-0.70	0.06	1.00	-0.47	-0.54	0.32	-0.22	0.34	-0.10	-0.15	-0.43
6	0.56	0.91	0.74	0.03	-0.47	1.00	0.95	-0.17	0.66	-0.12	0.13	0.49	0.91
7	0.58	0.91	0.78	-0.03	-0.54	0.95	1.00	-0.22	0.65	-0.18	0.00	0.54	0.91
8	-0.05	-0.11	-0.32	0.13	0.32	-0.17	-0.22	1.00	-0.16	0.10	-0.18	-0.17	-0.12
9	0.19	0.52	0.37	0.01	-0.22	0.66	0.65	-0.16	1.00	-0.02	0.07	0.92	0.52
10	-0.11	-0.15	-0.18	0.06	0.34	-0.12	-0.18	0.10	-0.02	1.00	0.05	-0.03	-0.15
11	0.04	0.09	0.20	0.28	-0.10	0.13	0.10	-0.18	0.07	0.05	1.00	0.04	0.09
12	0.13	0.39	0.24	-0.03	-0.15	0.49	0.54	-0.17	0.92	-0.03	0.04	1.00	0.39
13	0.81	1.00	0.68	0.01	-0.43	0.91	0.91	-0.12	0.52	-0.15	0.09	0.39	1.00

Table 1: Correlation coefficients for the thirteen parameters in the control data, block 3. Columns and rows are denoted by the parameter number. For example, the correlation between parameters 3 and 4 is 0.03.

value closer to -1 or 1 indicates a stronger linear relationship between the two variables than a value near 0. There is no variable dependence when r is close to 0. The sign on r indicates the direct ('+') or inverse ('−') relationship between the two variables. While there are other correlation coefficients that are more sensitive to nonlinear relationships, here we focus on Pearson's correlation coefficient, which is the most common method for determining correlation.

The sample correlation coefficient of the observed variables X and Y, both of size n, is defined as

$$r = \frac{\sum_{i=1}^{n}(x_i - \mu_x)(y_i - \mu_y)}{\sqrt{\sum_{i=1}^{n}(x_i - \mu_x)^2}\sqrt{\sum_{i=1}^{n}(y_i - \mu_y)^2}} \tag{7}$$

where μ_x and μ_y are the means of the data x and y, respectively [4].

In order to create a prediction model for the data in this paper, we first reduced the number of parameters by appealing to the correlation coefficients across the data set. When two parameters were determined to be highly correlated, we considered removing one of the two parameters from the parameter set as they both contributed similar information in prediction modeling, and thus it was unnecessary to retain redundant information. To determine which parameter was retained in the parameter set given two highly correlated parameters, we kept the parameter with the least variance, as per the request of LORD.

Considering only the third block of the AE data, we obtained correlation coefficients of each group. The correlation coefficients for the thirteen parameters in the third block of the control data are given in Table 1. Correlation coefficients for the third block of the defect 1 data set are given in Table 2 while those for defect 2 are given in Table 3. All twenty tests were compiled into one data set per each group (control, defect 1, and defect 2) for these calculations. Similar coefficients were produced when the entire data set (all three blocks) was considered.

Note the diagonal in the each correlation table all have value 1.00 because a variable will be perfectly correlated with itself. Furthermore, the correlation coefficient can be squared to explain how much variation in one variable is explained by a linear model on the other variable. For example, in the control data given in Table 1, parameters 6 and 7 have a correlation coefficient of 0.95. Thus $r^2 = 0.9025$ indicates that 95% of the variation in parameter 6 can be explained by a linear model on parameter 7, and vice versa, since the correlation coefficient matrix is symmetric.

After examining Tables 1, 2, and 3, a perfect correlation of 1.00 is observed between parameters 2 and 13. While parameters 6 and 7 have strong correlation of $r \approx 0.95$ regardless of the group (control, defect 1

Parameter Number	Parameter number												
	1	2	3	4	5	6	7	8	9	10	11	12	13
1	1.00	0.68	0.20	0.00	-0.08	0.35	0.34	-0.02	0.06	-0.02	0.02	0.06	0.68
2	0.68	1.00	0.65	0.02	-0.35	0.85	0.86	-0.12	0.33	-0.10	0.12	0.23	1.00
3	0.20	0.65	1.00	0.07	-0.65	0.89	0.85	-0.35	0.43	-0.13	0.34	0.24	0.65
4	0.00	0.02	0.07	1.00	0.07	0.07	-0.01	0.09	0.03	0.13	0.28	-0.06	0.01
5	-0.08	-0.35	-0.65	0.07	1.00	-0.49	-0.53	0.37	-0.23	0.46	-0.16	-0.12	-0.35
6	0.35	0.85	0.89	0.07	-0.49	1.00	0.96	-0.23	0.52	-0.09	0.27	0.34	0.85
7	0.34	0.86	0.85	-0.01	-0.53	0.96	1.00	-0.26	0.49	-0.15	0.18	0.40	0.86
8	-0.02	-0.12	-0.35	0.09	0.37	-0.23	-0.26	1.00	-0.26	0.13	-0.24	-0.28	-0.12
9	0.07	0.33	0.43	0.03	-0.23	0.52	0.49	-0.26	1.00	-0.02	0.15	0.78	0.33
10	-0.02	-0.10	-0.13	0.13	0.46	-0.08	-0.15	0.13	-0.02	1.00	0.07	-0.04	-0.10
11	0.02	0.12	0.34	0.28	-0.16	0.27	0.18	-0.24	0.15	0.07	1.00	0.07	0.12
12	0.06	0.23	0.24	-0.06	-0.12	0.34	0.40	-0.28	0.78	-0.04	0.07	1.00	0.23
13	0.68	1.00	0.65	0.02	-0.35	0.85	0.86	-0.13	0.33	-0.10	0.12	0.23	1.00

Table 2: Correlation coefficients for defect 1 data, block 3. Columns and rows are denoted by the parameter number. For example, the correlation between parameters 7 and 2 is 0.86.

Parameter Number	Parameter number												
	1	2	3	4	5	6	7	8	9	10	11	12	13
1	1.00	0.73	0.23	0.00	-0.09	0.46	0.42	-0.03	0.10	-0.02	0.03	0.07	0.73
2	0.73	1.00	0.65	0.01	-0.33	0.88	0.87	-0.11	0.33	-0.09	0.10	0.22	1.00
3	0.23	0.64	1.00	0.06	-0.66	0.86	0.82	-0.35	0.41	-0.12	0.33	0.23	0.64
4	0.00	0.01	0.06	1.00	0.07	0.05	-0.03	0.09	0.02	0.13	0.28	-0.06	0.00
5	-0.09	-0.33	-0.66	0.07	1.00	-0.48	-0.50	0.40	-0.23	0.45	-0.16	-0.11	-0.33
6	0.46	0.88	0.86	0.05	-0.48	1.00	0.96	-0.22	0.50	-0.08	0.23	0.32	0.88
7	0.42	0.87	0.82	-0.03	-0.50	0.96	1.00	-0.25	0.48	-0.14	0.15	0.39	0.87
8	-0.03	-0.11	-0.35	0.09	0.40	-0.22	-0.25	1.00	-0.25	0.13	-0.25	-0.27	-0.12
9	0.10	0.33	0.41	0.02	-0.23	0.50	0.48	-0.25	1.00	-0.03	0.13	0.79	0.33
10	-0.02	-0.09	-0.12	0.13	0.45	-0.08	-0.14	0.13	-0.03	1.00	0.09	-0.05	-0.09
11	0.03	0.10	0.33	0.28	-0.16	0.22	0.15	-0.25	0.13	0.09	1.00	0.06	0.10
12	0.07	0.22	0.23	-0.06	-0.11	0.32	0.39	-0.27	0.79	-0.05	0.06	1.00	0.22
13	0.73	1.00	0.64	0.00	-0.33	0.88	0.87	-0.12	0.33	-0.09	0.10	0.22	1.00

Table 3: Correlation coefficients for defect 2 data, block 3. Columns and rows are denoted by the parameter number. For example, the correlation between parameters 12 and 2 is 0.22.

	Variance		
Parameter	Control	Defect 1	Defect 2
1	9.7×10^8	4.4×10^9	5.6×10^9
2	1.0×10^3	5.1×10^2	7.3×10^2
3	6.7×10^1	6.4×10^1	6.9×10^1
4	9.3×10^3	7.4×10^3	7.1×10^3
5	3.5×10^3	4.4×10^3	4.3×10^3
6	1.2×10^4	3.5×10^3	4.2×10^3
7	1.1×10^6	3.9×10^5	5.0×10^5
8	8.1×10^4	8.1×10^4	7.9×10^4
9	7.1×10^2	9.4×10^1	1.1×10^2
10	3.3×10^3	3.6×10^3	3.5×10^3
11	7.0×10^3	4.2×10^3	4.0×10^3
12	8.4×10^4	1.2×10^4	1.7×10^4
13	3.9×10^{10}	2.0×10^{10}	2.8×10^{10}

Table 4: Variance of the thirteen parameters for each group (control, defect 1 and defect 2), using third block data only.

or defect 2), it appears that parameters 2 and 13 are also moderately correlated with parameters 6 and 7, with correlation values between .85 and .91. Thus, of the four parameters 2, 6, 7, and 13, only one should be used in prediction modeling, according to the correlation coefficient results. Referring to the variances of the parameters in Table 4, parameter 2 has the least variation, thus we may choose to discard parameters 6, 7, and 13.

Parameter 1 has a correlation coefficient of .81 with parameters 2 and 13 for the control data, and a correlation coefficient of .68 and .73 for defect 1 and defect 2 data, respectively. Thus it may be practical to also remove parameter 1 from the set since it has a larger variance than parameter 2. Lastly, parameters 9 and 12 range in correlation across the groups from $r = .78$ to $r = .91$. Since parameter 9 has less variance, we may opt to discard parameter 12.

Parameter reduction via correlation coefficients suggest the removal of parameters 1, 6, 7, 12, and 13, reducing the thirteen parameter set to eight, though further inspection using other statistical means is necessary.

4.2 Significance testing

In this section, we use hypothesis testing to determine if specific functions of the AE data are significantly different between the control and defective composites. Since we have many acoustic emissions in each data set, the functions we use will be means, medians, and maximums of the AE data between composite types for each test. We will also directly compare all the control and data set AE parameters for significance.

The significance will be determined by applying a t-test. First, we assume that the samples being compared follow a normal distribution with different variances. Accordingly, we use Welch's t-test [22], described in the next section.

4.2.1 Welch's t-test

Welch's t-test is a generation of Student's t-test, used with data sets that have different population variances [22], which is a reasonable assumption for our data sets. In general, the t-statistic is defined by [18]

$$t_0 = \frac{\mu_1 - \mu_2}{\sqrt{\frac{s_1^1}{N_1} + \frac{s_2^2}{N_2}}}$$

where μ_1 and μ_2 are the sample means, s_1^2 and s_2^2 are the sample variances, and N_1 and N_2 are the sample sizes for the two populations, respectively. Based on the normal assumption, t follows the Student's t distribution,

with the degrees of freedom ν defined by

$$\nu = \frac{\left(\frac{s_1^2}{N_1} + \frac{s_2^2}{N_2}\right)^2}{\frac{s_1^4}{N_1^2 V_1} + \frac{s_2^4}{N_2^2 V_2}}$$

where $V_1 = N_1 - 1$, $V_2 = N_2 - 1$ are based off the number of data N_i in the two data sets being compared. Our null hypothesis is that the means are equal between data sets. We then compute the p-value of Welch's t-test by

$$\text{p-value} = 2P(t > |t_0|),$$

where t is Student's t distribution with ν degrees of freedom. If the p-value is less than some user-defined significance level α, we say the test is significant, which means that we reject the null hypothesis (the same mean) at the level of α. If the p-value is greater than the significance level, we do not reject the null hypothesis. Here we choose the significant level $\alpha = 0.01$. In other words, if an AE parameter is determined to be significant, we think that signifies it is meaningful in differentiating between control and defective composites.

4.2.2 Comparing means

First, for the control group, we took the mean value of each parameter in the 3 blocks. Since we had 20 control data sets, we got 20 mean values for each block, μ_i^C. Then we found $\mu_C = \frac{1}{20}\sum_{i=1}^{20} \mu_i^C$ and the corresponding σ_C, the mean and standard deviation of the 20 control data means. We did the same computation for the defect 1 and defect 2 groups, obtaining the pairs μ_{D1}, σ_{D1} and μ_{D2}, σ_{D2}. Plug in the formulas for t test, and then we performed Welch's t-test for each block between each pairwise combination of control, defect 1 and defect 2 composites. For example, for the third block between control and defect 1 composites, examining parameter 5, the p-value for the third block was 7.9×10^{-5}, which came from the test statistic $t = 4.7747$ with $\nu = 23.303$. Since this p-value was less than $\alpha = 0.01$, we rejected the null hypothesis and believed there was a difference between means of parameter 5 for the control and defect 1 groups. We concluded that the parameter was important in differentiating between control and defect 1 parts.

The p-values for Welch's t-test were computed for all thirteen parameters in each block, comparing the means of the computed data means between the control and defect 1 groups (Table 5), between the control and defect 2 groups (Table 6), and between the defect 1 and defect 2 groups (Table 7). At the $\alpha = 0.01$ level, Table 5 indicated that parameters 5, 9 and 10 show significant difference between means of block 3 for the control and defect 1 groups. The mean of parameter 4 in the second block also showed significant differences between the control and defect 1 groups.

Given the same α level for Table 6, we saw that, again, parameters 5, 9 and 10 showed significant differences between means of block 3 for the control and defect 2 groups. In addition, the means of parameters 5 and 11 in the first block and the mean of parameter 4 in the second block showed significant differences between the control and defect 2.

However, using the same α level for Table 7, there were no parameters showing significant differences between defect 1 and defect 2 in any block. This might indicate that differentiating between defect 1 and defect 2 would be difficult.

4.2.3 Comparing medians

We used the same method as before, except now we took the median AE values across data sets instead of the mean. So, we found Med_i^C for the 20 control data sets. We then compute $\mu_C = \frac{1}{20}\sum_{i=1}^{20} Med_i^C$ and the corresponding σ_C of the median values. This was done for defect 1 and defect 2 data, as before. We then again computed the t-test for each block between each two of groups the control, defect 1 and defect 2.

The p-values for Welch's t-test were computed for all thirteen parameters for each block, comparing the means of the medians between the control and defect 1 groups (Table 8), between the control and defect 2 groups (Table 9), and between the defect 1 and defect 2 groups (Table 10). At the $\alpha = 0.01$ level, Table 8 indicated that parameters 5, 9, 10 and 12 showed significant differences between medians of block 3 for the control and defect 1 groups. Given the same α level for Table 9, we saw that, again, parameters 5, 10 and 12 showed significant differences between medians of block 3 for the control and defect 2 groups. In addition, the medians of parameters 4 and 5 in the first block and the median of parameter 4 in the second block showed significant difference between the control and defect 2.

Parameter number	Block 1	Block 2	Block 3
1	0.180	0.298	7.6×10^{-1}
2	0.474	0.330	2.7×10^{-1}
3	0.066	0.984	7.4×10^{-2}
4	0.567	0.006	8.2×10^{-1}
5	0.044	0.230	7.9×10^{-5}
6	0.682	0.300	1.7×10^{-1}
7	0.617	0.123	1.2×10^{-1}
8	0.320	0.582	6.7×10^{-1}
9	0.904	0.353	4.7×10^{-3}
10	0.523	0.305	2.1×10^{-4}
11	0.014	0.212	9.9×10^{-1}
12	0.461	0.277	2.9×10^{-2}
13	0.474	0.311	2.8×10^{-1}

Table 5: The p-value for the t-tests computed of each block between control and defect 1, comparing the means for each test.

Parameter number	Block 1	Block 2	Block 3
1	0.068	0.863	2.7×10^{-1}
2	0.810	0.660	7.0×10^{-1}
3	0.025	0.593	1.2×10^{-2}
4	0.280	0.009	8.0×10^{-1}
5	1.2×10^{-4}	0.074	9.7×10^{-6}
6	0.397	0.347	2.7×10^{-1}
7	0.036	0.191	2.7×10^{-1}
8	0.284	0.329	8.7×10^{-1}
9	0.195	0.227	7.2×10^{-3}
10	0.605	0.473	1.1×10^{-5}
11	8.7×10^{-4}	0.090	7.9×10^{-1}
12	0.034	0.157	7.2×10^{-2}
13	0.812	0.650	7.1×10^{-1}

Table 6: The p-value for the t-tests computed of each block between control and defect 2, comparing the mean value for each test.

Parameter number	Block 1	Block 2	Block 3
1	0.511	0.156	0.042
2	0.360	0.315	0.014
3	0.513	0.577	0.019
4	0.380	0.877	0.854
5	0.026	0.418	0.030
6	0.203	0.830	0.226
7	0.122	0.660	0.124
8	0.949	0.647	0.114
9	0.157	0.462	0.371
10	0.821	0.829	0.066
11	0.637	0.695	0.207
12	0.208	0.403	0.091
13	0.358	0.314	0.014

Table 7: The p-value for the t-tests computed of each block between defect 1 and defect 2, comparing the mean value for each test.

Parameter number	Block 1	Block 2	Block 3
1	0.961	0.360	7.1×10^{-1}
2	0.464	0.193	9.0×10^{-1}
3	0.024	0.842	1.3×10^{-1}
4	0.020	0.030	2.4×10^{-1}
5	0.070	0.644	5.7×10^{-5}
6	0.548	0.384	7.5×10^{-1}
7	0.074	0.153	6.0×10^{-1}
8	0.392	0.982	2.7×10^{-2}
9	0.790	0.313	7.2×10^{-3}
10	0.798	0.156	7.7×10^{-5}
11	0.035	0.188	5.2×10^{-2}
12	0.305	0.267	5.5×10^{-4}
13	0.446	0.210	6.9×10^{-1}

Table 8: The p-value for the t-tests computed of each block between control and defect 1, comparing the medians for each test.

However, using the same α level for Table 10, there were no parameters showing significant differences between defect 1 and defect 2. This again indicated that determining between defect 1 and defect 2 might prove difficult.

4.2.4 Comparing maximums

We now used the same methodology, except we took the maximum of the AE data in each of the 20 data sets for the three types of material. The p-values for Welch's t-test were computed for all thirteen parameters for each block, comparing the means of the maximums between the control and defect 1 groups (Table 11), between the control and defect 2 groups (Table 12), and between the defect 1 and defect 2 groups (Table 13). At the $\alpha = 0.01$ level, Table 11 indicated that parameters 2, 3, 4, 5, 11 and 13 show significant differences between medians of block 3 for the control and defect 1 groups.

Given the same α level for Table 12, we saw that, all but parameter 8 showed significant differences between medians of block 3 for the control and defect 2 groups. In addition, the medians of parameters 2, 3, 11 and 13 in the first block and the medians of parameter 3, 4, 5, 8, 10 and 11 in the second block showed significant

Parameter number	Block 1	Block 2	Block 3
1	0.635	0.255	9.9×10^{-1}
2	0.166	0.184	1.0
3	0.051	0.799	2.8×10^{-2}
4	0.003	4.8×10^{-4}	1.4×10^{-1}
5	1.9×10^{-4}	0.175	9.4×10^{-6}
6	0.978	0.209	9.5×10^{-1}
7	0.021	0.129	6.7×10^{-1}
8	0.432	0.619	7.6×10^{-2}
9	0.970	0.214	1.1×10^{-2}
10	0.666	0.527	6.0×10^{-7}
11	0.013	0.118	1.1×10^{-2}
12	0.026	0.119	6.7×10^{-4}
13	0.118	0.140	9.0×10^{-1}

Table 9: The p-value for the t-tests computed of each block between control and defect 2 comparing the median values for each test.

Parameter number	Block 1	Block 2	Block 3
1	0.577	0.631	0.202
2	0.554	1.000	0.701
3	0.777	0.595	0.051
4	0.599	0.511	0.438
5	0.063	0.299	0.064
6	0.480	0.615	0.454
7	0.349	0.845	0.725
8	0.727	0.488	0.229
9	0.731	0.384	0.504
10	0.458	0.282	0.045
11	0.859	0.474	0.038
12	0.467	0.213	0.747
13	0.435	0.735	0.362

Table 10: The p-value for the t-tests computed of each block between defect 1 and defect 2 comparing the median values for each test.

Parameter number	Block 1	Block 2	Block 3
1	0.162	0.798	2.0×10^{-2}
2	0.162	0.823	6.5×10^{-4}
3	0.083	0.097	3.7×10^{-7}
4	0.788	0.004	5.5×10^{-9}
5	0.441	0.012	5.8×10^{-9}
6	0.244	0.548	9.8×10^{-2}
7	0.317	0.746	1.8×10^{-2}
8	0.416	0.111	1.6×10^{-1}
9	0.186	0.509	1.3×10^{-1}
10	0.861	0.013	2.4×10^{-12}
11	0.158	0.003	5.6×10^{-5}
12	0.227	0.509	9.1×10^{-2}
13	0.163	0.823	6.5×10^{-4}

Table 11: The p-value for the t-tests computed of each block between control and defect 1, comparing the maximum value for each test.

Parameter number	Block 1	Block 2	Block 3
1	1.2×10^{-2}	2.2×10^{-1}	1.7×10^{-3}
2	8.8×10^{-3}	9.2×10^{-2}	1.7×10^{-6}
3	1.6×10^{-6}	1.4×10^{-4}	6.4×10^{-9}
4	2.1×10^{-2}	1.1×10^{-5}	3.6×10^{-9}
5	1.7×10^{-2}	1.6×10^{-6}	2.8×10^{-15}
6	1.2×10^{-1}	8.0×10^{-2}	6.3×10^{-4}
7	1.3×10^{-1}	7.6×10^{-2}	2.3×10^{-5}
8	8.3×10^{-2}	5.4×10^{-3}	4.2×10^{-2}
9	1.6×10^{-1}	3.6×10^{-1}	1.7×10^{-3}
10	1.8×10^{-3}	2.5×10^{-8}	5.1×10^{-17}
11	2.6×10^{-5}	5.4×10^{-6}	7.9×10^{-9}
12	3.4×10^{-2}	7.1×10^{-2}	1.8×10^{-4}
13	8.8×10^{-3}	9.2×10^{-2}	1.7×10^{-6}

Table 12: The p-value for the t-tests computed of each block between control and defect 2 comparing the maximum value for each test.

differences between the control and defect 2 groups.

However, using the same α level for Table 13, only parameters 7 and 12 showed significant differences between defect 1 and defect 2 in the third block; only parameter 3 showed significant differences between defect 1 and defect 2 in the first block. This might indicate that some parameters could differentiate between defective parts.

4.3 Significance tests on combined data

Instead of taking the means/medians/maximums of the acoustic parameters in the 60 data sets individually, we would examine if anything changed when we first combined all the AE data of the 20 tests as one test for the control group, defect 1 group, and defect 2 group individually and then comparing the AE parameters using the t-test between the three groups. We used the combined data as described in Sec. 3.2 and computed Welch's t-test. The p-values for Welch's t-test were computed for all thirteen parameters for each block, between the control and defect 1 groups (Table 14), between the defect 1 and defect 2 groups (Table 15) and between the control and defect 2 groups (Table 16).

Parameter number	Block 1	Block 2	Block 3
1	0.738	0.206	0.183
2	0.872	0.095	0.028
3	0.001	0.029	0.011
4	0.003	0.219	0.162
5	0.149	0.345	0.206
6	0.999	0.180	0.001
7	0.665	0.083	0.001
8	0.162	0.166	0.389
9	0.651	0.975	0.008
10	0.002	0.101	0.007
11	0.003	0.004	0.018
12	0.415	0.282	0.000
13	0.869	0.094	0.028

Table 13: The p-value for the t-tests computed of each block between defect 1 and defect 2 comparing the maximum value for each test.

Parameter number	Block 1	Block 2	Block 3
1	1.9×10^{-1}	0.290	6.1×10^{-1}
2	4.7×10^{-1}	0.298	5.8×10^{-3}
3	8.6×10^{-2}	0.409	1.6×10^{-1}
4	1.6×10^{-1}	0.006	5.7×10^{-1}
5	4.1×10^{-5}	0.000	9.5×10^{-25}
6	8.1×10^{-1}	0.217	9.8×10^{-5}
7	4.4×10^{-1}	0.096	3.6×10^{-6}
8	6.0×10^{-1}	0.976	8.6×10^{-2}
9	9.3×10^{-1}	0.171	1.2×10^{-5}
10	9.6×10^{-2}	0.100	5.3×10^{-14}
11	5.1×10^{-3}	0.134	6.7×10^{-1}
12	7.3×10^{-1}	0.153	8.6×10^{-4}
13	4.7×10^{-1}	0.293	5.9×10^{-3}

Table 14: The p-value for the t-tests computed of each block between control and defect 1 comparing the means for the combined data of all tests.

At the $\alpha = 0.01$ level, Table 14 indicated that parameters 2, 5, 6, 7, 9, 10, 12 and 13 showed significant differences between data of block 3 for the control and defect 1 groups. In addition, parameters 5 and 11 in the first block and parameter 4 and 5 in the second block showed significant differences between the control and defect 1 groups.

Given the same α level for Table 15, we saw that, parameters 3, 5, 6, 7, 9, 10 and 12 showed significant differences between data of block 3 for the control and defect 2 groups. In addition, parameters 3,4,5,7 and 11 in the first block and parameter 4, 5 and 11 in the second block showed significant differences between the control and defect 2.

Using the same α level for Table 16, we saw that, parameters 2, 3, 5, 6, 7, 10, 11, 12, and 13 showed significant differences between data of block 3 for the defect 1 and defect 2 groups. In addition, parameters 5 in the first block and parameter 5 and 10 in the second block showed significant differences between the defective groups.

Parameter number	Block 1	Block 2	Block 3
1	3.6×10^{-2}	8.4×10^{-1}	5.6×10^{-1}
2	8.4×10^{-1}	8.0×10^{-1}	5.7×10^{-2}
3	3.3×10^{-3}	1.7×10^{-2}	4.2×10^{-3}
4	8.4×10^{-3}	8.2×10^{-3}	6.0×10^{-1}
5	8.4×10^{-20}	1.9×10^{-16}	34.0×10^{-34}
6	2.2×10^{-1}	3.7×10^{-1}	4.0×10^{-4}
7	9.5×10^{-3}	2.7×10^{-1}	7.4×10^{-5}
8	8.4×10^{-1}	7.8×10^{-1}	2.1×10^{-1}
9	1.6×10^{-1}	1.5×10^{-1}	1.8×10^{-5}
10	3.2×10^{-2}	3.0×10^{-1}	7.5×10^{-23}
11	3.0×10^{-4}	3.2×10^{-3}	9.3×10^{-1}
12	6.1×10^{-2}	2.0×10^{-1}	3.3×10^{-3}
13	8.5×10^{-1}	7.9×10^{-1}	5.9×10^{-2}

Table 15: P-value of t-test for combination data between control and defect 2

Parameter number	Block 1	Block 2	Block 3
1	5.2×10^{-1}	7.9×10^{-2}	1.5×10^{-2}
2	3.9×10^{-1}	4.8×10^{-2}	7.7×10^{-8}
3	3.3×10^{-1}	1.4×10^{-2}	4.1×10^{12}
4	2.1×10^{-1}	5.3×10^{-1}	8.1×10^{-1}
5	3.1×10^{-6}	9.2×10^{10}	3.6×10^{18}
6	2.2×10^{-1}	3.7×10^{-1}	2.2×10^{-3}
7	9.6×10^{-2}	1.4×10^{-1}	2.8×10^{-7}
8	3.3×10^{-1}	5.8×10^{-1}	2.1×10^{-2}
9	3.1×10^{-1}	9.5×10^{-1}	2.2×10^{-1}
10	8.8×10^{-1}	3.4×10^{-5}	1.0×10^{27}
11	9.5×10^{-1}	4.8×10^{-2}	1.8×10^{-3}
12	2.7×10^{-1}	6.3×10^{-1}	4.1×10^{-6}
13	3.9×10^{-1}	4.7×10^{-2}	6.9×10^{-8}

Table 16: P-value of t-test for combination data between defect 1 and defect 2

4.3.1 Summary of significance testing

Given all the results we get from the significance tests, we can clearly see that parameters 5, 9, and 10 show significant differences in all the tests based on different statistics. Parameters 2, 3, 6, 7, and 12 showed significance in most of the tests. Thus, from significance testing the recommended parameter set would be parameters 2, 3, 5, 6, 7, 9, 10, and 12.

4.4 Principal component analysis (PCA)

Principal component analysis (PCA) is a mathematical procedure that computes the orthogonal components of a data set in which the majority of the variability of the data lie [21]. As previously discussed, it can be inefficient to use all variables to analyze the data, especially when variables are largely correlated to each other. PCA considers the set of observations with possibly correlated variables and returns a set of values of linearly uncorrelated variables called principal components.

A benefit of PCA is that the number of principal components returned is less than or equal to the original number of variables. The principal components are organized by PCA such that the first principal component accounts for as much of the variability in the original variables as possible, with each successive component accounting for as much variability as possible with the requirement that it be orthogonal to the previously defined principal components.

Analysis with PCA can be performed using Matlab's `princomp` function. This function requires the data matrix, X, as input with each column representing a variable and each row an observation. To reduce the variation between variables, it is recommended to first standardize the data matrix with the command `zscore`.

The essence of computing the principal components consists of centering the data matrix, such that $\tilde{X} = X - \bar{X}$ (where $\bar{X}$ is the mean of the data matrix). The eigen-decomposition of $\tilde{X}'\tilde{X} = VD^2V'$ is computed, where V is the matrix of principal components and D^2 is a diagonal matrix of the eigenvalues of $\tilde{X}'\tilde{X}$ [8]. Moreover the columns of V, denoted as $\boldsymbol{v}_i$, are the eigenvectors of $\tilde{X}'\tilde{X}$.

The first principal component $\boldsymbol{v}_1$ is used to calculate $\boldsymbol{z}_1 = \tilde{X}\boldsymbol{v}_1$, a vector that contains the largest variance of all the normalized linear combinations of the columns of $\tilde{X}$. The eigenvalues corresponding to each principal component describe how much variation in the data that particular component describes. The principal components indicate how much of each variable is contributing to the variation in each orthogonal direction.

PCA was performed on the first test of each group using all three blocks. The first four eigenvalues for the control group are $d_1 = 0.520$, $d_2 = 0.171$, $d_3 = 0.112$, and $d_4 = 0.081$. Adding these four eigenvalues indicates that the first four principal components describe 88.4% of the variation in the control data set. The corresponding principal components for test 1 of the control are in Table 17. To calculate the role of importance of each parameter, we multiply the eigenvalue times the absolute value of the eigenvector and sum across the first four components. We find the original variables 4, 8, 10, and 11 play the least important roles in the data sets. This is repeated for each of the remaining 19 control tests.

The same process was repeated for the defect 1 data, test one. The first four eigenvalues, $d_1 = 0.416$, $d_2 = 0.152$, $d_3 = 0.115$, and $d_4 = 0.102$ indicate that 79.5% of the variation in this data set is captured by the first four components given in Table 18. Computing the importance of the parameters as above, we found the same result that the original variables 4, 8, 10, and 11 play the least important role in the data sets. This result was confirmed across all 20 tests of the defect 1 data.

The same process was repeated for the defect 2 data, test one. The first four eigenvalues, $d_1 = 0.452$, $d_2 = 0.130$, $d_3 = 0.111$, and $d_4 = 0.100$ indicate that 79.3% of the variation in this data set is captured by the first four components given in Table 19. Computing the importance of the parameters as above, we found the a nearly same result as before that the original variables 4, 8, and 11 play the least important role in the data sets. This result was confirmed across all 20 tests of the defect 1 data.

The common least important parameters between the three groups of acoustic emissions are parameters 4, 8, and 11. Therefore, based on PCA, we determined that the following nine parameters contribute the most information towards the variation in the original AE data sets: 1, 2, 3, 5, 6, 7, 9, 12, and 13.

4.5 Generalized Linear Model (GLM)

Linear regression is a very common technique in statistics that is used to relate independent variables to some response. In our case, however, the "response" we wish to examine is categorical, namely whether the part is

Parameter	1st component	2nd component	3rd component	4th component
1	0.365	0.067	-0.181	0.028
2	0.368	-0.165	-0.059	0.007
3	0.335	0.238	-0.051	0.157
4	0.211	0.434	0.061	0.226
5	-0.023	0.462	0.228	0.111
6	0.359	0.148	-0.203	0.043
7	0.302	-0.376	0.133	0.024
8	-0.154	0.052	-0.630	-0.104
9	0.356	0.164	-0.168	-0.074
10	-0.080	-0.149	0.143	0.876
11	0.057	0.355	0.509	-0.297
12	0.238	-0.381	0.373	-0.189
13	0.371	-0.152	-0.038	0.024

Table 17: The first four principal components (eigenvectors) for test one of the control group using all three blocks.

Parameter	1st component	2nd component	3rd component	4th component
1	0.232	0.488	0.027	-0.087
2	0.368	0.35	0.027	-0.054
3	0.375	-0.155	-0.048	0.189
4	0.023	-0.015	0.387	0.551
5	-0.249	0.214	0.458	-0.222
6	0.408	-0.017	0.074	0.038
7	0.406	-0.019	0.001	-0.0490
8	-0.148	0.342	0.220	0.015
9	0.233	-0.374	0.287	-0.302
10	-0.0818	0.075	0.619	-0.006
11	0.159	-0.231	0.214	0.527
12	0.181	-0.357	0.266	-0.474
13	0.368	0.351	0.026	-0.054

Table 18: The first four principal components (eigenvectors) for test one of the defect 1 group using all three blocks.

Parameter	1st component	2nd component	3rd component	4th component
1	0.286	0.414	-0.092	0.040
2	0.374	0.245	0.014	0.196
3	0.345	-0.259	0.183	0.113
4	0.019	-0.003	0.605	-0.116
5	-0.229	0.480	0.146	-0.196
6	0.388	-0.034	0.141	0.091
7	0.392	0.006	0.013	0.112
8	-0.138	0.437	0.174	0.296
9	0.268	0.024	-0.111	-0.532
10	-0.071	0.297	0.460	-0.278
11	0.140	-0.332	0.463	-0.195
12	0.217	0.145	-0.287	-0.591
13	0.374	0.243	0.013	0.195

Table 19: The first four principal components (eigenvectors) for test one of the defect 2 group using all three blocks.

defective or control. Linear regression can be modified to incorporate such binary data using the generalized linear model framework in [12].

4.5.1 Method description

To make a model, our response data is control or defect, which is binary. We can consider that it is chosen from a Bernoulli distribution instead of a Normal distribution. Here we assume that Y is a random variable with a Bernoulli density, which is defined by

$$p(Y = 1) = p; \quad p(Y = 0) = 1 - p; \quad p \in (0, 1), \tag{8}$$

where $Y = 0$ represents the control group, and $Y = 1$ represents the defect group. The value of p is related to determining the particular class for a data set. If $p < 0.5$, the data point in question would be classified as a control group; for $p > 0.5$, the data would be classified as defective.

With this framework, we develop a Generalized Linear Model (GLM) [7]. In order to link the usual linear regression [11] model $Y = X\beta$ (where Y is assumed to follow a normal distribution) to a binary model like ours, we require a link function. The standard link function used in the binary classification model is the *logit* link

$$g(p) = \ln\Big(\frac{p}{1-p}\Big), \tag{9}$$

which transforms p from $(0, 1)$ to $(-\infty, +\infty)$. We then assume $g(p) = X\beta$, linking the normally distributed variable on the right hand side with the Bernoulli-distributed variable on the left hand side. Inverting the relationship, we can form the generalized linear model and find p by

$$\mathrm{E(Y)} = p = g^{-1}(\mathrm{X}\beta). \tag{10}$$

Since Y is distributed as a Bernoulli random variable (taking on values of either 0 or 1), we can easily write down the log likelihood

$$l(\beta) = \sum_{i=1}^{n} [y_i \log(X_i\beta) + (1 - y_i) \log(1 - X_i\beta)]. \tag{11}$$

We obtained the vector parameter β by using Maximum Likelihood Estimation.

Coefficients	Estimation	Std. Error	z value	P value
β_0	-2.745	2.364×10^{-1}	-11.610	$< 10^{-15}$
β_1	-5.913×10^{-6}	3.943×10^{-7}	-14.996	$< 10^{-15}$
β_2	9.545×10^{-3}	3.538×10^{-2}	0.270	0.787
β_3	1.551×10^{-1}	4.489×10^{-3}	34.554	$< 10^{-15}$
β_4	-2.317×10^{-4}	1.107×10^{-4}	-2.093	0.036
β_5	-9.292×10^{-3}	2.436×10^{-4}	-38.146	$< 10^{-15}$
β_6	-1.222×10^{-2}	1.078×10^{-3}	-11.336	$< 10^{-15}$
β_7	-2.763×10^{-3}	8.021×10^{-5}	-34.450	$< 10^{-15}$
β_8	7.609×10^{-4}	3.982×10^{-5}	19.108	$< 10^{-15}$
β_9	-4.809×10^{-2}	2.161×10^{-3}	-22.249	$< 10^{-15}$
β_{10}	-2.188×10^{-3}	1.898×10^{-4}	-11.526	$< 10^{-16}$
β_{11}	-5.533×10^{-4}	1.444×10^{-4}	-3.833	0.0001
β_{12}	4.362×10^{-3}	1.640×10^{-4}	26.601	$< 10^{-16}$
β_{13}	8.629×10^{-6}	5.664×10^{-6}	1.523	0.127

Table 20: GLM results for the model with respect to all parameters

4.5.2 GLM parameter significance determination

First, we tried the GLM for all the data for control and defect 1 using all the parameters. The model is

$$\mathrm{E(Y)} = p = g^{-1}(\beta_0 + \sum_{i=1}^{13} \mathrm{X_i}\beta_i). \tag{12}$$

Since the number of data for control is too small to do the GLM, we randomly sampled the control data multiple times to obtain a data set with numbers of AE events on a similar level as the number of AE events for defect 1.

The GLM results were showing at Table 20. The Estimation column gave the values obtained for each coefficient β_i. The Std. Error was the standard error of the estimated values. These values were not necessarily our main objective; we wished to determine which values were significant in matching the data.

To check which parameter was significant in this GLM, we assumed each coefficient had a normal density. The null hypothesis for each component was that the component was 0. We then ran a significance test to see if the coefficient should be nonzero. The z statistic was found by normalizing the coefficients

$$z_0 = \frac{\text{Estimate}}{\text{Std.Error}}, \tag{13}$$

and the p-value is calculated by

$$\text{p-value} = p(z > |z_0|), \tag{14}$$

where z had a standard normal distribution. We again chose the significance value $\alpha = 0.01$. When the p-value < 0.01, we rejected the null hypothesis, which means we thought that the coefficient was significantly different from 0. Based on Table 20, we can see that the parameters 1, 3, 5, 6, 7, 8, 9, 10, 11, and 12 are significantly different from 0.

Parameter subsets can also be found through model comparison tests, like the Akaike Information Criterion (AIC) [1, 2]. The results for significant parameters were similar to those obtained using the comparison test previously described, so we did not include results here. However, using AIC (or other performance criterion) could be important if one wishes to examine further parameter classification methods.

4.5.3 Testing the GLM model

Using the significant parameters, we tested to see if parts were being properly classified. The example here would be using the defect 2 data sets. We used the model for each data set, finding the predicted p-value for

0.6061306	0.5879914	0.5621036	0.6230668	0.5694919
0.5919222	0.6137649	0.5783133	0.6160086	0.5962945
0.5704178	0.5925726	0.6257801	0.5978169	0.6185930
0.5872174	0.5794906	0.6026244	0.5756116	0.5544666

Table 21: The mean of prediction value of p of each test for defect 2 group using the GLM with respect to all parameters.

each data set. Based on (8), $p > 0.5$ means the group was more likely to be defective, otherwise the composite was more likely to be a control composite. We calculate the mean of the p for each test of defect 2 groups shown in Table 21. The means in Table 21 were all greater than 0.5, which meant the part were more likely to be classified as defective. In addition, Figure 4.5.3 showed the density plot of the p in test 34233. We could see that most of the data ($2147/3154 = 68\%$), which would predict that the part is defective with high probability.

4.6 Additional Parameter Selection Methods

We also tried Factor Analysis [15], which is somewhat similar to PCA, and Bayes-based data analysis [6]. Factor Analysis resulted in a parameter subset similar to those we have already found, and Bayes-based data analysis was unsuccessful on our data set. Given time constraints, we have chosen to include no further information on these tests.

4.7 Selecting the final parameter sets

To summarize, the parameters selected by the various methods are found in Table 22. In addition, without revealing the physical meaning behind the thirteen parameters, it has been determined that parameters 1 and 6 have an essential physical meaning to the analysis. Recall that parameters 1, 2, and 13 have been found to be correlated, as well as the sets $\{3, 6, 7\}$ and $\{9, 12\}$. Taking the correlation into account and choosing the correlated parameters with least variance, the common parameters observed between all the methods are $\theta_1 = \{2, 3, 5, 8, 10\}$. We will refer to this set of five parameters as our core set of parameters. Based on additional information provided by the project sponsor, we also felt it may be beneficial in prediction modeling to keep a few of the other parameters in the final parameter set based on this physical knowledge, so we choose to also use the set $\theta_2 = \{2, 3, 4, 5, 8, 9, 10, 11\}$. Further modeling will use both sets in comparison.

Method	Parameter Set
Correlation Coefficients	$\{2, 3, 4, 5, 8, 9, 10, 11\}$
t-test	$\{1, 2, 3, 5, 6, 9, 10, 12\}$
PCA	$\{1, 2, 3, 5, 6, 7, 9, 10, 12, 13\}$
GLM	$\{1, 3, 5, 6, 7, 8, 9, 10, 12\}$

Table 22: Summary of the various approaches to parameter reduction. The parameter set in this table is the set of parameters suggested by each particular method to use in prediction modeling.

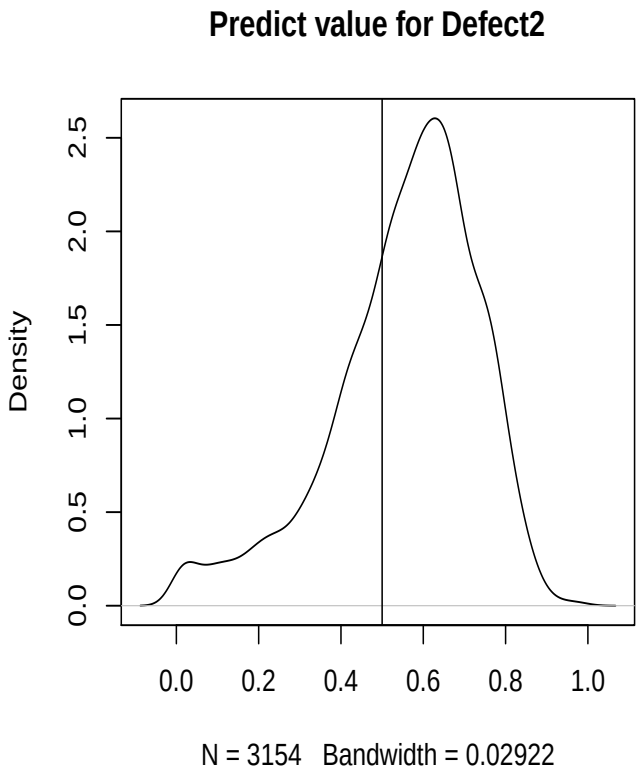


Figure 8: Prediction for Defect 2 using the GLM with respect to all parameters. The vertical line is $p = .5$

5 Classification: Discriminating Between Controls and Defects

The problem of classification seeks to segment a given data set into classes, or groups. Classification techniques are used on a wide range of scientific problems. For example, classification is used in spam detection systems for email and for patient diagnosis in medicine [8]. Furthermore, classification is used in image processing when ecologists wish to classify aerial images in order to determine how landscapes change over time [16], and the U.S. Post Office uses classification techniques for handwriting recognition [8]. Without an underlying mathematical model, we turn to classification techniques in order to predict if data comes from control or defective groups.

Supervised classification methods require a priori knowledge about the data set, i.e., the user must provide the set of classes in the image, as well as a training field for each class. The set of training fields is the training data. For the acoustic emissions data, the training data would be akin to an acoustic emission vector of parameter emission data with its corresponding group: control, defect 1 or defect 2. A classifier is built from the training data, which is then applied to the test set. The test set is the set of remaining data for which the corresponding class is unknown.

Typically in classification, a vector of data corresponds directly to a group. For example, in image processing, a single pixel vector corresponds to a particular group (grass, field, trees, water, etc.) while in medical diagnosis, each data set of medical observations corresponds to a single patient. In the case of the acoustic emissions, we have multiple (up to thousands of) observations for each parameter within a classification group. Here, we will classify each vector of emissions as corresponding to a single composite. After classification of all the observations for a single test, we take the mode of the individual classifications for a final classification on the composite material tested.

We use linear and quadratic discriminant analysis, Mahalanobis distance and k-nearest neighbor classifiers for the AE data set, which are explained below. Results of these classifiers follow the descriptions of the methods.

5.1 Linear Discriminant Analysis

The reasonably robust linear discriminant analysis (LDA) classifier requires a Gaussian assumption on the data. Even when this assumption does not hold true for the data, oftentimes the desired result can still be achieved [10]. The derivation for the LDA classifier follows from the work set forth in [8, 16, 10], where further information on these classifiers can also be found.

Let ω_i represent the i^{th} of the n classes. When considering to which class a given data vector $\boldsymbol{x}$ belongs, we denote the conditional probability of $\boldsymbol{x}$ belonging to ω_i by $p(\omega_i|\boldsymbol{x})$, for $i = 1, \ldots, n$.. We are interested in the class that attains the greatest probability given data $\boldsymbol{x}$. Thus classification will yield $\boldsymbol{x} \in \omega_i$ when

$$p(\omega_i|\boldsymbol{x}) > p(\omega_j|\boldsymbol{x}) \quad \text{for all} \quad j{\neq}i. \tag{15}$$

We define the probability that a data vector $\boldsymbol{x}$ belongs to a given class ω_i by $p(\boldsymbol{x}|\omega_i)$ and use Bayes' theorem to relate the two conditional probabilities:

$$p(\omega_i|\boldsymbol{x}) = \frac{p(\boldsymbol{x}|\omega_i)p(\omega_i)}{p(\boldsymbol{x})}, \tag{16}$$

with $p(\omega_i)$ the probability the i^{th} class occurs in the data and $p(\boldsymbol{x})$ the probability of finding a data vector from any of the classes at the location $\boldsymbol{x}$. In Bayes' terminology, $p(\omega_i)$ is referred to as the prior probability and is used to model a priori knowledge.

Substituting (16) into (15) the method will classify $\boldsymbol{x}$ as an element of class ω_i when

$$p(\boldsymbol{x}|\omega_i)p(\omega_i) > p(\boldsymbol{x}|\omega_j)p(\omega_j) \qquad \text{for all} \quad j{\neq}i. \tag{17}$$

The probability $p(\boldsymbol{x}|\omega_i)$ can be determined using training data supplied by the analyst, while $p(\omega_i)$ is determined using prior knowledge of the data. Here we assume a particular vector $\boldsymbol{x}$ has the same probability of being classified into any of the classes. However, if we have reason to believe that a data vector is more likely to be in one class than another (i.e. more likely to be a control than a defect), then different priors can be assigned. To simplify the above equation, we use the discriminant functions $g_i(\boldsymbol{x})$ defined to be the

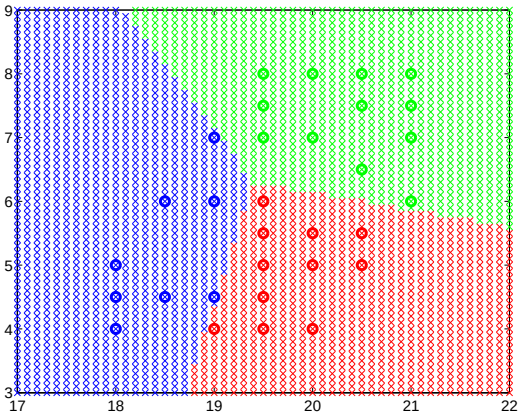


Figure 9: A contrived training set used to classify $\mathbb{R}^2$ in order to visually demonstrate the behavior of LDA. Three arbitrary classes are represented by the colors red, green and blue. Training data appear as circles while test data appear as 'x' marks.

natural logarithm of the left hand side of (17) for any i, which preserves the inequality since the logarithm is a monotone, increasing function:

$$g_i(\boldsymbol{x}) = \ln p(\boldsymbol{x}|\omega_i) + \ln p(\omega_i). \tag{18}$$

For simplification, we assume the probability distributions for the classes are multivariate normal, though this is not necessarily verifiable. We make this assumption since the properties of the multivariate normal are well-known, In practice, [10] states that classes are not generally normally distributed even though it is assumed so, nor are the class means and data covariances known, though they can be calculated using training data.

For the LDA classifier, it is assumed that the covariance matrices between classes, Σ_i, are equal, i.e. $\Sigma_i = \Sigma$ for all i. Thus we have

$$p(\boldsymbol{x}|\omega_i) = \frac{1}{\sqrt{(2\pi)^N |\Sigma|}} e^{-\frac{1}{2}(\boldsymbol{x}-\boldsymbol{m}_i)'\Sigma^{-1}(\boldsymbol{x}-\boldsymbol{m}_i)}, \tag{19}$$

where N is the dimension of data vector $\boldsymbol{x}$, $\boldsymbol{m}_i$ is the mean vector of data in class ω_i and Σ is the covariance matrix of all the data. Then the discriminant function has the form

$$g_i(\boldsymbol{x}) = -\frac{N}{2}\ln 2\pi - \frac{1}{2}\ln|\Sigma| - \frac{1}{2}(\boldsymbol{x}-\boldsymbol{m}_i)'\Sigma^{-1}(\boldsymbol{x}-\boldsymbol{m}_i) + \ln p(\omega_i). \tag{20}$$

Finally, we can simplify (20) by removing terms that are independent of i. Recall also the assumption that $\ln p(\omega_i)$ is equal for all i, and thus can be removed. Moreover, we equivalently use the discriminant function

$$g_i(\boldsymbol{x}) = \boldsymbol{m}_i'\Sigma^{-1}(\boldsymbol{x} - \frac{1}{2}\boldsymbol{m}_i). \tag{21}$$

Thus linear discriminant analysis classifies $\boldsymbol{x}$ as an element of ω_i when

$$g_i(\boldsymbol{x}) > g_j(\boldsymbol{x}) \qquad \text{for all} \quad j \neq i, \tag{22}$$

for the discriminant function given in (21).

For a visual example of the LDA classifier, a contrived training set was created in $\mathbb{R}^2$ with three groups (not actual control/defect data). Figure 9 shows the training set (red, green, and blue circles), along with the classification of $\mathbb{R}^2$ (red, green and blue 'x' marks). Note that the LDA creates linear boundaries.

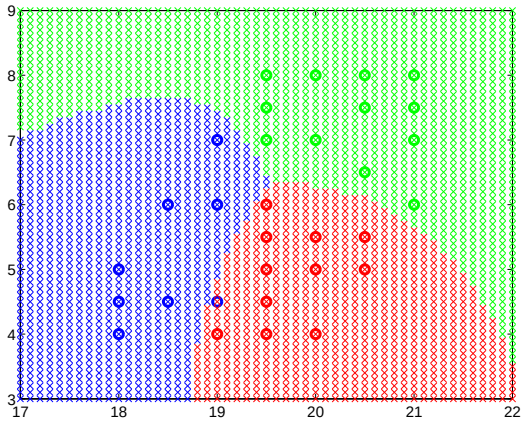


Figure 10: A contrived training set used to classify $\mathbb{R}^2$ in order to visually demonstrate the behavior of QDA. Three arbitrary classes are represented by the colors red, green and blue. Training data appear as circles while testing data appear as 'x' marks.

5.2 Quadratic Discriminant Analysis

Quadratic discriminant analysis (QDA) is a simple method and popular amongst classifiers for its simplicity and compact training process [14]. This method also assumes a multivariate Gaussian model on the data, however the class covariances are not assumed to be equal. QDA uses the discriminant function

$$g_i(\boldsymbol{x}) = -\ln|\Sigma_i| - (\boldsymbol{x}-\boldsymbol{m}_i)'\Sigma_i^{-1}(\boldsymbol{x}-\boldsymbol{m}_i) \tag{23}$$

to determine classification of the data $\boldsymbol{x}$ using (22).

A visual example of the QDA classifier is given in Figure 10 for the same contrived data set shown above under LDA. Notice that the difference between Figures 9 and 10 lies within the boundaries created by the classifiers. QDA has quadratic boundaries that allow for nonlinear separation within the data set parameter space, but in this example, misclassifies one of the training data points.

5.3 Mahalanobis Distance

Let us again reconsider the discriminant function in (20). As with LDA, we assume equal covariances across classes and examine only the third term because the other terms are identical on both sides of the discriminant inequality in (22). By reversing the sign, we can regard this equation as a distance squared measure. Classification is now based on (22) where

$$g_i(\boldsymbol{x}) = -\sqrt{(\boldsymbol{x}-\boldsymbol{m}_i)'\Sigma^{-1}(\boldsymbol{x}-\boldsymbol{m}_i)}. \tag{24}$$

Just as with LDA and QDA, we provide a visual example of the Mahalanobis classifier on the example from above. Figure 11 shows the same training set (red, green, and blue circles), along with the Mahalanobis distance classification of $\mathbb{R}^2$ (red, green and blue 'x' marks). Note that Mahalanobis distance classifier is very similar to the QDA classification in this example, but upon close inspection, the classification of $\mathbb{R}^2$ is not the same. Notice for this example, just as with QDA, there is a training data point misclassified by this method.

5.4 k-Nearest Neighbor

The k-Nearest Neighbor (kNN) procedure is a relatively simple method used to classify data sets [8], and can be used with categorical classifications like our data set. The kNN procedure is premised upon the existence of a training set composed of data points and their classifications, and thus is a type of memory-based classifier.

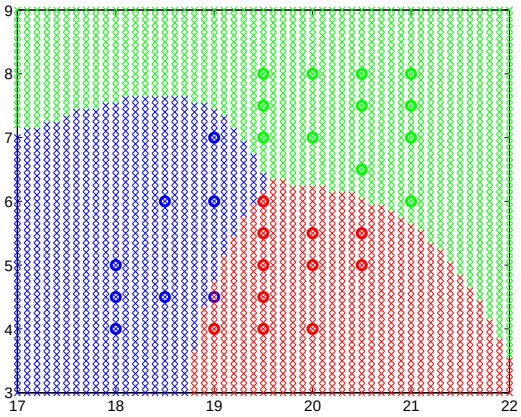


Figure 11: A contrived training set used to classify $\mathbb{R}^2$ in order to visually demonstrate the behavior of the Mahalanobis distance classifier. Three arbitrary classes are represented by the colors red, green and blue. Training data appear as circles while test data appear as 'x' marks.

In order to classify a new unclassified data point, the algorithm compares the distance between the new data point and the k nearest points in the training data set, referred to as neighbors. The new point is classified based on some criterion, usually the class that the most nearest neighbors are in, with a procedure for tie-breaking. Larger values of k incorporate more neighbors into the classification, reducing the impact of noise, but can then make it more likely that neighbors of a test point will lie in multiple classes making decisions more difficult. In the special case of $k = 1$, the algorithm simply classifies a new point as being in the same category as its single nearest neighbor; this is also known as the Voronoi tessellation of the training data.

The procedure does not make any assumptions about the underlying data structure, other than the hoped-for separation of the data into clusters. The downside is a relatively large computational cost to evaluate the classification of a new point, as computing distance to training points is costly particularly as the number of data parameters rises.

Short intuitive kNN example on Fisher iris data

We will use the Fisher iris data [5] to visually demonstrate the performance of the k-Nearest Neighbor method. This data set is based on data collected in the 1930's on various properties of iris flowers; here we use only two properties, petal length and petal width. In Matlab, this data is stored internally and loaded by using the command `load fisheriris`. The example shown here is based heavily off the Matlab manual page "Examine the Quality of a KNN Classifier." In Figure 12, we show a visual representation of the data.

For demonstration purposes, we will determine the class of a new test flower which has petal length 5, and petal width 1.45. Using the Matlab routine `knnsearch`, we find the ten nearest neighbors to our test point (exhaustive search using Euclidean distance, for this test case); the results of this process are shown in Figure 13, where the test point is near the middle of the figure and marked with an 'x'. Note that if we used the nearest neighbor (represented by the point marked by a diamond just above the test point), the test point would be classified as a virginica iris plant. However, if we expand and use the ten nearest neighbors (shown by points covered with circles), the majority of these points are classified as versicolor which would result in a classification of our test point as a versicolor plant.Thus, we see that increasing the k value results in a different classification.

This points out an inherent issue with the kNN classifier, namely the necessity of a proper choice for k. This is not known beforehand, and so one hopes that classifying test data will lead to determining an adequate value for k for a particular application. In this example, we can see that values for k between 5 and 10 would all classify the test flower as versicolor, meaning we would see some robustness in the classifier with respect to k. As we will later see, this is not the case for our AE data.

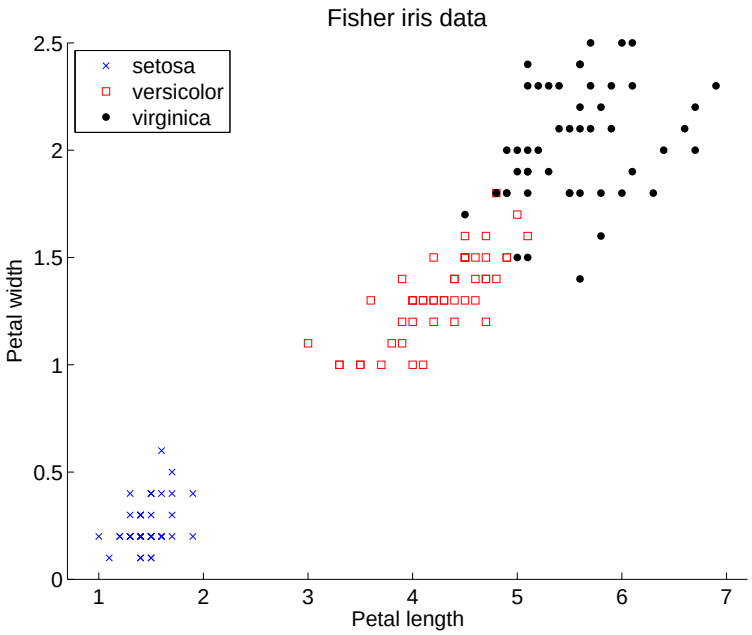


Figure 12: Fisher iris data. Three types of flowers are represented graphically, and their petal length and petal width are used to mark each flower data point.

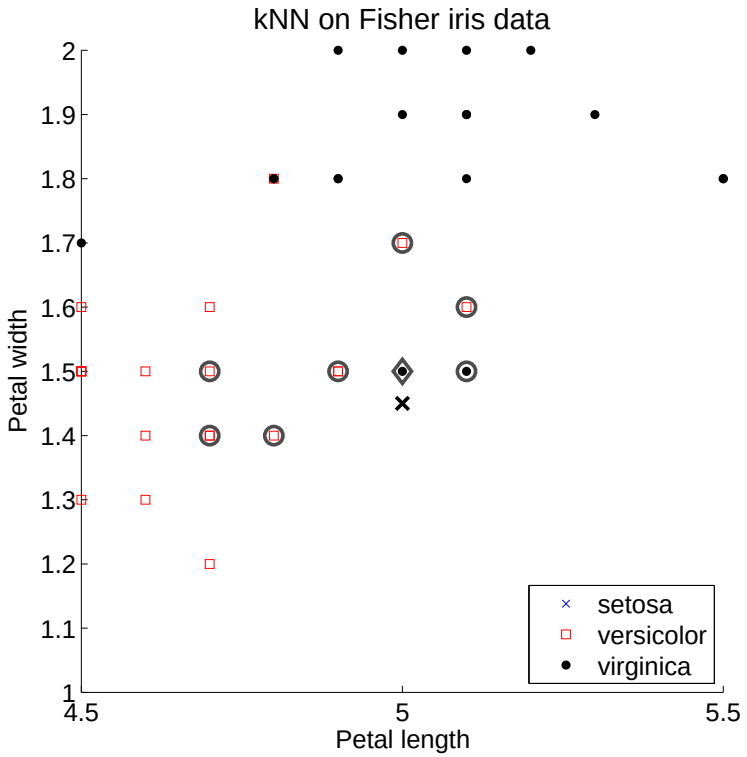


Figure 13: Fisher iris data, zoomed near test point (5,1.45) which is marked with an 'x'. The nearest neighbor is represented by the point marked by a grey triangle, and the next 9 nearest neighbors are shown as the points with grey circles around them.

Method	Error rate	
	θ_1	θ_2
LDA	0.6667	0.6667
QDA	0.6667	0.5667
Mahalanobis	0.4000	0.4000

Table 23: Error rate for the full AE data set as classified into three groups (control, defect 1 or defect 2) in Matlab using the command `classify` under LDA, QDA, and Mahalanobis classifiers. All three blocks were used. The parameter sets used for prediction were $\theta_1 = \{2, 3, 5, 9, 10\}$ and $\theta_2 = \{2, 3, 4, 5, 8, 9, 10, 11\}$.

Method	Error rate	
	θ_1	θ_2
LDA	0.3000	0.3667
QDA	0.6667	0.6667
Mahalanobis	0.3667	0.4333

Table 24: Error rate for the third block of the AE data set as classified into three groups (control, defect 1 and defect 2) in Matlab using the command `classify` under LDA, QDA, and Mahalanobis classifiers. The parameter sets used for prediction were $\theta_1 = \{2, 3, 5, 9, 10\}$ and $\theta_2 = \{2, 3, 4, 5, 8, 9, 10, 11\}$.

5.5 Classifier method results on AE parameter data

We now turn to the results of our classifier methods. Due to time constraints, unless otherwise specified, the training data set was composed of the first ten data sets in each of the control, defect 1, and defect 2 groups. The testing data sets were the remaining ten of each group. Recall that we are examining the AE data from both of the parameter sets $\theta_1 = \{2, 3, 5, 8, 10\}$ and $\theta_2 = \{2, 3, 4, 5, 8, 9, 10, 11\}$ previously selected in Section 4.7. In general, each AE signal was classified separately, and then the mode was taken for each AE signal classification corresponding with the control, defect 1, or defect 2 data set being classified.

5.5.1 Results: LDA, QDA, Mahalanobis

LDA, QDA and Mahalanobis classifiers were applied to the acoustic emissions data set using the Matlab command `classify`. This command requires the training set with group labels and the test set as inputs, while returning the classification for each entry in the test set.

The classifier was applied to the full set of data (all three blocks together), as well as just the third block. We discussed above the merit of using fewer than the thirteen parameters supplied in the data set. The correlation coefficients reduced the parameter set to $\theta_2 = \{2, 3, 4, 5, 8, 9, 10, 11\}$. When combined with the other methods discussed above, we chose the core set of parameters to be $\theta_1 = \{2, 3, 5, 9, 10\}$. Both will be used in the analysis.

To determine an error rate post-classification, we took the error as the proportion of observations misclassified by the classifier against the truth. For example, of the thirty training composites, if two controls were misclassified as defect 1 and three defect 1 composites were misclassified as defect 2, with the remaining tests all classified correctly, the error rate would be equal to 5/30, or 0.16667. Results are given in Table 23 for classification under LDA, QDA and Mahalanobis distance, using all three blocks of data and both parameter sets selected in Section 4.7.

These are high error rates, thus we hypothesized that the third block of AE may be the most important regarding predictive power. We then applied the classifiers to the third block on its own. Results are given in Table 24, though it is clear that this was not necessarily a large improvement over the error rate when using all three blocks together.

Furthermore, when looking at the classification for the three groups, it was noticed that the control group

Method	Error rate	
	θ_1	θ_2
LDA	0.0667	0.0667
QDA	0.6667	0.6667
Mahalanobis	0.2333	0.0333

Table 25: Error rate for the full AE data set as classified into two groups (control or defect) in Matlab using the command `classify` under LDA, QDA, and Mahalanobis classifiers. The parameter sets used for prediction were $\theta_1 = \{2, 3, 5, 9, 10\}$ and $\theta_2 = \{2, 3, 4, 5, 8, 9, 10, 11\}$.

Method	Error rate	
	θ_1	θ_2
LDA	0.0333	0.1000
QDA	0.3333	0.0667
Mahalanobis	0.0667	0.0333

Table 26: Error rate for the third block of the AE data set as classified into two groups (control or defect) in Matlab using the command `classify` under LDA, QDA, and Mahalanobis classifiers. The parameter sets used for prediction were $\theta_1 = \{2, 3, 5, 9, 10\}$ and $\theta_2 = \{2, 3, 4, 5, 8, 9, 10, 11\}$.

was generally classified accurately but the classifier had difficulty differentiating between the two defect groups. By combining the defects into a single group and classifying whether a composite is a control or defect, the results generally showed error rate improvement, regardless of the data set used (full or third block only). Results for the combined defect groups are found in Tables 25 and 26 for both parameter sets and both data sets.

In the results for the combined defect groups, we notice that all three methods can perform well depending on the parameter/data set combination. QDA performs the poorest overall. The Mahalanobis distance returns the lowest misclassification rate of 0.0333 when using the parameter set garnered from the correlation coefficients regardless of the data set. LDA can match this error rate when the core parameter set, θ_1, is used to classify the composites into either the control or defect group. The only composites misclassified in the tests in Tables 25 and 26 are control composites; defects are not being classified as controls. In practice, it is safer to misclassify a control as a defect, rather than vice versa.

5.5.2 Results: k-Nearest Neighbor

In order to apply kNN to the acoustic emission data set, we split the data set into training and testing sets. Only the data in block 3 was used in the training and testing data sets, since block 3 data contains the most information. The training set was the first 10 data sets for each of the control, defect 1, and defect 2 data; testing sets were the remaining data, for a total of thirty each of training and test data sets. Each entry in the training set was tagged with its class, 0 for control, 1 for defect 1, and 2 for defect 2. We utilized the `ClassificationKNN` structure that was implemented in the most recent Matlab version 7.14. This class takes the training data and corresponding labels as input, has a `predict` subroutine that takes in new data and computes the kNN results using a specified distance metric and k value. By default, the distance metric is the standard Euclidean distance and $k = 1$. These are both parameters that can be dynamically changed in the classification structure. The parameters used here are the core set, $\theta_1 = \{2, 3, 5, 9, 10\}$, as well as the set obtained just by eliminating correlated variables, $\theta_2 = \{2, 3, 4, 5, 8, 9, 10, 11\}$. The Euclidean distance for two general vectors $\vec{p}$, $\vec{q}$, of equal length n is then defined by

$$d(\vec{p}, \vec{q}) = \sqrt{(\vec{q} - \vec{p})^T (\vec{q} - \vec{p})}.$$

Recall that the Mahalanobis distance is similar to the Euclidean distance, but takes into account correlations between parameters by incorporating the covariance matrix $\Sigma = Cov(\vec{p}, \vec{q})$, and is defined by

$$d(\vec{p}, \vec{q}) = \sqrt{(\vec{q} - \vec{p})^T \Sigma^{-1} (\vec{q} - \vec{p})}.$$

Since Mahalanobis was known to be somewhat successful in the discriminant analysis tests, we again will try Mahalanobis along with Euclidean distance metrics when building the kNN classifier. By default Matlab uses the covariance matrix as previously defined. If additional information arose which indicated particular AE parameters were relatively more or less important, Matlab does allow for the specification of a different weight matrix.

The testing procedure involved classifying each acoustic emission individually for a particular test data set. Once each data point for that set was classified, the classification with a plurality (i.e., the mode of the AE classifications) was assigned to the entire test. This mimics our other classification procedures, and is again logical as a procedure since the data for each test are connected and thus must be considered as a group for classification purposes.

Put simply, kNN using AE parameters to delineate between control, defect 1, and defect 2 was unsuccessful. In most instances, all of the test data sets were classified as defect 2. This occurred for $k = 1, 2, 3$, and 4. Similarly, we tried using only two classes, control and defect. Even when combining the defect types together into a single category and attempting to classify control versus defect, the kNN method classified all test sets as defective.

This poor performance may be due to the fact that the number of acoustic emissions are quite small for the control parts, a moderate amount for defect 1, and a large number for defect 2. In fact, on average defect 2 composites give off nearly twice as many acoustic emissions as defect 1 and two orders of magnitude more than the control group. Since the kNN procedure compares the individual acoustic emissions in each of the test data sets, it is conceivable that each point may be nearer a defective trial point than a control trial point; similarly, being near a defect 2 point is twice as likely as a defect 1 point given the larger number of defect 2 acoustic emissions in the training set. By classifying each AE event as a separate point, we seem to lose information that ties *groups* of acoustic emissions from the control sets to each other. Perhaps other classification methods could be implemented, but for our current purposes we consider the kNN methodology on acoustic emission parameters to be unsuccessful.

5.5.3 Summary of results on AE parameter-based classification

In general, using LDA, QDA, and MD, we were able to differentiate between control and defective composites. Determining the difference between defect 1 composites and defect 2 composites has proved elusive. The results here should provide a good starting point for further study on classification using AE parameters.

5.6 Classifier method results using the number of emissions

The number of total emissions from the control composites were much lower than that of either of the defects, and defect two tended to have a larger number of total emissions than defect one. To visually determine if using the number of total emissions would be an avenue worth exploring by means of classification, we separated each test into its three blocks and totaled the number of emissions. The total number of emissions per test are displayed in Table 27 for each group, and a 3-D plot of this data is found in Figure 14. Classification using the total number of emissions was performed using LDA, QDA, Mahalanobis distance and kNN classifiers.

Various training and test sets were used to gain an idea of the sensitivity of the classifier to different training sets. Recall that for each group there are 20 tests. Using the same sets from each group, we tested five different training sets in order to test robustness to the training set. The various additional training set combinations were: 11-20, 1-5/16-20, odds, and evens (with the test sets being the remaining data sets).

5.6.1 Results: LDA, QDA, Mahalanobis

Results for the classification of these five training sets using all three blocks with either 2 groups (control, defect) or 3 groups (control, defect 1, defect 2) are given in Table 28.

Test	1	2	3	4	5	6	7	8	9	10
Block 1	4	29	28	2	9	12	4	6	48	16
Block 2	2	8	10	4	5	11	5	12	42	20
Block 3	9	59	75	44	16	83	31	34	76	22

Test	11	12	13	14	15	16	17	18	19	20
Block 1	6	8	15	19	6	25	12	35	53	19
Block 2	19	4	12	19	5	7	6	33	14	9
Block 3	35	11	58	56	35	18	33	64	22	28

(a) Control Group

Test	1	2	3	4	5	6	7	8	9	10
Block 1	11	21	6	10	22	26	38	32	24	19
Block 2	5	78	26	11	147	24	45	35	14	15
Block 3	1839	1206	1097	1703	1746	1569	751	706	191	1320

Test	11	12	13	14	15	16	17	18	19	20
Block 1	43	29	107	13	9	21	10	5	7	9
Block 2	46	23	21	88	70	2	32	62	17	54
Block 3	1405	2624	1302	1468	1062	1382	2161	437	260	1039

(b) Defect 1 Group

Test	1	2	3	4	5	6	7	8	9	10
Block 1	72	62	49	143	42	130	201	172	125	31
Block 2	43	115	437	62	44	67	57	398	95	36
Block 3	2621	3269	3162	3872	2735	3221	2372	2050	3267	2359

Test	11	12	13	14	15	16	17	18	19	20
Block 1	80	33	126	89	43	20	44	37	54	119
Block 2	155	21	52	307	114	66	102	27	46	40
Block 3	2816	2639	3694	2414	1703	2273	3816	2740	3087	3154

(c) Defect 2 Group

Table 27: Total number of acoustic emissions of all 20 tests observed for each block in each of the three groups: control, defect 1, and defect 2.

While it appears that LDA performed satisfactorily with low error rate in all trials, regardless of the number of groups considered, there was at least one defect composite misclassified as a control in each trial. As mentioned above, this is an undesireable type of misclassification and should be taken into consideration with the error rate. When determining if a composite is a control or defect, it appears that QDA performs best with lowest error rate, is not as sensitive to the training set as other methods, and when a misclassification occurred, it was a control composite misclassified as a defect.

When considering classification into all three groups, it appears QDA again performs the best of the three methods. In this case, QDA did not misclassify a defect as a control, but it did occasionally label a control as defective and/or mix up the two defective groups.

5.6.2 Results: k-Nearest Neighbor

In the first set of tests, we input each classified acoustic emission from the training set individually into the Matlab kNN structure. This creates a large set of points (in $\mathbb{R}^5$ for the core set or $\mathbb{R}^8$ for θ_2). The testing

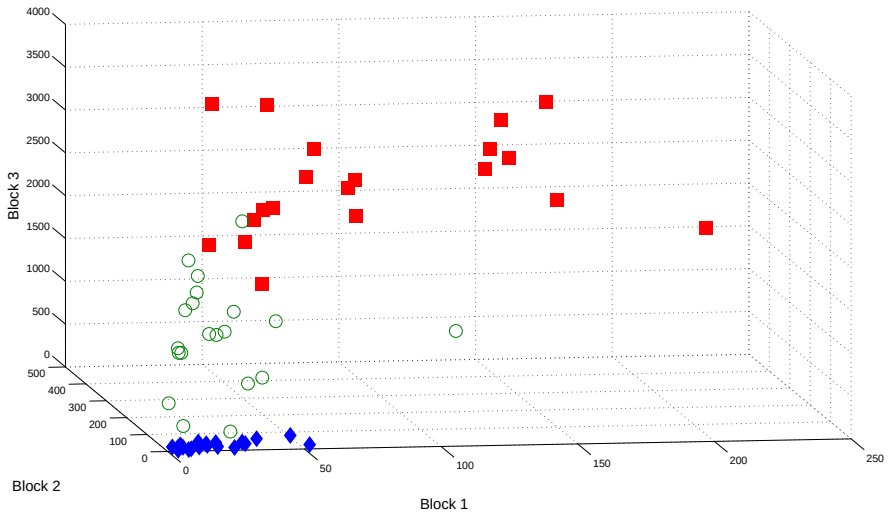


Figure 14: A 3-D plot of the number of total emissions per block per group. The control data is represented by blue diamonds, defect 1 by green circles, and defect 2 by red squares. Since the types of data are grouped separately using the eye metric, this plot demonstrates intuitively the possibility of data separation using total number of emissions per block as parameters.

Number of groups	Method	Error rate in tests used for training sets				
		First 10	Middle 10	Last 10	Even	Odd
3	LDA	0.1667	0.0667	0.2000	0.1667	0.1000
	QDA	0.1667	0.0667	0.1000	0.0667	0.1000
	Mahalanobis	0.1333	0.2000	0.1667	0.1000	0.2000
2	LDA	0.1000	0.2000	0.1333	0.1667	0.0667
	QDA	0.0333	0.0000	0.0000	0.0000	0.0000
	Mahalanobis	0.1000	0.1667	0.1000	0.0667	0.0333

Table 28: Classification using the total number of emissions, broken into three blocks. Both 3 (control, defect 1, and defect 2) and 2 (control/defect) groups were considered for classification. Different training sets were considered to test for classifier sensitivity by using, out of the twenty available tests per group, the first ten, the middle ten, the last ten, all even tests and all odd tests.

procedure was then followed for all the test sets. We used both the core and correlated parameter sets. Both Euclidean and Mahalanobis distances were tried.

Instead of using the specific acoustic emission parameters, we used the counts of acoustic emissions for each test. This results in a total of 60 data sets, 20 each for control, defect 1, and defect 2. We again use Euclidean and Mahalanobis distance methods, and the values $k = 1, 2, 3$, and 4. Results shown in Tables 29-30 use the first 10 data sets in each group as the training data.

	11	12	13	14	15	16	17	18	19	20
Control	0	0	0	0	0	0	0	0	0	0
Defect 1	1	2	1	1	1	1	2	1	1	1
Defect 2	2	2	2	2	1	2	2	2	2	2

Table 29: Classifications on the test set using total AE counts for test data coming from the control, defect 1, and defect 2 groups, using Euclidean distance. Test set identification numbers go across the top row, and the true data set classification is listed in the left column. A 0 in a slot means that data set was classified as a control set; 1 was classification as a defect 1 part; and 2 was classification as a defect 2 part. For example, data set 14 from defect 1 was properly classified as defect 1, while data set 12 from defect 1 was misclassified as defect 2.

	11	12	13	14	15	16	17	18	19	20
Control	0	0	0	0	0	0	0	0	0	1
Defect 1	1	2	0	1	1	1	1	0	0	1
Defect 2	2	2	2	1	1	2	2	2	2	2

Table 30: Classifications on the test set using total AE counts for test data coming from the control, defect 1, and defect 2 groups, using Mahalanobis distance. Test set identification numbers go across the top row, and the true data set classification is listed in the left column. A 0 in a slot means that data set was classified as a control set; 1 was classification as a defect 1 part; and 2 was classification as a defect 2 part. For example, data set 14 from defect 1 was properly classified as defect 1, while data set 12 from defect 1 was misclassified as defect 2.

As can be seen in Table 29, 100% of the control test set was classified properly. All the defective sets were classified as defects, though one can see that certain data sets were misclassified in terms of which defect they represented. In Table 30, we see that the nearest neighbor algorithm with the Mahalanobis distance has significantly worse performance, in the sense that several defect 1 pieces were misclassified as control pieces. Though the tables are not included here, for $k = 2, 3$ and 4, the performance decreased. Using both Euclidean and Mahalanobis distances, with these higher k values at least some defect 1 composites were misclassified as controls. In using Euclidean distance was consistently better than using the Mahalanobis distance.

As discussed before, we used different training data sets to test the robustness of the classifiers. The results were the same for the additional training set combinations, except for training on the even data set numbers. In the case of training on even test numbers, for $k = 1$ and 2 two defect 1 data sets out of 10 were misclassified as controls, when using both the Euclidean and Mahalanobis distance options.

It seems that kNN using acoustic emission counts is a relatively successful procedure, but there is still some risk of a defective part being misclassified as a control part. In the data sets studied, all control composites were correctly classified. The dependence on the tuning parameter k is another source of discomfort, in that we have only seen good results for $k = 1$. We would be more comfortable if multiple k values were successful. Since we have other methods that do not risk missing a defective part and do not incorporate a user-defined tuning parameter, it is our recommendation to not use kNN methods unless in conjunction with another classification method or with larger training sets.

5.6.3 Summary of results on total AE-based classification

Using total acoustic emissions has proven to be a more consistently successful methodology for differentiating between control parts and defective parts. Success rates near 100% are possible. This makes intuitive sense, as

the defective parts are known to result in larger numbers of acoustic emission events; we have quantitatively confirmed this intuition. We were also able to find methods (QDA and kNN) which could often differentiate between the two types of defective parts. This is an improvement on the results from the previous section which used the AE parameters individually.

6 Break and Deformation Load Estimation

Now that we have various techniques at our disposal to determine if the composites is safe or defective, we turn to attempting to estimate the break load based on the acoustic emission signatures. Additionally, defective pieces experience plastic deformation in block 3. We would also like to examine whether acoustic emission data could be utilized to predict at which load level plastic deformation begins.

6.1 Estimating the plastic deformation load and break load

One of the goals that we tried to achieve was to be able to use an AE test to predict the plastic deformation load and the breaking load of a given sample. A rough indicator that could be used is first identify whether the composite is a control or defect. We then examined the load and AE data to see if we could predict the plastic and break load of the samples. From the experimental data received it is observed that defect 1 and defect 2 have similar plastic deformation loads. However the break load shows a greater variation as shown in table 32. In Figure 15 we illustrate in blue, red and green the experimental load data for the control, defect 1 an defect 2 groups respectively.

To summarize once we are able to identify a sample in either control group or a defective one, the plastic load can be inferred from the data shown in the tables 31 and 32. To some extent we can predict very generally the break load for each type; however, the greater variance in break load makes this method difficult to implement and unlikely to be successful.

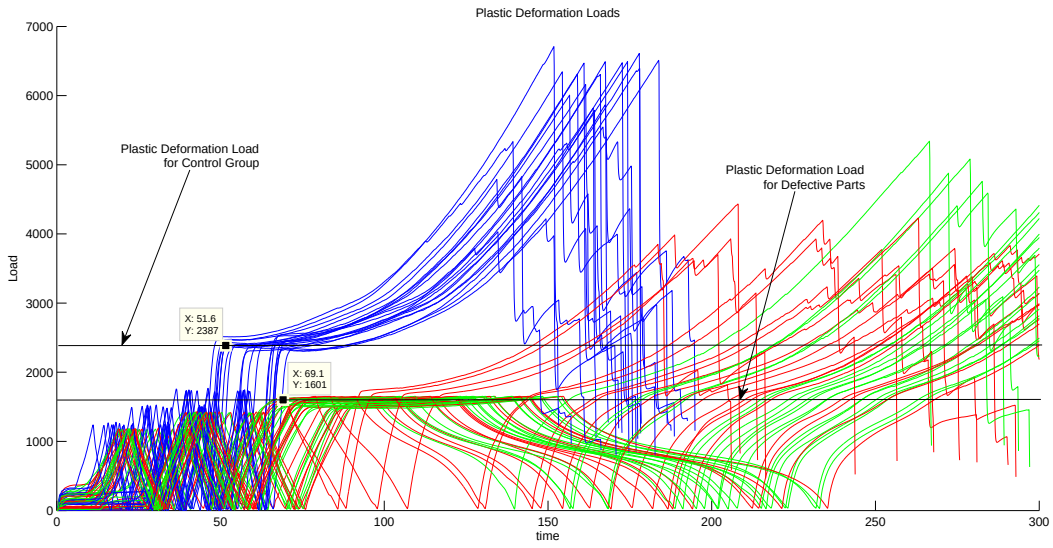


Figure 15: This plot shows that the plastic deformation load for the control group and the defective parts. We can appreciate that the plastic deformation load is very similar for both type of defects.

Table 31: Plastic Deformation Loads

Test	Control	Defect 1	Defect 2
1	2511.607	1659.380	1556.417
2	2286.560	1670.880	1494.949
3	2344.998	1662.166	1476.328
4	2320.693	1657.769	1520.011
5	2332.391	1650.884	1524.552
6	2377.380	1662.926	1540.643
7	2348.077	1663.081	1599.383
8	2324.075	1668.885	1455.657
9	2268.877	1733.062	1586.930
10	2354.190	1659.936	1508.331
11	2395.568	1661.603	1531.051
12	2439.318	1665.221	1505.990
13	2380.266	1663.293	1540.687
14	2492.415	1669.014	1515.458
15	2416.389	1672.434	1521.585
16	2421.562	1670.725	1472.048
17	2372.620	1665.094	1502.810
18	2380.824	1650.351	1571.482
19	2426.905	1665.718	1564.248
20	2462.590	1668.371	1602.866
minimum	2268.877	1650.351	1455.657
maximum	2511.607	1733.062	1602.866
mean	2382.87	1667.04	1529.57
STD	64.31	16.65	41.28

6.2 Using AE to detect damage

Now we describe briefly one approach that uses AE to detect major damage suffer by a piece during the tests. The idea can be summarized as follows.

- We take the cumulative sum of the parameter 2 acoustic emissions in block three.
- We scale the time for block 3 by setting the starting time of the block equal to zero and the final time equal to one houndred.
- We take the time derivative of the cumulative parameter 2, with respect to the scaled time.
- Now we define the **detection time** as the point where parameter 2 reaches one tenth of of its peak value. This is going to be a value between 0 and 100 since we use the scaled time.
- Finally we plot the detection times for all the tests and all groups. (See Figure 16)

From the experimental information we set a detection scaled time threshold at 29 seconds and consider all pieces with smaller detection time as pieces that must be analyzed for structural damage.

Note that for the defect 1 group tests 9, 18 and 19 had detection times above the threshold. We observe the load curves for this tests (an example for test 9 is shown in Figure 18) we can see that the pieces did not suffer extended plastic deformation. This confirms that the criteria outlined could be used to identify the amount of damage to composites, even if it can not be used for a classification.

Table 32: Break Load

Test	Control	Defect 1	Defect 2
1	6490.9	3278.9	4956.7
2	6305.5	3772.9	4583.7
3	6510.3	3268.7	4526.4
4	5544.6	3892.9	4916.9
5	6475.3	3392.5	5591.2
6	6469.2	3504.8	4395.9
7	6313.1	3927.8	5341.7
8	6161.1	3831.5	4430.0
9	5867.4	3849.9	5343.5
10	6492.0	3789.8	4131.9
11	6007.1	3613.1	4591.5
12	6344.9	3228.4	4949.4
13	6610.7	3928.3	4759.2
14	5786.8	3668.0	5275.6
15	6708.5	3598.6	5082.1
16	5819.4	3557.7	4799.5
17	4827.4	4197.7	4573.5
18	4218.1	4432.2	5083.4
19	6392.5	3700.3	5354.8
20	5334.8	4231.9	4878.0
minimum	4218.1	3228.4	4131.9
maximum	6708.5	4432.2	5591.2
mean	6033.98	3733.30	4878.25
STD	641.85	324.15	385.09

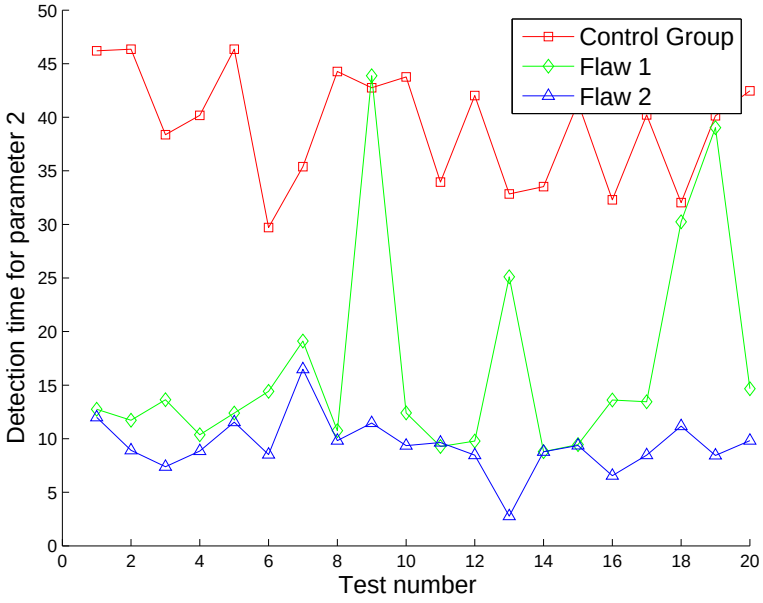


Figure 16: Detection time for cycle 3, for Parameter 2.

Table 33: Detection times for cycle 3

Test	Control Group	Defect 1	Defect 2
1	46.2	12.8	12
2	46.4	11.7	8.9
3	38.4	13.6	7.4
4	40.2	10.4	8.9
5	46.3	12.4	11.5
6	29.7	14.4	8.5
7	35.4	19.1	16.5
8	44.3	10.7	9.8
9	42.7	43.9	11.5
10	43.8	12.4	9.4
11	33.9	9.3	9.6
12	42	9.8	8.5
13	32.8	25.1	2.8
14	33.5	8.8	8.8
15	41.4	9.4	9.4
16	32.3	13.6	6.5
17	40.2	13.5	8.5
18	32	30.2	11.2
19	40.1	39	8.4
20	42.5	14.7	9.8

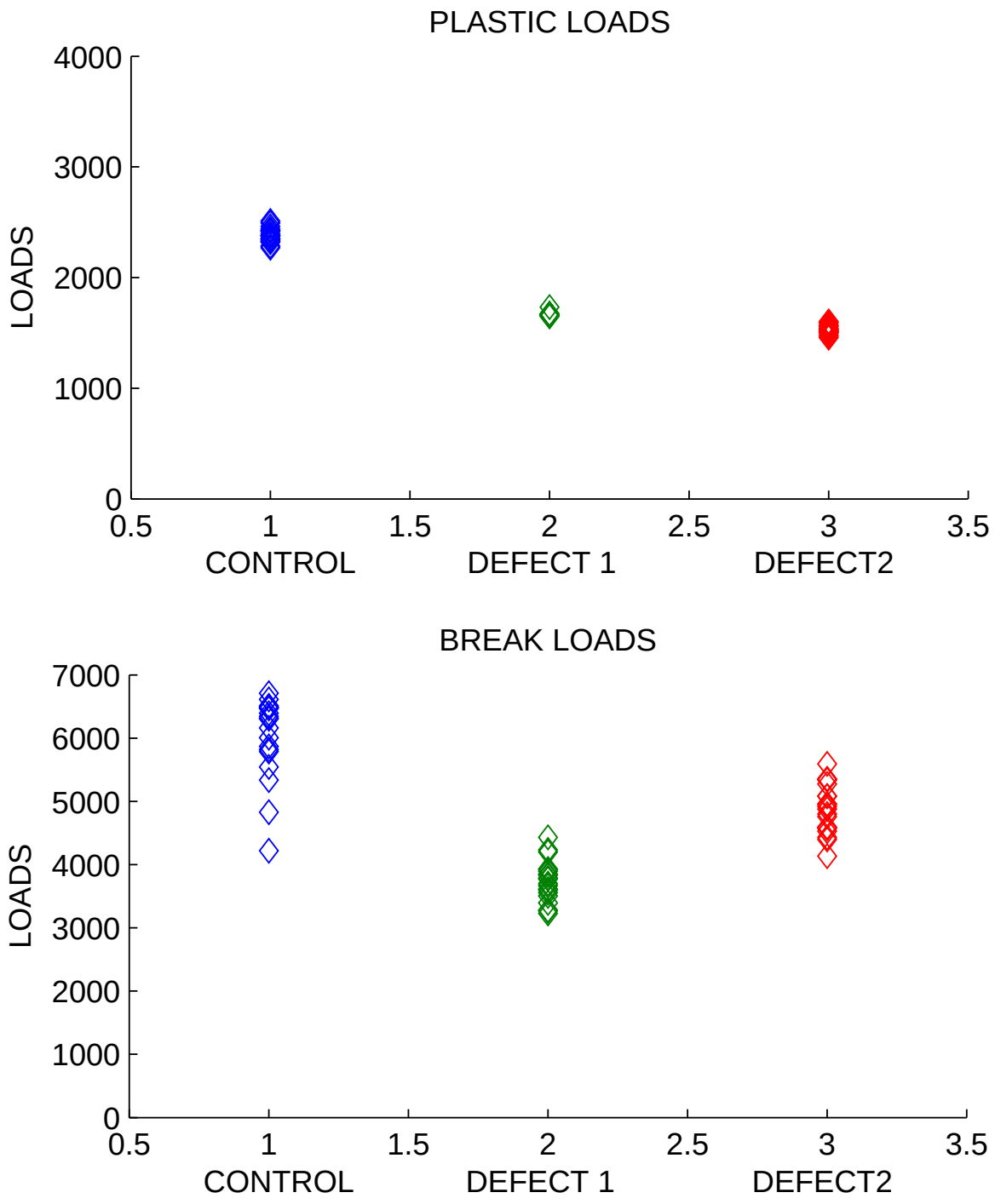


Figure 17: This figure shows a scatter plot of the plastic and break loads of the control group and defective groups.

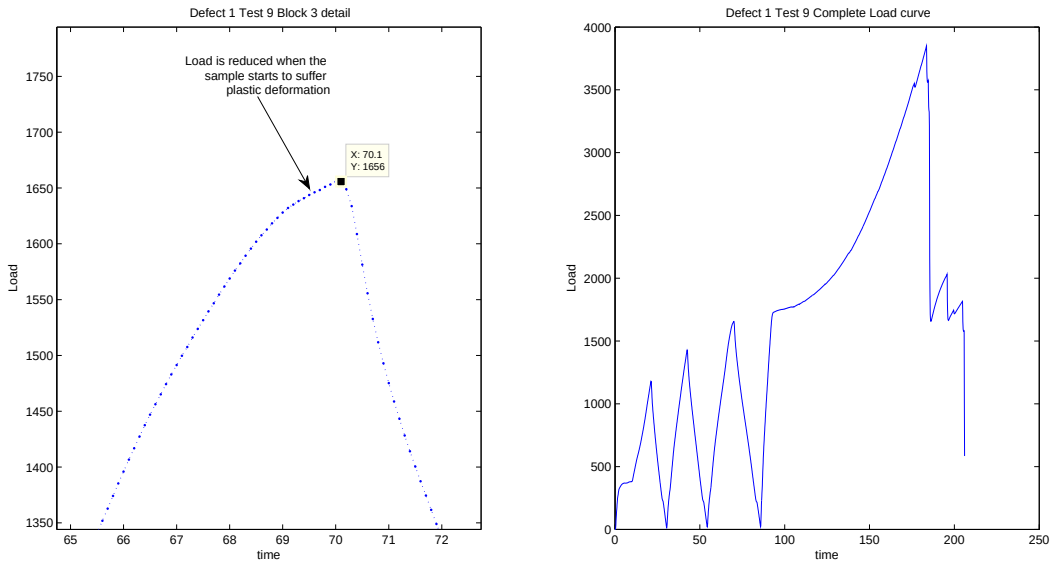


Figure 18: For defect 1 test 9 this figure shows a detail where we can observe that the sample did not suffer extended damage during the third load cycle. The observation is in agreement with the criteria used to detect damaged pieces.

7 Future Work

A project of this magnitude naturally leads to a plethora of ideas for examination. Since this workshop lasted only ten days, we have further ideas that we would try if we had the time. These could be natural extensions of our work for LORD Corporation as they continue to study acoustic emissions of their composite materials, or a basis for further collaboration with LORD.

- For the AE data set, the first ten tests of each group were used as training, due to time constraints. In practice, we would randomly select training and test sets, and further exploration of this should be considered.
- If one wished to examine kNN methods further, trying other distance methods would be the logical next step. Matlab allows for over ten different metrics. There are also options for using classification trees instead of an exhaustive search. Given time constraints, we did not examine these ideas, though they could improve the performance of kNN.
- We ran out of time to explore models, such as linear regression, for predicting the load at which different composites experience plastic deformation and the load at which the composites break. Many models exist for predicting quantitative variables, and with the success we had in using acoustic emissions and total number of emissions to predict group type, we think there may be predictive power in those variables to determine these load points of interest.

8 Acknowledgments

This research was conducted during the 18th Annual Industrial Mathematical and Statistical Modeling Workshop for graduate students hosted by the Center for Research in Scientific Computation and the Mathematics Department at North Carolina State University with support for graduate student investigators provided by the National Science Foundation through the Statistical and Applied Mathematical Sciences Institute. The authors are grateful for the advice and conversation with the project presenter, Haris Halilovic, as well as Nicole Downing, both of LORD Corporation. The guidance of our advisors Drs. H.T. Banks and Shuhua Hu has been instrumental in helping us learn to conduct research in a fast, efficient manner during our short Workshop.

References

[1] H. Akaike, A New Look at the Statistical Model Identification, *Automatic Control*, IEEE Transactions 1974.

[2] K.P. Burnham and D.R. Anderson, *Model Selection and Multimodel Inference: A Practical Information Theoretic Approach* (2nd Edition), Springer, New York, 2002.

[3] L.A. Duesing, *Acoustic Emission Testing of Composite Materials,* Proceedings of the Annual Reliability and Maintainability Symposium, Jan. 1989, Atlanta, GA, 1989.

[4] M.H. DeGroot and M.J. Schervish, *Probability and Statistics,* Addison-Wesley Longman, 3rd Edition, 2001.

[5] R.A. Fisher, The Use of Multiple Measurements in Taxonomic Problems, *Annals of Human Genetics*, **7.2** (1936), 179-188.

[6] A. Gelman, J.B. Carlin, H.S.Stern, and D.B. Rubin *Bayes data analysis*, Chapman and Hall/CRC, 2004.

[7] C.J. Geyer, *Generalized Linear Models in R*, Personal notes, 8 December 2003, `www.stat.umn.edu/geyer/5931/mle/glm.pdf`.

[8] T. Hastie, R. Tibshirani, and J. Friedman, *The Elements of Statistical Learning,* New York: Springer-Verlag, 2001.

[9] M. Huang, L. Jiang, P.K. Law, C.R. Brooks, R. Seeley, and D.L. Klarstrom, Using Acoustic Emission in Fatigue and Fracture Materials Research, *JOM*, **50.11** (1998), `http://www.tms.org/pubs/journals/JOM/9811/Huang/Huang-9811.html`.

[10] L.I. Kuncheva, *Combining Pattern Classifiers: Methods and Algorithms,* John Wiley and Sons, Inc., 2004.

[11] M. Kutner, C. Nachtsheim, J. Neter, and W. Wasserman, *Applied Linear Statistical Models* 5th Ed., McGraw-Hill/Irwin, 2004.

[12] J.A. Nelder and R.W.M. Wedderburn, Generalized Linear Models, *Journal of the Royal Statistical Society*, **135.3** (1972), 370-384.

[13] NIST/SEMATECH *e-Handbook of Statistical Methods* http://www.itl.nist.gov/div898/handbook/, April, 2012.

[14] M. Pal and P. Mather, *An assessment of the effectiveness of decision tree methods for land cover classification*, Remote Sensing of the Environment, **86.4** (2003), 554-565.

[15] A.C. Rencher, *Methods of Multivariate Analysis* John Wiley & Sons, Inc., 2002.

[16] J. Richards, *Remote Sensing Digital Analysis, an Introduction,* Berlin Heidelberg, Germany: Springer-Verlag, 2006.

[17] T.M. Roberts, M. Talebzadeh, *Acoustic emission monitoring of fatigue crack propagation,* Journal of Constructional Steel Research **59** (2003) 695-712.

[18] G.D. Ruxton, The Unequal Variance T-test is an Underused Alternative to Student's T-test and the MannCWhitney U Test, *Behavioral Ecology* **17.4** (2006), 688-690.

[19] S.J. Sheather, Density Estimation, *Statist. Sci.* **18.4** (2004), 588-597.

[20] J.W. Tukey, *Exploratory data analysis*, Reading, MA, 1977

[21] B. Wang, *Multivariate Statistical Analysis and Modeling for R Language,* Jinan University Press, 2010.

[22] B.L. Welch, The generalization of "Student's" problem when several different population variances are involved, *Biometrika* **34** (1947), 28-35.

Establishing an Automated Measurement of Allergic Reactivity

Gyuhyeong Goh[1], Nicole Mendoza[2], Oluseyi Odubote[3], Yao Yu[4],

Problem Presenters:
Agustin Calatroni, Jeremy Wildfire, Herman Mitchell
Rho, Inc.
Faculty Mentor:
Emily L. Kang
University of Cincinnati

Abstract

Using form outlines of wheals, an algorithm in R was created to measure wheal size and wheal area. We found that hand measurements of longest diameter and wheal size underestimates the computer-measured values. Also, we found that there is not a strong linear relationship between wheal area and wheal size ,nor between wheal area and longest diameter. Further area of research concerns extending our algorithm to be able to image process wheals with out the aid of outlines.

1 Introduction and Motivation

Understanding allergic reactivity in humans has become a pressing area of research due to a significant increase in respiratory diseases, such as asthma, since 1980 [1]. Using data from the National Health and Nutrition Examination Survey (NHANES) III, which was conducted in the US from 1988 to 1994, [2], estimated that 54.3% of the US population is allergic to at least one allergen as evidenced by Skin-Prick Test (SPT) data. In fact, this statistic may actually underestimate the prevalence of allergy reaction in the US [2]. Furthermore, there is an urgent need to understand why asthma morbidity is especially high for children that live in less-affluent, inner-city communities in the US [3]. Asthma and allergies significantly degrade the quality of life for those affected and hence, improving current methods of analyzing allergic reactivity is an important avenue to address.

1.1 Goals

The first goal of this paper is to address the measurement inaccuracies evident in current manual SPT procedures. The second purpose of this paper is to evaluate the importance of measuring wheal area in addition to measuring wheal size(sometimes also called average diameter).

1.2 Results

In this paper we were able to develop an image-processing algorithm for measuring wheal size and wheal area using the R programming language. We measured the performance of this algorithm by utilizing built-in functions in MatLab to measure wheals.

Furthermore, hand measurements of longest diameter and wheal size underestimate the computer-measured values. With regard to measuring wheal area, we found that there does not seem to be a simple linear relationship between wheal area and wheal size, nor between wheal area and longest diameter. However, the correlations between wheal area and wheal size are almost one, hence whether or not our results are clinically significant is unclear.

[1]University of Connecticut
[2]University of California, Santa Cruz
[3]Georgia Southern University
[4]University of Rochester

2 Methods

2.1 Skin Prick Test Procedures

The SPT and IgE blood tests are the most common methods for testing and measuring allergic reactivity in human subjects, but which method is superior is debatable. [5] In this paper we focus on the wheals that develop from SPT's. In general, SPT's are conducted at a medical facility where a clinician uses a pre-ordered SPT system which includes allergen extracts, trays with which to prepare the extracts, and tools with which to apply the extracts to the skin. Specifically, a clinician often uses a tool that allows for simultaneous application of a test-panel of allergens. An example of such a tool is the ComforTen (Hollister-Stier, Spokane, WA) which can apply a test-panel of ten allergens to the skin.

After a patient has been "pricked" with allergens, clinicians typically wait fifteen minutes for wheals to develop. A positive control, histamine, and a negative control (no allergen) are always included in the test-panel. A wheal should always develop for the positive control, whereas a wheal may or may not develop for the negative control. A patient is considered to have a valid SPT if she developed a wheal size of 3mm or greater for the positive control and a wheal size less than 3mm for the negative control. Furthermore, a patient is considered to have a positive reaction to an allergen if the corresponding wheal has a wheal size 3mm greater than the negative control wheal size.

Now, current methods of measuring wheal size are the main motivation for this paper as they are prone to human error and inaccuracy. Wheal size is typically measured manually, although there has been recent progress towards measuring wheal size with the aid of technology [6, 7]. The manual process first requires a clinician to find and measure the longest diameter of the wheal, using special rulers designed for measuring wheals. The next step is to find the mid-point of the longest diameter, and then measure the perpendicular diameter that goes through the midpoint. Then, the wheal size is calculated by taking the average of the longest diameter and the perpendicular diameter. The clinician records these measurements on a pre-prepared form designed for recording wheal sizes, which we will refer to as forms. A clinician then makes an outline of the wheals on the skin using a dark colored pen. Finally, the clinician transfers the wheal outline to the back of the form using transparent masking tape. See Figure 1(middle, right) in section 2.8. We will refer to these tape-transferred wheal outlines as form outlines.

2.2 Image Processing Methods

As described previously SPT's are prone to inaccuracy due to human error. Furthermore, the precise area of a wheal has rarely been taken into account because it is near impossible to manually calculate the area of a wheal, especially if the wheal is non-circular or non-spherical. Hence, we propose the use of image processing in order to measure wheal area, and to obtain precise measurements for the longest diameter, perpendicular, and wheal size. [7, 8, 9, 10] have also proposed the use of image processing to measure wheals.

Although there has been some success in using image processing to measure wheals, a procedure that is automated and does not use the assistance of wheal outlines has yet to be established. For example, [6, 7, 4], have had success in automating wheal measurement using image processing software, but only from form outlines. Now, [10, 9] have shown progress in measuring wheals straight form a digital image, but automation and ease-of-use of their image processing methods are not apparent.

2.3 Digital Images

In this section we aim to explore image processing techniques for measuring wheals with emphasis on mathematical and statistical techniques used in this paper. First, a very simplified explanation of digital images is in order. In this paper we work with Red-Green-Blue (RGB) images. RGB images are stored in pixels, which have a square dimension. Many times in bad quality images, one can actually see these square pixels. Now, each pixel has a red, green, and blue value. Usually the red, green, and blue values range from 0-255, where 0 is the lowest intensity shade of the color and 255 is the highest intensity shade. Note, that any color can be made from shades of red, green and blue. So, a digital image is essentially a very large matrix of numbers, and we can use this fact to manipulate digital images. Hence, the terms image and image-matrix will be used interchangeably.

At the high-level, the general procedure of processing wheal images begins by manipulating the image so that the wheal, or more specifically the wheal edges, are easily recognized by a computer. Next, edge detection procedures are implemented to recognize the borders of a wheal. These first steps are challenging, but once accomplished it is fairly easy to calculate wheal area.

2.4 Image Manipulation

The goal of image manipulation is to distort the original image so that it will be easier for a computer to decipher a wheal from skin. In other words, the wheal is our region of interest in an image, and the surrounding skin is noise. In order to enhance the wheal in an image we can for example, distort the brightness/contrast of the colors of the image. Furthermore, we can change the color values of an image based on certain thresholds. Linear smoothing filters are a successful and popular technique for getting rid of noise in an image.

Linear smoothing filters are positive, (usually discrete)linear matrices with very small dimensions that are applied to each pixel in an image (and it's surrounding pixels). The resulting image will look smoother than the original image. Say for example, we have an image-matrix, Γ, and a linear smoothing filter matrix, γ. A noise-reduced or smoother image-matrix, Γ', is then calculated using the following equation.

$$\Gamma'(\mu,\nu) = \sum_{(x,y)\in\gamma} \Gamma(\mu+x,\nu+y)\cdot\gamma(x,y)$$

Where (μ,ν) represents pixel coordinates in the image, (x,y) represents pixel coordinates in the linear smoothing filter, and $x << \mu$, $y << \nu$.

Now, a Gaussian filter is a type of linear smoothing filter where the values in the filter matrix follow a discrete, two-dimensional Gaussian kernel.

$$K(x,y|\sigma) = \exp\left(-\frac{x^2+y^2}{2\sigma^2}\right)$$

where x and y determine the distance from the center of the filter-matrix. Below is an example a Gaussian filter matrix.

$$\begin{bmatrix} 0 & 1 & 2 & 1 & 0 \\ 1 & 3 & 5 & 3 & 1 \\ 2 & 5 & 9 & 5 & 2 \\ 1 & 3 & 5 & 3 & 1 \\ 0 & 1 & 2 & 1 & 0 \end{bmatrix}$$

See [11]

Another widely used image manipulation technique is to extract the red, green, or blue values from the image matrix and create a grey-scale image. Note, that to make any shade of grey in a digital image, one only needs to make the red, green, and blue values equal. This technique has proven to be very useful for identifying wheals. [6]

2.5 Edge Detection

There are many ways to approach edge detection in digital images. For example, [7] uses watershed techniques to identify pen outlines of wheals, which were described in more detail in the introduction. In this paper we consider two widely used edge detection algorithms, the Marr-Hildreth algorithm and the K-Means clustering algorithm.

The Marr-Hildreth algorithm is based on the fact that red, green, or blue values at edges are drastically different than the RGB values for the rest of the image. The first step of the algorithm smooths the image using the same Guassian filter described previously. The next step is to compute the Laplacian for all points (μ,ν) in the filtered image, Γ'.

$$\nabla^2 = \frac{\partial^2}{\partial\mu^2} + \frac{\partial^2}{\partial\nu^2}$$

The final step of the algorithm finds edge pixels by recognizing pixels for which there is a zero-crossing in the Laplacian. See [12]

Now, the K-Means clustering method assumes that the image-matrix data is distributed according to K clusters with K centers. Given K, the K-Means algorithm can identify the clusters, $\zeta_1, .., \zeta_K$, by minimizing the sum of squared distances among all the points, (μ, ν) in the image-matrix, Γ. Specifically, the goal is to minimize the sum of squared distances within in each cluster.

The algorithm first picks random points, $\alpha_1^{(\ell)}(\mu,\nu)..., \alpha_K^{(\ell)}(\mu,\nu)$, in Γ to represent the K centers for each cluster, $\zeta_1, .., \zeta_K$. It then assigns each image-matrix point, (μ, ν), to one of the clusters based on the following criteria.

$$(\mu,\nu) \in \zeta_n \quad if \quad ||(\mu,\nu) - \alpha_n^{(\ell)}|| < ||(\mu,\nu) - \alpha_m^{(\ell)}|| \tag{1}$$

for all $m, n = 1, .., K$ and $m \neq n$.

The next step in the algorithm then assigns new centers, $\alpha_1^{(\ell+1)}, ..., \alpha_K^{(\ell+1)}$, such that the sum of squared distances in each cluster is minimized. This minimum is attained using the following equation.

$$\alpha_n^{\ell+1} = \frac{1}{\Omega_n} \sum_{(\mu,\nu)\in\zeta_n} (\mu,\nu) \quad n = 1, ..., K \tag{2}$$

where Ω_n, represents the total number of image-matrix points in cluster n. The algorithm then reassigns image-matrix points as in equation (1) and reassigns new centers as in equation (2) for L iterations until $\alpha_n^{(\ell)} = \alpha_n^{(\ell+1)}$ for all $n = 1, .., K$ clusters.[13]

There are many different versions of the K-means algorithm, and in fact we specifically use a more robust version explained in more detail in section 3.

2.6 Calculating Area

The method used for calculating wheal area in this paper involves counting the number of pixels that lie within a wheal. Of course, once a wheal object is detected in an image, the methods of counting the pixels in the wheal vary. For example [8] uses an ellipse fitting approach for counting the number of pixels in a wheal. In this paper we count the number of pixels in the wheal by implementing a convex hull algorithm.

Now, consider the image of a wheal, θ, that is within a bigger image, Γ, and both θ and Γ exist in $\mathbb{R}^2$. The convex hull of θ is the smallest convex set that contains all points in θ. Hence, an convex hull algorithm will try to find the smallest convex polygon in Γ that contains all the points in θ. A convex hull algorithm also provides the vertices of this optimal convex polygon, and thus we can easily count the number of pixels once the convex hull is found. Note, that the vertices of the convex hull can also assist in calculating the longest diameter of a wheal. See Figure 2

Now, once the number of pixels in the wheal have been counted, we must convert pixels into a standard measurement of area, such as mm^2. This is somewhat a matter of experiment design. For example, [9] position a camera at a specific distance from a patient's arm when taking a picture of a wheal or wheals, and take this specified distance into account. For studies that measure wheals by scanning form outlines, they simply take into account the resolution of a scanner to convert to mm^2 [6]. In this paper we use reference objects in our wheal images, such as pennies or nickels, or we take into account the area of a standard piece of printer paper.

2.7 Image Processing Software

There is a wide variety of software available for implementing the image processing techniques described in this section. For example [6, 7] use Khoros, [6] uses ImageJ, [9] uses Adobe PhotoShop, and [9] uses OpenCV. For this paper, we decided to use R because it is quite flexible and is convenient for performing statistical analysis and wheal measurement data. We also used built-in image processing functions in MatLab to validate our results from R, and we get similar results under both programs. Lastly, we utilize ImageJ to obtain the box-bounding coordinates of form outlines.

Figure 1: FO image processing: cropped image(left), detected edges using Marr-Hildreth (middle), enveloped wheal using convex hull (right)

2.8 Data collection

The wheal images were collected at University of North Carolina Chapel Hill research facility where nurses performed SPT's, as described in subsection 2.1, on fourteen volunteers. Only seven of the fourteen volunteers reacted to the allergens whom provided 30 valid wheals. Hence, we analyze 30 wheals from form outlines.

Two test-panels, each with eight allergens, were applied to the volunteers. The first panel tested for Histamine (positive control), Glycerin (negative control) Mite Pteronyssinus, Mite Farinae, Cockroach, Tree Mix (GS) 10, Grass Mix GS7, and Weed. The second panel tested for Mold 1, Mold 2, Cat, Guinea Pig, Rabbit, Dog, Rat, and Mouse.

Pictures were then taken of the wheals with reference objects such as rulers or coins. After nurses measured wheals and outlined the wheals, more pictures were taken of the wheals with outlines on the skin. We were given the original copies of forms with longest diameter measurements and form outlines of wheals. We scanned in the form outlines, and we mainly imaged processed these form outlines to obtain wheal measurements. See Figure 1. Later, a researcher of this study also measured the longest diameter, perpendicular, and wheal size manually from the form outlines.

3 Establishing an Automated Measurement of Allergic Reactivity via Image Processing

In this section, we describe how to apply image processing for measuring wheal size and wheal area. In this study we worked with three types of wheal images: form outlines (FO), bare skin with outlines (BWO), and bare skin with no outlines (BS). See Figure 1. Note, As mentioned previously, using image processing to manipulate an image to detect a wheal is the main challenge. While detecting wheals from BS images is not the impossible, our BS images were of poor quality and were very difficult to manipulate in an automated fashion. Hence we were only able to work with FO and BWO images.

3.1 Calculating Wheal Measurements for Form Outlines

Fortunately, the FO image has a relatively simple image structure in terms of RGB. Hence, we were able to developed an automated method for FO images. The FO images contained several wheals in each image See Figure 1, thus we cropped the images to include only one wheal, and we performed our algorithm on each cropped wheal image. The first step of our algorithm manipulates the image using a Gaussian smoothing method, which blurs the image and removes noise. The next step of our algorithm implements the Marr-Hildreth algorithm to detect the edges of the wheal. Once the edges have been detected our algorithm then uses convex hull to envelop all the pixels within the wheal. Next, the algorithm counts the number of pixels within the convex hull and converts pixels to mm^2 to provide the area of the wheal. See Figure 2

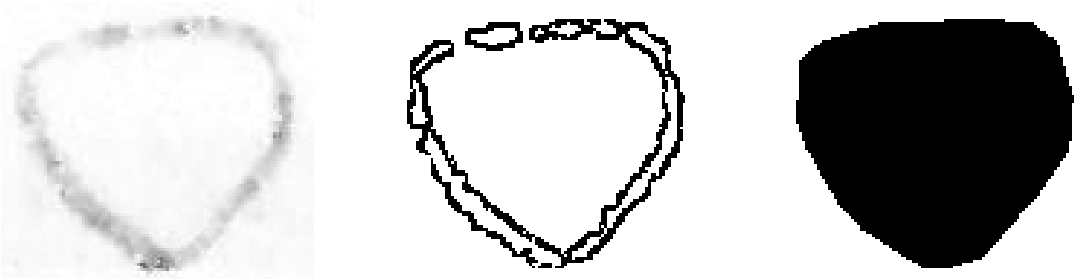


Figure 2: FO image processing: cropped image(left), detected edges using Marr-Hildreth (middle), enveloped wheal using convex hull (right)

As mentioned earlier we can obtain the vertices of the convex hull. Since the convex hull is the smallest convex polygon containing all points in the wheal, the longest distance between any two vertices is the longest diameter of the wheal. Also, by rotating the image we can measure the length of the perpendicular diameter. Hence, our algorithm automatically calculates longest diameter, perpendicular diameter, and wheal size.

3.2 Calculating Wheal Measurements for Bare Skin with Outlines

The procedure for BWO images is very similar to that of FO images. However, we use a different edge detection method because BWO images are more complex because they have a complicated color structure due to skin tone. The Marr-Hildreth algorithm is not sufficient for detecting wheal edges in BWO images because it is sensitive to noise. See Figure 4. Usually, this kind of problem can be solved by adjusting contrast and/or intensity of colors. However, manipulating the color tone requires special adjustments depending on the image, which defeats the purpose of automation. Therefore, we need a more robust method of edge detection.

First, we investigate the matrix structure of each color channel in BWO images in order to determine an appropriate method since general edge detection techniques rely on the image-matrix structure. From the exploration, we find that there is a pattern in the image-matrix. Figure 3 is a plot of color intensities for the $100th$ row in the image-matrix. We can see that all colors dramatically decrease at around the $50th$ and $300th$ column, and these points correspond to the outline around the wheal. Since a black pen was used to draw the outline, the plot looks reasonable. Now it is easy to see that we can use clustering methods to detect the edges of the wheal. Specifically, we use a similar version to the K-Means algorithm that is more robust than the version explained in section 2.

Robust k-means algorithm Let $\zeta^R = \{\zeta_1^R, \zeta_2^R\}$, $\zeta^G = \{\zeta_1^G, \zeta_2^G\}$, and $\zeta^B = \{\zeta_1^B, \zeta_2^B\}$ be classes of clusters for red, blue, and green channels, respectively.
1) For each channel, the point (μ, ν) is assigned to each channel cluster via a k-means algorithm in section 2.
2) If $(\mu, \nu) \in \zeta_1^R \cap \zeta_1^G \cap \zeta_1^B$, then assign $(\mu, \nu) \in \zeta_1$, else assign $(\mu, \nu) \in \zeta_2$.

Here, ζ_1 indicates the wheal outline and ζ_2 is skin. Since the algorithm addresses the cluster ζ_1 only when the element of the image matrix is assigned the first clusters for all channels, it becomes robust against noises. Figure 4(right) shows the result of the edge detection via the robust k-means algorithm. According to the figure, we know that the robust detector captures the wheal outline well.

4 Data analysis

In this section we first provide some descriptive statistics and definitions for the variables used in the analysis. Next we evaluate the inaccuracies of human wheal measurements. Then we analyze three relationships among

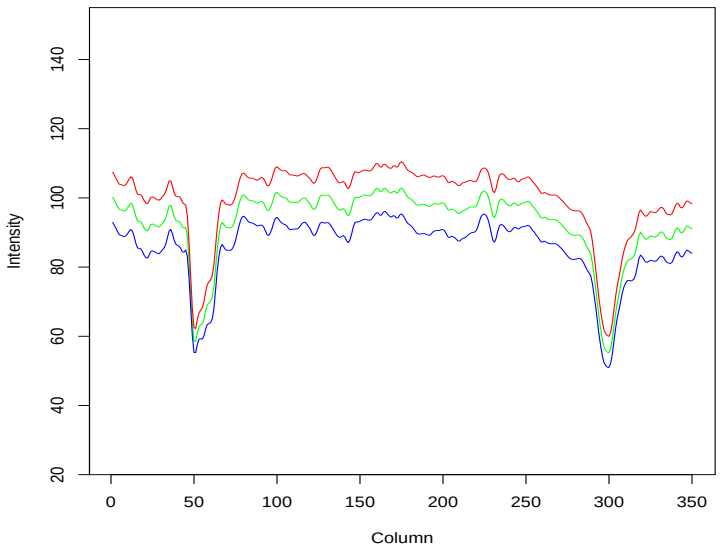


Figure 3: Plot of Intensity for $100th$ row color channels

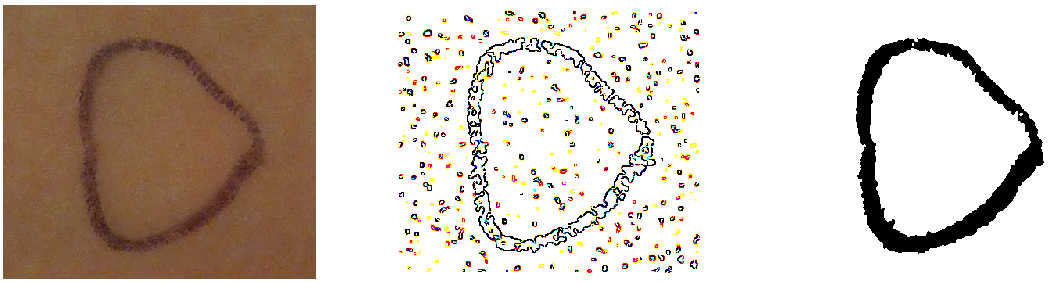


Figure 4: OS type image processing: original image(left), detected edges by Marr-Hildreth(middle), and detected edges by robust k-means(right)

our variables. First, longest diameter measured at the UNC Research Lab (LLD) versus wheal size measured by researcher of study (SWS). Next, SWS versus wheal size measured by R (RWS). And finally, RWS versus wheal area measured by R (RA).

4.1 Variables and Descriptive Statistics

The variables analyzed in this study are provided in Table 1. Our variables were obtained by three different methods, by hand, in Matlab, and in R, for each of the 30 wheals. See Appendix for raw data.

Variable	Description
LLD	Length of longest diameter of wheal measured at UNC Research Lab
SLD	Length of longest diameter of wheal measured by researcher of study
RLD	Length of longest diameter of a wheal measured in R
MLD	Length of longest diameter of a wheal measure in MatLab
SLP	Length of Perpendicular diameter (to the longest diameter) measured by researcher of study
RLP	Length of Perpendicular diameter measured in R
MLP	Length of Perpendicular diameter measured in ML
SWS	The size of the wheal calculated by researcher of study (SLD+SLP)/2
RWS	The size of the wheal calculated in R (RLD+RLP)/2
MWS	The size of the wheal calculated in MATLAB (MLD+MLP)/2
RA	Wheal area measured in R
MA	Wheal area measured in MatLab

Table 1: List of variables

Table 2 gives a list of mean and standard deviations of each variable. We can see that the difference between the average longest diameters, perpendicular diameter and wheal size among each method of measurement is negligible. Table 3 provides correlation coefficients for the variable according to the three measurement methods. We can see that all the variables are strongly correlated.

	Measurements	Mean	Sd
Matlab	LD	12.54	5.57
	LP	8.53	2.70
	Wheal Size	10.54	3.93
	Area	91.28	66.78
Hand	LD	12.13	5.42
	LP	8.23	2.70
	Wheal Size	10.18	3.84
R	LD	12.71	5.60
	LP	8.45	2.72
	Wheal Size	10.58	3.96
	Area	91.37	67.18

Table 2: The summary statistics for wheal's measurements

4.2 Measurement Error Evaluation

The first box plot on the left in Figure 4.2 plots the difference in longest diameter measured at the UNC Lab (LLD) and the longest diameter measured by R (RLD). Now, the second box plot in the figure plots the difference in the longest diameter measured by UNC Lab and the longest diameter measured by researcher of study (SLD). The first box plot shows most of the difference between LLD and RLD are negative, which strongly suggests that humans may be underestimating the true longest diameter. However, the second box plot suggests that longest diameter measurements between different humans are not obviously dissimilar.

		Wheal Size	Area
Matlab	Wheal Size	1.00	0.99
	Area	0.99	1.00
R	Wheal Size	1.00	0.99
	Area	0.99	1.00
Hand	Wheal size	1.00	0.99

Table 3: The analysis of correlation coefficients

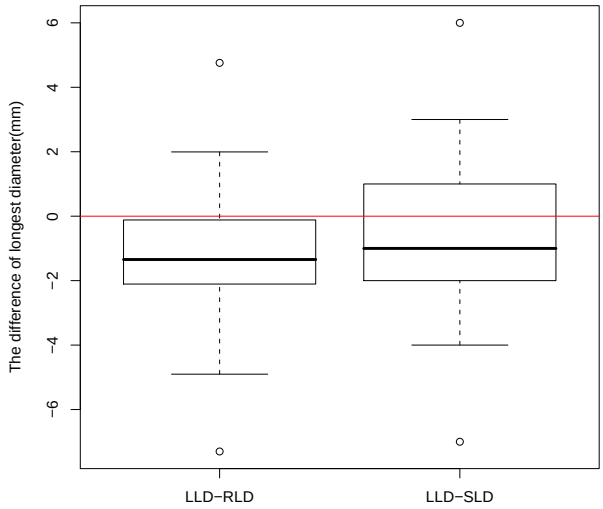


Table 4.2 provides T-tests for the longest diameter difference between LLD and RLD, and for the difference between LLD and SLD. We can see from the table that the difference between LLD and RLD are statistically significant at the 5% significance level. This result confirms our observation from the left box plot if Figure 4.2. Now, for LLD and SLD, there is no evidence that the two measures are statistically significantly different. These t-tests are valid as shown by the Shapiro-Wilk test.

	LLD-RLD	LLD-SLD
Shapiro-Wilk test	0.1902	0.1350
t-test	0.0075	0.1590

Table 4: The P-values of Shapiro-Wilk test and one sample t-test.

4.3 LLD vs SWS, SWS vs. RWS, RWS vs. RA

For the relationships between LLD and SWS; SWS and RWS; and RWS and RA, we fit regression models (linear or quadratic) and non-parametric models to the data for these variables. A breif explanation of non-parametric models is in order. When a regression model cannot catch the curvature of the data efficiently a more flexible model, such as a non-parametric model is needed. In these models we only deal with one independent variable, therefore the following non-parametric simple regression model is appropriate.

$$Y_i = f(X_i) + \varepsilon_i, i = 1, \ldots, n \tag{3}$$

where $f(.)$ is an unknown smooth function, ε_i are random errors with mean zero and finite variance σ^2. $f(.)$ will be estimated by using the local polynomial technique [14, 15].

Looking at Figure 4.3 and Table 5 we investigate the linear relationship between LLD and SWS. First, the linear model had a significant coefficient, however the residuals plot showed a slight pattern. Therefore, we fit a non-parametric model for our data, and found that the 95% confidence bands cannot cover the linear model, hence the data is not linear.

Now, again looking at Figure 4.3and Table 5 we also examine the linear relationship between SWS and RWS. In this case, the linear model can fit the data very well and the residuals plots look normal. Hence, SWS can totally represent RWS.

For RWS and RA we first fit a quadratic regression model to our data since wheal size is also considered the average diameter of a wheal. Furthermore, most of the wheals in the form outlines are circular in shape. Even with a quadratic term in our linear regression, there is still an obvious pattern in our residuals. Therefore, we fit a non-parametric model to the data. As in the first model, the 95% confidence bands cannot completely cover the fitted line of the quadratic linear regression. This implies that RWS and RA do not have a perfect linear relationship.

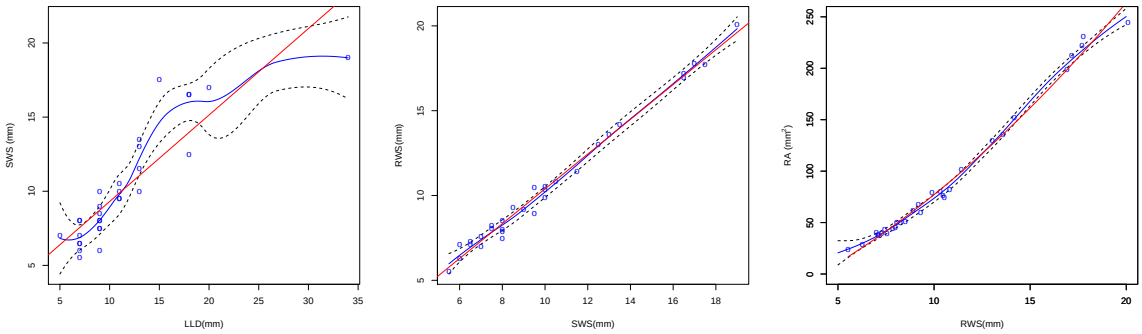


Figure 5: ((left) Red line indicates fitted line for linear model, blue line fitted line for non-parametric model for LLD vs SWS. (middle) Red line indicates fitted line for linear model, blue line fitted line for non-parametric model for SWS vs RWS. (right)Red line indicates fitted line for quadratic regression model, blue line fitted line for non-parametric model for RWS vs RA. The blue dots are real observations. The dash line is 95% confidence intervals for the non-parametric models.

		Estimate	Std. Error	t value	Pr(>|t|)
LLD vs SWS	Intercept	3.4765	0.7801	4.4566	0.0001
	Longest diameter	0.5832	0.0609	9.5830	0.0000
SWS vs RWS	Intercept	0.1375	0.2263	0.6079	0.5482
	Wheal Size true	1.0252	0.0208	49.2004	0.0000
RWS vs RA	Intercept	91.3659	1.2810	71.3264	0.0000
	Wheal Size	358.8912	7.0161	51.1527	0.0000
	(Wheal Size)^2	27.4925	7.0161	3.9185	0.0005

Table 5: The analysis results for parametric regression models.

5 Concluding Remarks

The R algorithm created in our study was extremely close to fully automating the process of estimating wheal area (and other measurements) using form outlines. The code for the algorithm can be found in the appendix.

The only manual part of our algorithm is that we used ImageJ to identify the bounding box coordinates of wheal outlines on a form. These images and the bounding box coordinates correspond to the two parameters of our algorithm. Hence an area of further research would certainly be how to create R code to recognize wheal outlines in in form outlines. Collecting more wheal samples and exploring the relationship between possible variables such as skin tone, race, gender and wheal sizes warrant future research work. In the future we hope to have better quality images so that we can create a machine learning algorithm to recognize wheals directly from skin.

6 Appendix A - Raw Data

	Matlab				Manual			R			
Wheal ID	LD	LP	Wheal Size	Area	LD	LP	Wheal Size	LD	LD	Wheal Size	Area
AC01	13.03	7.98	10.51	80.40	12	8	10.00	12.91	7.77	10.34	79.82
AC02	8.90	6.35	7.63	43.47	9	7	8.00	8.90	5.98	7.44	43.45
AC03	9.69	5.29	7.49	39.65	9	5	7.00	9.90	5.22	7.56	39.19
AC04	10.38	6.42	8.40	51.47	11	5	8.00	10.65	6.37	8.51	51.23
AC05	11.94	7.25	9.59	66.78	11	8	9.50	11.72	6.11	8.92	61.69
AC06	14.15	13.82	13.99	151.09	14	13	13.50	14.62	13.75	14.19	152.23
AC07	9.24	6.21	7.73	44.50	10	6	8.00	9.60	6.37	7.98	45.34
EK01	10.78	9.11	9.95	75.95	10	9	9.50	11.40	9.55	10.47	76.35
EK02	7.33	4.34	5.84	24.37	7	4	5.50	6.97	4.07	5.52	23.97
EK03	8.26	6.25	7.25	39.98	8	6	7.00	8.03	5.98	7.01	40.16
EK04	7.42	6.42	6.92	36.93	7	6	6.50	7.44	6.75	7.09	37.44
EK05	7.09	6.90	7.00	37.94	6	6	6.00	7.34	6.87	7.11	38.24
GG01	14.60	12.14	13.37	136.26	14	12	13.00	15.20	11.97	13.59	135.90
GG02	21.88	12.97	17.43	220.54	22	13	17.50	22.30	13.11	17.71	222.14
GG03	21.32	12.04	16.68	197.52	21	12	16.50	21.89	11.97	16.93	199.08
J01	11.55	9.36	10.45	82.74	12	9	10.50	12.09	9.55	10.82	82.47
J02	7.00	5.35	6.17	28.58	7	5	6.00	7.21	5.35	6.28	28.71
J03	8.93	7.19	8.06	49.86	8	7	7.50	9.11	7.00	8.06	50.08
J04	9.28	6.27	7.78	44.90	10	6	8.00	9.34	6.37	7.85	44.40
J05	29.10	10.88	19.99	243.07	28	10	19.00	29.24	10.95	20.10	244.47
J06	9.07	7.03	8.05	49.30	8	7	7.50	9.12	7.38	8.25	49.84
L01	10.69	8.13	9.41	67.10	10	8	9.00	10.59	7.77	9.18	67.19
L02	10.70	7.22	8.96	59.98	10	7	8.50	11.11	7.51	9.31	60.06
NW01	10.88	9.25	10.06	78.06	11	9	10.00	10.88	8.91	9.89	78.73
NW02	22.34	13.38	17.86	231.25	21	13	17.00	22.18	13.37	17.78	231.18
NW03	12.14	10.97	11.56	101.47	12	11	11.50	12.29	10.57	11.43	101.87
OS01	10.49	9.27	9.88	74.78	11	9	10.00	11.27	9.80	10.54	74.03
OS02	16.23	10.30	13.26	128.93	15	10	12.50	16.01	10.06	13.03	129.53
OS03	22.92	12.05	17.49	212.30	22	11	16.50	22.87	11.46	17.16	212.64
OS04	8.87	5.75	7.31	39.30	8	5	6.50	8.96	5.60	7.28	39.53

Table 6: The dataset collected from 7 volunteers

7 Appendix B - R Code

```
library(biOps)
library(cluster)
library(dismo)
library(AnalyzeFMRI)

image = readJpeg("C:\\IMSM\\elk.jpg")
crop = read.csv("C:\\IMSM\\elkcrop.csv", header=TRUE)

allergy = function(image, crop){

total.pixel=dim(image)[1]*dim(image)[2]
area.mm = (8.5*25.4)*(11*25.4)
```

```
ratio= area.mm/total.pixel
ratio2=11*25.4/2200

attach(crop)

bumpArea = vector("numeric", length=nrow(crop) )
LD = vector("numeric", length=nrow(crop))
Perp = vector("numeric", length = nrow(crop))
WhealSize = vector("numeric", length = nrow(crop))
label = seq(1, nrow(crop), 1)


for(i in 1:nrow(crop)){
  bump= imgCrop(image,BX[i],BY[i],Width[i],Height[i])
  plot(bump, main = label[i])

  Gau<-GaussSmoothArray(bump[,,],ksize=5)
  bump[,,]<-Gau

  bump.MH = imgMarrHildreth(bump,2)
  plot(bump.MH, main = label[i])

  #bump.new = r_dec_intensity(bump,0.2)
  #bump.new.MH = imgMarrHildreth(bump.new,2)

  G<- bump.MH
  train = which(as.matrix(G[,,1])==0,arr.ind = TRUE)
  test = which(as.matrix(G[,,1])<300,arr.ind = TRUE)
  ch = convHull(train)
  Pred = (1-predict(ch, test))*255
  G[,,] = Pred
  plot(G, main=label[i])
  BlueVal = imgBlueBand(G)
  pixels=sum(c(BlueVal)==0)
  bumpArea[i] = pixels*ratio

  Poly<-ch@polygons@polygons[[1]]@Polygons[[1]]@coords

  H<-length(Poly[,1])
  MaxDist<-0
  for(k in 1:H){
    I<-k+1
    if(I<H){
      for(m in I:H){
        Diam<-sqrt((Poly[k,1]-Poly[m,1])^2+(Poly[k,2]-Poly[m,2])^2)
        if(Diam>MaxDist){
          MaxDist<-Diam
          Point<-rbind(Poly[k,],Poly[m,])
        }
      }
    }
  }

  Slope<-(Point[1,2]-Point[2,2])/(Point[1,1]-Point[2,1])
```

```
  Middle<-round(c(mean(Point[,1]),mean(Point[,2])))
    G[Point[1,1],Point[1,2],]<-c(255,0,0)
  G[Point[2,1],Point[2,2],]<-c(255,0,0)

  theta<-atan(Slope)*2/pi*90
  Middle<-round(c(mean(Point[,1]),mean(Point[,2])))
  m1<-Middle[1]:(Middle[1]+1)
  m2<-Middle[2]:(Middle[2]+1)
  G[m1,m2,]<-c(200,200,200)
  plot(G, main=label[i])

 Rotate.G<-(imgNearestNeighborRotate(G,theta))
  M.rotate<-which(as.matrix(Rotate.G[,,1])==200,arr.ind = TRUE)
  alpha<-M.rotate[1,1]
  gamma<-M.rotate[1,2]
  T1<-0
  while(Rotate.G[alpha,gamma,1]<255){
    T1<-T1+1
    alpha<-alpha+1
  }
  Rotate.G[alpha,gamma,]<-c(200,200,200)
  alpha<-M.rotate[1,2]
  while(Rotate.G[alpha,gamma,1]<255){
    T1<-T1+1
    alpha<-alpha-1
  }
  ##################
  alpha<-M.rotate[1,1]
  gamma<-M.rotate[1,2]
  T2<-0
  while(Rotate.G[alpha,gamma,1]<255){
    T2<-T2+1
    gamma<-gamma+1
  }
  Rotate.G[alpha,gamma,]<-c(200,200,200)
  gamma<-M.rotate[1,2]
  while(Rotate.G[alpha,gamma,1]<255){
    T2<-T2+1
    gamma<-gamma-1
  }
  T<-min(T1,T2)
  #Rotate.G[alpha,gamma,]<-c(200,200,200)
  #plot(Rotate.G)
  #sum(G[,,1]==0)/(1700*2200)*28*21.65 # area of wheal

  Perp[i] = (T-2)*0.01273128*10 #perpendicular
  LD[i] = MaxDist*0.01273128*10 #longest diameter
  WhealSize[i] = (Perp[i]+LD[i])/2

}

  measurement = data.frame(Bump = seq(1, nrow( crop), 1), Area = bumpArea, Perpendicular = Perp,Longest
  return(measurement)
```

```
}
```

References

[1] Mannino DM, Homa DM, Akinbami LJ, Moorman JE, Gwynn C, Redd SC. *Surveillance for asthma-nited States,* 1980-1999. MMWR Surveill Summ 2002;51:1-13.

[2] Arbes SJ Jr, Gergen PJ, Elliott L, Zeldin DC. *Prevalences of positive skin test responses to 10 common allergens in the US population: Results from the Third National Health and Nutrition Examination Survey.* J. Allergy Clin. Immunol. 2005 116:377-83

[3] Gruchalla RS, Pongracic J, Plaut M, et al.*Inner City Asthma Study: Relationships among sensitivity, allergen exposure, and asthma morbidity.* J Allergy Clin Immunol. 2005;115:478-85.

[4] Konstantinou GN, Bousquet PJ, Zuberbier T, Papadopoulos NG. *The longest wheal diameter is the optimal measurement for the evaluation of skin prick tests.* Int Arch Allergy Immunol 2010;151:343-345.

[5] Calabria CW, Dietrich J, Hagan L. *Comparison of serum-specific IgE (ImmunoCAP) and skin-prick test results for 53 inhalant allergens in patients with chronic rhinitis.* Allergy Astma Procc 2009, 30:386-96.

[6] Prinz M, Vigl K, Wohrl S. *Automatic measurement of skin wheals provoked by skin prick tests,* Stud Health Technol Inform 2005; 116:441-446.

[7] Wohrl S, Vigl K, Binder M et al.*Automated measurement of skin prick tests: an advance towards exact calculation of wheal size.* Exp Dermatol 2006; 15: 119-24.

[8] Lamminen H, Voipio V. *Computer-aided skin prick test.* Exp Dermatol 2008;17:975-6.

[9] Vieira dos Santos R, Titus R, Cavalcante Lima H, *Objective evaluation of skin prick test reactions using digital photography,* Skin Research and Technology, 2007,13 148-53.

[10] Scheecki Robert,Lipiec Agnieszka, Grzanka Antoni *Quantitative Results of the Estimation of an Allergological Skin Prick Test Using Computer Planimetry and an Electrical Model of Simulation of Skin Reaction* Pol J Med Phys Eng 2006; 12(1): 43-51

[11] Parker, J., *Algorithms for Image Processing and Computer Vision.* John Wiley and Sons; 1996.

[12] Burger W, Burge MJ. *Digital image processing: an algorithmic introduction using Java.* New York: Springer-Verlag; 2008.

[13] J. A. Hartigan and M. A. Wong. *A K-Means Clustering Algorithm.* Applied Statistics,1979,28: 100-108.

[14] Fan, J. and Gijbels, I., *Local Polynomial Modelling and Its Applications,* Chapman and Hall, London, 1996.

[15] Loader, C. (2007), locfit: Local Regression, Likelihood and Density Estimation, R package version 1.5-4 **h**ttp://locfit.herine.net/

PROBLEM 6: SIGNAL DETECTION FOR DRUG SAFETY

Viviana García[1], Peng He[2], Xuefei Jia[3], Andrea Knezevic[4], Ying Lu[5], Mahmoud Shehadeh[6], Ranye Sun[7]

Problem Presenter:
Mark Wolffe
SAS Institute, Inc.

Faculty Mentor:
Chia Ying Lee
SAMSI

Abstract

Post-market vigilance of drug safety has been legally mandated for pharmaceutical companies and regulatory agencies. However, new developments in the quantitative methodologies of what has been called the *science of safety* have been scarce in the past few decades. The main source of post-market data for potential drug induced adverse events (AE) are spontaneous reporting systems (SRS) such as the Adverse Event Reporting System (AERS) managed by the Food and Drug Administration (FDA). The goal of analysis of these reporting systems is detection of new and unexpected drug-AE relationships that may be of potential harm to the public; in the literature this is referred to as signal detection. In this paper we review existing quantitative methods for signal detection in SRS that are in widespread use, the so-called disproportionality analysis (DA) methods. We identify known drug-AE relationships using historical data on FDA labelling changes and use AERS data on these pairs as case studies. We analyze these case studies using the existing methods, employing novel approaches of signal detection over demographic strata and over time. Using our case studies, we find that these analytic approaches are potentially valuable. Furthermore, we generate simulated SRS data for the purpose of testing the sensitivity of the existing DA methods. From this exercise we conclude that more simulation should be done and we strongly advocate the development of a reference database on which to test these DA methods. We conclude that DA in the context of signal detection in SRS are an important tool for pharmacovigilance and we conclude that the development of more sophisticated statistical methods to deal with the unique and complex problems presented by analysis of SRS are valuable.

1 Introduction and Motivation

Pharmacovigilance concerns the monitoring and detection of adverse events associated with the use of medicines. This process starts with designed clinical trials, and continues throughout the drug's life cycle after approval, when its use is widespread among the population.

In the post-approval environment, the primary method of data collection for surveillance purposes comes from spontaneous reporting systems (SRS), such as the Adverse Event Reporting System (AERS) of the Food and Drug Administration (FDA). These tools produce databases which contain a collection of reports of side effects, all of which are submitted voluntarily by clinicians, patients, or product manufacturers. Each report in an SRS database typically includes limited demographic information (such as age, sex, and weight), one or more drugs, and one or more adverse events.

The objective of creating these systems was to provide data that allows for the investigation of possible safety problems associated with the use of drugs, since some of these would be impossible to detect during the limited run of a clinical trial. In addition, clinical trials are unlikely to reliably detect rare, serious adverse

[1]Harvard University
[2]University of Cincinnati
[3]Georgetown University
[4]Georgia Institute of Technology
[5]University of Minnesota
[6]Northern Illinois University
[7]Texas A&M University

events that occur in subpopulations who have not participated in studies. Furthermore, as new medical products enter the market, the potential for interactions with other drugs, biologics, medical devices, and foods increases. Detecting possible relationships between drugs and adverse events in a timely fashion could prove extremely important to public health. Signal detection can be the first indication that a certain association should be studied more closely. It could also be informative for pharmaceutical companies as a continued testing scheme, to avoid potential lawsuits and to comply with the FDA regulations regarding surveillance.

In 2007, active post-market drug safety surveillance and analysis was mandated by a law passed in Congress, the Food and Drug Administration Amendments Act. In response, the FDA put in place the Sentinel Initiative with the ultimate goal of creating and implementing a national, integrated, electronic system for monitoring medical product safety. The Sentinel Initiative represents the implementation of what the FDA calls the *science of safety* which combines medical and pharmacological data with quantitative methods, with the goal to "generate hypotheses about, and confirm the existence and causal factors of, safety problems in the populations using the products"[2].

Hence the appeal for the development of analytic methods that might help identify possible starting points of interest. In particular, the drug safety literature often uses the term *signal* to refer to early hints that point at the possibility of novel and unintended drug effects. The approach of this investigation has to do with signal detection in SRS databases.

However, as can be imagined, there are a number of limitations inherent to the type of databases obtained from SRS. These issues should be noted and kept in mind when formulating conclusions or making decisions based on the information provided by the data. We just briefly mention some of them in this document.

First of all, these datasets are incredibly large and disorganized. To give an idea of their magnitude, the FDA receives more than 400,000 of these spontaneous reports each year [1]. The lack of a standardized nomenclature for drug names (including different names for the same drugs, misspellings, or the inclusion of dosages with the name, among others) and the use of multiple terms for similar clinical conditions presents a challenge.

Several problems appear due to the voluntary nature of the reporting process. One is the serious problem of over-reporting, which could occur for example because of the influence of publicity or a warning set up on a certain drug. In addition, there could be under-reporting, which may depend on the event and its severity, for example, or the lack of knowledge of the reporting system. Finally there could be multiple submissions: for example, when a person is taking a combination of drugs the report might sent to all of the manufacturers, who in turn file separate reports to the FDA. Additional problems present themselves because many reports of events do not necessarily reflect associations to the drugs that they allude to, and because there is limited information regarding the order of exposure and condition, or even the duration of exposure. Most importantly, SRS databases don't contain information about the number of patients at risk, that is, the population that was exposed to a specific drug. In short, there is considerable bias and noise in the data that undermines its reliability.

It is extremely important to note that any conclusions obtained from these databases *cannot* establish causality. At best, the analyses might identify potential issues and associations that must be confirmed by expert epidemiologists and clinicians in follow-up studies. Actually, many signals that emerge from spontaneous report databases are mostly noise, because there are many factors that are intermingled in a report such as treatment indications, co-prescribed drugs, reporting artifacts, etc., or because the reported adverse events are already labeled, are medically trivial, or biologically implausible.

Despite the many limitations of the available datasets necessary for post-approval analyses, there is an interest in the pharmacology community to develop analytic methods to quantify and detect signals that might appear in the spontaneous report databases, since these are the only sources of information currently available about drugs once they are widespread in the market. As previously mentioned, there are thousands of drugs and thousands of adverse events (AEs) that need to be studied. The complexity of these large datasets makes drawing inferences about the extremeness of drug-event counts intractable without the help of quantitative

summaries and analysis. This difficulty, in addition to the terms of the recently approved Sentinel Initiative, has generated more interest in these methods on the part of the regulators, the health care community, and industry.

There are multiple methods of signal detection presently in use, and one of the objectives of this project is to understand the current approaches and to explore and identify potential modifications or areas of improvement for these methods. Additional objectives include analyzing a dataset for a particular drug-adverse event that has not been investigated before, and exemplifying the importance of stratification based on demographic covariates.

This document is organized as follows. In section 2 we describe the AERS database and the particular subsets that we used for analysis. In section 3 we outline the four most commonly used signal detection methods, and highlight some of their advantages and limitations. In section 4, we propose three novel sequential methods to detect possible signals from a time series of disproportionality scores. Section 5 introduces three case studies constituted by specific pairs of drug and adverse event, which will serve as examples for our new methodologies. In section 6 we present our results. We first applied the four well-known signal detection methods to a particular case study and exemplified the importance of stratification to control for demographics. Secondly, we show the results of applying our new longitudinal signal detection methods to 3 case studies. The last part in this section concerns a simulation study. Finally, we wrap up with our conclusions and future work. All the analyses performed in this paper were done using available SAS software.

2 Description of the AERS Database

The data that we have used throughout this research is a subset of the Adverse Event Reporting System (AERS) database, which contains information of medical adverse events reported to the FDA. This database is publicly available, and it is updated every 3 months, which means that the reports are grouped in quarters per year. Data is available at the FDA website[8] for the first quarter of 2004 through the first quarter of 2012, and we focused on this specific subset of data corresponding to 33 quarters.

For each quarter, the AERS database consists of six major segments, including separate files for demographic, drug, reaction, patient outcomes, report sources, and therapy dates information. These datasets are connected by a primary link key with a unique number that identifies the AERS reports, as can be seen in Figure 1.

The demographic dataset contains 231,945 unique reports. Most of them are from United States (162,336), Japan (11,199), Germany (7,313), France (7,022), United Kingdom (6,872), and Canada (6,015). Among these reports, there are $169,272$ initial reports and $62,673$ follow up reports. The number of reports by females exceeds the one for males by over 50,000 reports, and there are also unknown and unspecified genders reported. Most of these reports were issued by consumers, followed by lawyers, and then medical doctors. 33,579 unique drugs and 9,516 adverse events are included in the reports. In the outcome dataset, 161,252 unique reports were filed. As a final outcome, 68,951 of cases were hospitalized, 29,138 cases of death, and 8,044 cases reported as life-threading.

One part of our analysis was conducted only on the data pertaining to the first quarter of 2012, and will appear in Section 6.1. The longitudinal part of the analysis will consider the information in all 33 datasets, and is detailed in Section 6.2.

For the purpose of visualization, we constructed a drug-adverse event network for 1,000 combinations of drugs and adverse events from the first quarter of 2012 of the AERS database (Figure 2). The network was drawn with the open source software Cytoscape[9]. The nodes represent both drugs and adverse events, while

[8] http://www.fda.gov/Drugs/GuidanceComplianceRegulatoryInformation/Surveillance/AdverseDrugEffects/ucm082193.htm
[9] http://www.cytoscape.org

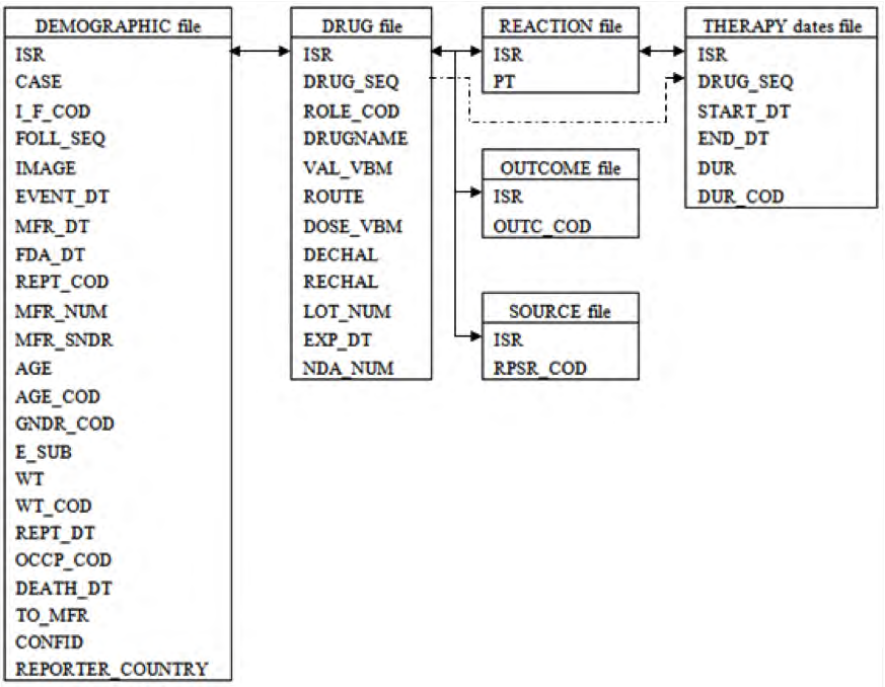


Figure 1: AERS database description.

the edges denote the relationships between them.

3 Existing methods for signal detection

The objective of signal detection methods is to filter the dataset to try to obtain evidence of potential associations between drugs and adverse events that were not known before, by providing a measure of how rare or common a particular combination is. Disproportionality analysis methods comprise the most widely used class of analytic methods for signal detection in SRSs. These methods quantify the extent to which a given condition is "disproportionally" reported with a given drug , compared to what would be expected (a control). In other words, an observed/expected ratio of probabilities or cell counts is obtained.

Considering the limitations of the databases mentioned above, it is clear that there is no real "control group". That is, since all the reports come in a voluntary basis, it is impossible to know how many people were exposed to the drug, how many people actually experienced an event, or even how many people experienced a particular event after taking a specific drug. This gives rise to a big complication in the quantification of the rarity of an adverse event, since without the total exposures it is difficult to evaluate the importance of its occurrence. To put this in terms of the mentioned disproportionality methods, the expected counts cannot be computed directly for any drug-adverse event pair.

The existing methods try to compensate for the fact that it is impossible to quantify a drug-adverse event rate directly, by using all other drugs and all other events in the dataset as a control (or background noise) against which to compare. Therefore, they focus on low-dimensional projections of the data, particularly 2-dimensional contingency tables, of the form shown in Table 1. The difference between the methods is the way in which the expected counts are modeled.

The most commonly used methods are the proportional reporting ratios (PRR), reporting odds ratios (ROR), the multi-item gamma Poisson shrinker (MGPS), and the Bayesian Confidence Propagation Neural Network (BCPNN). PRRs and related measures based on 2x2 contingency tables are currently used in routine

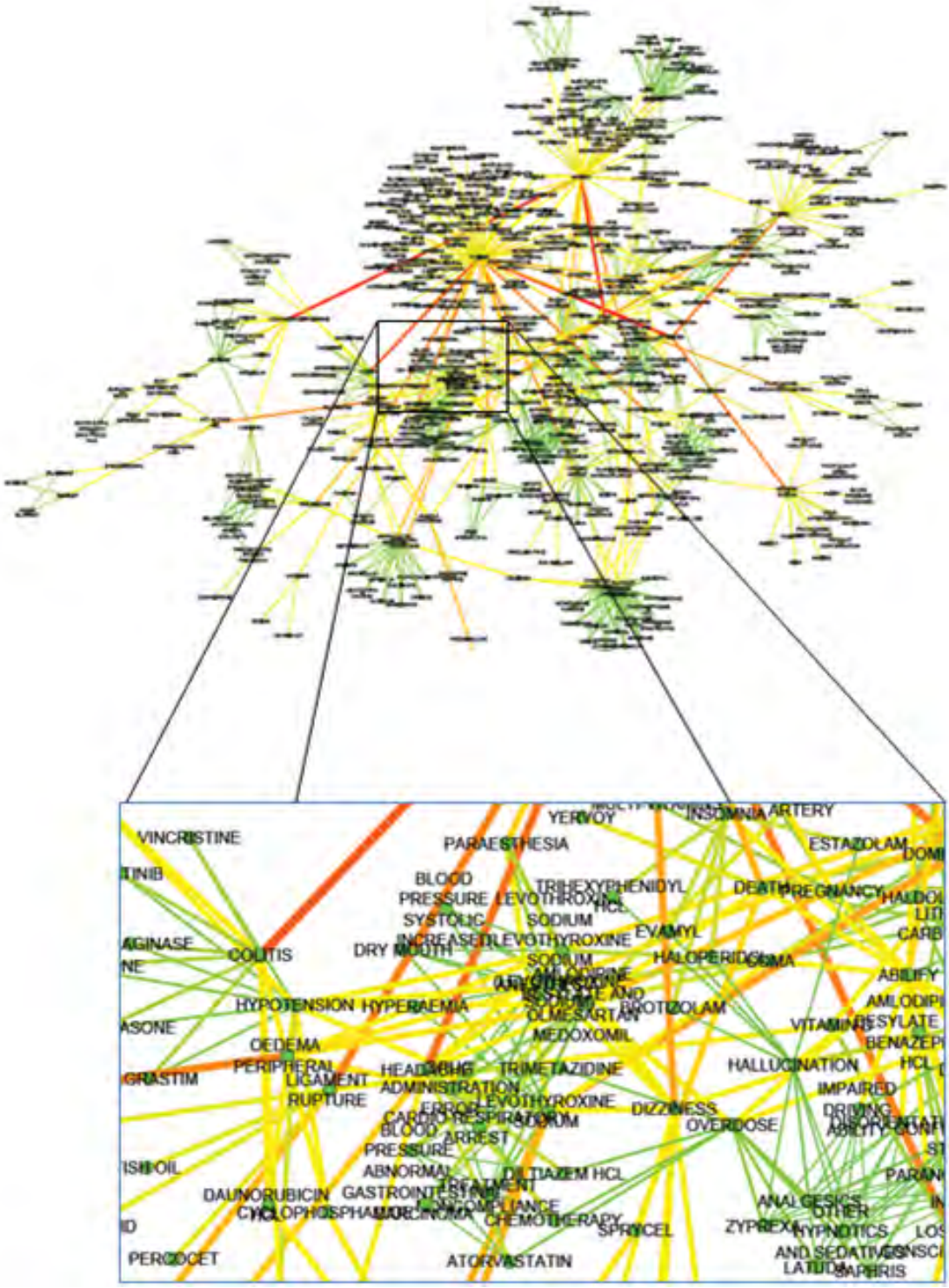


Figure 2: Network representation of drug and adverse event relationships in AERS database

pharmacovigilance activities by the Medicine Control Agency (MCA) in the UK. MGPS is currently used by the FDA, and BCPNN is employed by the World Health Organization (WHO)[7]. We give a brief description of these methods in the following sections.

	$\mathbf{AE_j = Yes}$	$\mathbf{AE_j = No}$
$\mathbf{Drug}_i$ = Yes	n_{00}	n_{01}
$\mathbf{Drug}_j$ = No	n_{10}	n_{11}

Table 1: Typical table for disproportionality analysis. AE stands for adverse event.

3.1 Proportional Reporting Ratio (PRR) and Reporting Odds Ratio (ROR)

The PRR is a very simple method inspired by the well-known relative risk calculation for contingency tables. By just focusing on a specific drug-adverse event combination, and pooling the counts over all other drugs and all other adverse events, it is possible to construct a 2x2 table as the one shown in Table 1. Then the PRR is computed as

$$PRR = \frac{n_{00}/n_{0\cdot}}{n_{10}/n_{1\cdot}}$$

where $n_{0\cdot} = n_{00} + n_{01}$, and analogously for $n_{1\cdot}$.

The ROR is very similar to the PRR, except for the fact that it tries to correct for certain kinds of under-reporting. It is calculated from the same 2x2 table (Table 1) as the PRR.

$$ROR = \frac{n_{00}n_{11}}{n_{01}n_{10}}$$

The interpretation of these quantities is that they measure how much more frequently the specific event is reported with the chosen drug, than with all other drugs.

It is important to keep in mind that whenever the count n_{00} is very small (which often happens in this type of datasets), this leads to substantial variability which increases the uncertainty about the true value of the measure of association to be computed. A known problem with PRR and ROR is that they do not address this issue, that is, there is no way to quantify the variability associated to this "sampling" variation. The two Bayesian methods that we will proceed to describe improve upon the methods based on relative ratios by addressing this issue, and provide solutions by considering all the reported drug-adverse event combinations at the time.

3.2 Bayesian approaches

The multi-item gamma Poisson shrinker (MGPS) and the Bayesian Confidence Propagation Neural Network (BCPNN) are Bayesian methods that aim to express possible associations between the reporting of events and drugs in terms of a function of the ratio of observed to expected frequencies mentioning drug i and adverse event j, n_{ij}/E_{ij}. That is, they look at a specific drug-event combination and try to quantify how "interestingly large" the number of reports is compared to what would be expected under the assumption of drug and event being statistically independent [7]. The expected counts E_{ij} are computed as

$$E_{ij} = \frac{n_{i\cdot}n_{\cdot j}}{n_{\cdot\cdot}}$$

where

$$n_{i\cdot} = \sum_{j=1}^{J} n_{ij}, \quad n_{\cdot j} = \sum_{i=1}^{I} n_{ij}, \quad n_{\cdot\cdot} = \sum_{i=1}^{I}\sum_{j=1}^{J} n_{ij}$$

are the total counts corresponding to drug i, the total counts corresponding to adverse event j, and the total number of reports, respectively. Throughout this document we denote the total number of drugs in a particular database by I, and the total number of adverse events by J.

In particular, the measure of disproportionality for a specific drug-adverse event combination is the *information criterion* (IC), computed as

$$IC_{ij} = log_2\left(\frac{n_{ij}}{E_{ij}}\right)$$

which is just the logarithm base 2 of the PRR.

3.2.1 Multi-item Gamma Poisson Shrinker (MGPS) - EBGM

In this method [5], each observed count for drug-adverse event pair is modeled as a draw from a Poisson distribution with varying unknown means. The means are considered to be random with a common prior distribution: a mixture of two gamma distributions (which have in total 5 parameters). In addition, an Empirical Bayes procedure is used to estimate the 5 parameters from the prior. In short, the model is

$$\begin{aligned} n_{ij} &\sim Poisson(\mu_{ij}) \\ \mu_{ij} &= \lambda_{ij} E_{ij} \\ \lambda_{ij} &\sim \text{Mixture Gamma} \end{aligned}$$

Finally, the posterior distribution of λ_{ij} is obtained, and the EBGM, defined as the geometric mean of the empirical Bayes posterior distribution of the true relative report ratio, is reported. This method is known as the Gamma Poisson Shrinker (GPS). As previously mentioned, the Bayesian methods try to account for the "sampling" variability in the reported counts, and studies have shown that the EBGM method does well even with very small n_{00} (even 1 or 2).

A variant of the above method, the Multi-Item Gamma Poisson Shrinker (MGPS), allows for higher order combinations of drugs and events that are significantly more frequent than their pairwise association would suggest.

3.2.2 BCPNN

The BCPNN method [4] is similar to EBGM, but uses a multinomial model instead of a Poisson for the counts, and calculates all cell counts for all potential drug-adverse event combinations in the database (not just those that appear together in at least one report). The fact that it is embedded in a neural network gives it the ability of to handle large data sets, and is robust to missing data.

In this case there is actually a proper prior (not estimated from the data as in the empirical Bayes approach), which is taken from the family of Beta distributions. Again a Bayesian procedure is used to obtain the posterior distribution of the IC between specific drugs and events present on the same report, as well as the 95% confidence intervals. In particular, an IC with a lower 95% confidence interval bound that increases with sequential time scans establishes a criterion for signal detection.

3.3 Discussion of the methods

Although these methods have widespread use as we have previously mentioned, we wish to highlight a few issues that became apparent when studying them in detail. We hope this will help practitioners to stay aware of the advantages and limitations of the existing methods, and to take them into account for interpretation of the results or development of new techniques.

The PPR and ROR are methods that are easily interpretable by practitioners because of the analogies that can be drawn to relative risks in epidemiology, and they simplify the problem to 2×2 contingency tables. If we consider that the dataset comprises I different drugs and J different adverse events in total, one would

essentially need to construct at total $I \times J$ tables of size 2×2 in order to analyze all the possible pairwise drug-adverse event combinations with these methods. But in this case, we run into a multiple testing scenario: we are running $I \times J$ tests on the same database without adjusting the family-wise error rate to account for the multiple comparisons, which will result in a large number of spurious relationships (or false-positive signals) that appear just by chance. That is, with millions of ratios being calculated, large ratios will inevitably appear just by chance, without necessarily meaning that there might be some interesting association. It is difficult to think how to incorporate an adjustment into the analysis, because the well-known methods such as the Bonferroni correction will potentially be too conservative, since the number of comparisons is huge (I and J are very large, and for example in our Case Study 1, we had an average of 2.7 million drug-adverse event combinations to look into).

Another issue that affects PPR and ROR is a small n_{00} count in the 2×2 table, as previously discussed. Analogously, a large count of a particular kind of adverse event report can potentially inflate the denominator for a specific drug, and reduce the sensitivity in detecting other signals associated with that drug. Finally, in comparing the frequentist methods (PRR and ROR) to the Bayesian methods, the first don't take into account the variability associated to the estimation of the measures of disproportionality, whereas the Bayesian methods do, by computing the entire posterior distribution.

In terms of the BCPNN, it seems to us that there is no available way to adjust for any stratification variables. As we will exemplify in our Section 6.1, stratification is important in order to control for demographic characteristics of the subjects, that is, in order to avoid spurious associations due to bad specification of the population that takes the drug.

The MPGS method seems to be very flexible in terms of its ability to potentially include drug-drug interactions (by creating a "new" drug which combines the counts for the two drugs included in the interaction), or even higher level interactions. The limitations of this ability would come in terms of the computational challenge of increasing the number of parameters to be estimated.

As we mentioned previously, in the MPGS-EBGM method, the estimate is a summary of the empirical Bayes posterior distribution of the true relative report ratio. Shrinkage towards the mean is a nice property derived from the fact that we are using an empirical Bayes approach to estimate the parameters of the prior distribution. What this method does is that it shrinks the calculated ratios in cases where the uncertainty is large (that is, when the variance in the estimate of the ratio is large), as would happen in the case of a really small n_{00} count. When this count is in the range of say, 10 to 20, there would be only a slight shrinkage; for large counts (e.g. over 100) there would be no shrinkage. This helps mitigate the peaks that would be obtained using a method like PRR, and therefore these estimates more stable in compareison.

All the methods suffer from the fact that all the calculations that are done are very dependent on what drugs and adverse events are included in the database. This is, we believe, partially inherent to the problem at hand since the whole issue is that there is no control group to compare the specific drug-event pair to, and so the "other drugs" and "other adverse events" that are used in the relative calculations will have a deep impact on the results. For example, if there are drugs that are included in the "control group" which have very high signals for the event of interest, the denominator would be inflated, which in turn would dilute the association that is the target of the particular analysis. Maybe some effort should be put into trying to define what group of drugs or events should be included in a specific analysis. This might include some sort of grouping by defined similarities, for example, or just following the same group of drugs across the different time periods.

As we have commented in several occasions, it is important to keep in mind that none of the conclusions that we obtain with these disproportionality methods should be interpreted as causal since there is no properly controlled randomized experiment involved. These methods can be useful as detectors of possible association between specific drug-adverse event combinations, that is, signals that can be identified for further study in a medical context.

4 Longitudinal signal detection

In this section we look at the historical score for a particular drug-adverse event combination. This serves as a reference for the physicians to judge whether the current score is off the track. We analyze the three drug-adverse event combinations that will be described in more detail in Section 5, namely Avandia & myocardial infarction, Finasteride & sexual dysfunction, and Thiazolidinedione & macular edema, from the first quarter of 2004 to the first quarter of 2012 (33 quarters overall). An EBGM score is calculated for each case, for each quarter. We decided to focus on the analysis of the EBGM scores because of its shrinkage and smoothness properties discussed in Section 3. We are particularly interested in whether there was a signal of disproportionality from the trend in the past. In this section, we present the methods that were developed in order to deal with this longitudinal analysis, and in Section 6.2 we discuss the results of applying them on our three case studies.

Our starting point is a time series of scores (say EBGM) for a specific drug-adverse event combination, which we denote by $\{X_i\}_{i=1}^N$, where i corresponds to the time period and N is the total number of time periods considered. If one were to plot it, by simply looking at the curve one would be able to spot certain spikes that might be deemed as signals. To conduct a more rigorous longitudinal analysis, we propose three novel ways to quantify sudden spikes that may potentially be signals: a method based on percent changes across time, a parametric approach and a non-parametric approach.

4.1 Method 1: Percent change in disproportionality score, relative to moving average

One way to quantify a sudden spike in a time series of disproportionality scores is to look at percent change in the score relative to the past. We can compare the score to a moving average of, for example, 1 year of scores. Percent change would thus be calculated as change relative to the average of the past 4 quarters:

$$\text{Percent change} = X_i \left(\frac{X_{i-1} + X_{i-2} + X_{i-3} + X_{i-4}}{4} \right)^{-1} - 1$$

Percent changes above some value, say 100% (that is, a doubling in value) may be considered worthy of investigation.

4.2 Method 2: Parametric approach

For the parametric approach, assume that the time series data $\{X_i\}_{i=1}^N$ comes from a Gaussian distribution. If no trend occurs, $\{X_i\}_{i=1}^N$ would be independent observations from a $N(\mu, \sigma)$ distribution. If there is a upward signal at time point $\tau + 1$, then $\{X_i\}_{i=1}^{\tau} \sim N(\mu, \sigma)$ independently, and $X_{\tau+1}$ would fall in the upper tail of Gaussian distribution. To implement this method, we assume that no signal occurs for the first four time points (which are used as a baseline). Starting from the fifth time point, we decide whether the current value is within two standard deviations of the mean. If so, we include that time point into the baseline, and re-estimate the mean and variance for the Gaussian distribution for further detection. If not, we report a signal. This sequential procedure continues until a signal is reported, that is, until the current time point is outside two standard deviations of the mean, where the mean and standard deviation are both estimated based on all the previous values.

4.3 Method 3: Nonparametric approach

The third algorithm we propose is the bootstrap approach. The key idea is as follows: if no trend occurs, we would expect the slope from the ordinary least squares (OLS) fit to be close to zero. On the other hand, if an upward trend occurs, we would expect the slope to be positive. To implement the method, again assume that the first four points do not show any trend. Starting from the fifth time point, we compute the OLS slope using

all the previous points and the current point. Its value is recorded as our test statistic. Next, we bootstrap from the previous points, say 10000 times, and compute the OLS slope each time. The p-value is calculated by computing the proportion of simulated slopes that are greater than our slope statistic. If the p-value is small, say less than 0.05, evidence exists that an upward trend is highly likely to occur. On the other hand, if the p-value is relatively large, we do not have evidence of an upward trend, and we include the current time point in the baseline group and proceed to check the next time point. This sequential procedure continues until a p-value is under a pre-defined threshold. The most commonly used threshold is a 0.05 level of significance.

In the end, these three methods may serve the following two purposes. First, they can be used to examine whether there was a signal in the past (for example, in a retrospective study to decide whether there was enough information to have captured certain signals before the public health situation became more difficult). Secondly, they can be used sequentially to determine whether the current time point is a signal or not. None of these methods detect multiple signals, which is an issue that generates another direction for future research.

5 Case studies

For each of our case studies, the goal is to examine the AERS data for evidence of reporting disproportionality in the given adverse event for patients taking the drug in question. In our first two case studies, the FDA issued a warning once the drug was on the market for a significant period of time[10]. Is is possible to detect a signal in the AERS data prior to the time of the FDA warning? Using the AERS data, how early could this potential link have been recognized? In our third case study, we examine an Adverse Event/Drug combination that has been documented in a recently published clinical trial, but has not resulted in FDA action. Can we find evidence in the AERS data to support the findings of this trial? Based on what we discover, can we make a recommendation to regulators about adding a warning?

5.1 Case Study 1: Avandia and Myocardial Infarction

The diabetes drug Avandia (Rosiglitazone) went on the market after FDA approval in 1999. The drug became popular; sales of the drug from GlaxoSmithKline peaked in 2006 at $3.2 billion in the United States that year [15]. In May 2007, the FDA issued a safety alert for the drug due to potential increased risk of heart attack. In 2010, the drug was suspended from the European market and the FDA severely restricted its use [15]. Pre-market clinical trials of Avandia showed no evidence of increased risk of heart attack; however, 8 years after the drug was approved, the FDA found enough evidence to lead to a warning contraindicating high risk patients and shortly thereafter, the drug was all but taken off the mass market.

There are two goals related to this case study. The first is to exemplify, via an analysis of the data for the first quarter of 2012, the consequences of stratification by age and gender. These results are presented in Section 6.1. The second goal is to analyze AERS data prior to the 2007 FDA warning for signal detection. We will apply our proposed trend analyses and the results are summarized in Section 6.2.1.

5.2 Case Study 2: Propecia and Sexual Dysfunction

Finasteride is a drug marketed as Propecia to treat male pattern baldness and Proscar to treat englarged prostate. Proscar went on the market in 1992 and Propecia in 1997. Pre-market clinical trials showed small but significant amount of sexual dysfunction [11]. The results of these trials were reported on the original label, however, in April 2011 and again in April 2012, the FDA revised the drug label to include new warnings that these drugs carry a potential risk of long-term sexual dysfunction. The goal in this case study is to look for evidence of disproportionality in reporting of sexual side effects in patients who reported taking Propecia prior to April 2011, and the results are discussed in Section 6.2.2.

[10] All specific drug label information below was retrieved from the Drugs-at-FDA website: http://www.accessdata.fda.gov/scripts/cder/drugsatfda/index.cfm. Leads to possible drug/AE combinations were found at the Pharmaceutical Drug Litigation Updates website: http://www.drug-injury.com/drug_injury.

5.3 Case Study 3: Thiazolidinediones and Macular Edema

Thiazolidinediones are a class of diabetes drugs that include Avandia (Rosiglitazone) and Actos (Pioglitazone). Macular Edema is an eye disease sometimes seen concurrently with diabetes; it is the leading cause of blindness in in diabetes patients. In July 2012, a retrospective cohort study of $103,368$ diabetic patients was published which found an increased risk of macular edema at 1-year and 10-year follow-up evaluations. Prior to this study, others that have investigated this link have found no causal evidence [10]. Post-market spontaneous reporting of Macular Edema is listed on the Avandia label; however, the FDA has not issued any warning regarding this particular drug/AE combination. The goal for this case study is to look for evidence of disproportionality in reporting of macular edema in patients who reported taking Avandia or Actos, and the results can be found in Section 6.2.3.

6 Results

6.1 Case Study 1: Avandia vs. Myocardial Infarction.

6.1.1 Description of datasets

Avandia has been brought under the scanner of the United States Food and Drug Administration (US FDA) in the context of adverse events related to Common Cardio Problem. FDA suspected that this drug yielded an unexpectedly large amount of heart related problems, thus issued a warning in the fourth quarter of 2007.

Is Avandia really that risky? By analyzing the retrospective data of the first quarter of 2012 that is publicly available in the US FDA Adverse Events database, it seems that Avandia deserves tremendous attention from physicians and drug manufacturers.

We begin by analyzing Avandia and each of the adverse events recorded in the database. There are 1136 Avandia-related adverse events. For each of the Avandia-related adverse event, we essentially compute its disproportionality, defined as observed counts/expected counts. Intuitively, if the observed count for a particular Avandia-adversed event pair is way higher than its expected count, it sends out alarms for investigation.

To serve this purpose, four methods are being used, namely PRR, ROR, BCPNN and EBGM. All of the four methods are based on analyzing the 2×2 contingency table(Table1) for each Avandia-adverse event pair.

After each pair is scored using those four methods, we rank the scores from highest to lowest. Results show that the four methods are in general consistent in the sense that four methods give similar ranking. Below are the top 10 Avandia-adverse related events rankings according to EBGM.

Drug	Event	EBGM	IC	ROR	PRR
AVANDIA	CORONARY ARTERY BYPASS	52.10	6.54	180.35	126.75
AVANDIA	CARDIOVASCULAR DISORDER	42.66	5.64	62.37	54.62
AVANDIA	STENT PLACEMENT	40.05	5.98	96.26	78.59
AVANDIA	MYOCARDIAL INFARCTION	30.10	4.97	37.37	34.62
AVANDIA	CORONARY ARTERY DISEASE	25.00	4.74	30.35	28.44
AVANDIA	CARDIAC FAILURE CONGESTIVE	20.89	4.44	24.60	23.37
AVANDIA	CEREBROVASCULAR ACCIDENT	17.44	4.18	20.12	19.28
AVANDIA	CARDIAC DISORDER	17.16	4.120	20.11	19.25
AVANDIA	CARDIAC PACEMAKER INSERTION	16.37	4.91	45.49	41.14
AVANDIA	IMPLANTABLE DEFIBRILLATOR INSERTION	15.65	5.11	64.02	55.67

Surprisingly, all of these adverse events are cardio related problems, indicating a strong warning that investigation be taken.
Moreover, we analyze Avandia-related adverse events by age and gender. We bracket age by classifying people with age 65 as old people, and age below 65 as young people (though most people in that group are mid-aged). This stratification is instructive because it may be that certain adverse events needs to be alerted for a specific

subgroup, while it is not a concern for another subgroup. In this study, we create subgroups of old people, young people, female and male. We also dig further by looking at old female, old male, young female and young male. All four scores are calculated and compared. We define it as our ungrouped dataset.
We also conduct another interesting study by subsetting all the drugs that were reported common cardio adverse events. This excludes irrelevant drugs in the study, and compares Avandia with all other drugs associated with common cardio adverse events. The group of common cardio adverse events is defined as the top 10 cardio related adverse events. By repeating the same stratification in the last paragraph, we create subgroups of old people, young people, female and male, old female, old male, young female and young male. All four scores are calculated and compared. And we define this as our grouped dataset.

The following tables and plots will give us more details and results.

6.1.2 Results of the analysis for the first quarter dataset of 2012 from AERS

- Combination of Avandia and cardiac disorder adverse event in the ungrouped dataset.

The four association measures - EBGM, IC, ROR, and PRR - of the combination of Avandia drug and cardiac disorder adverse event are computed using data from the first quarter of 2012. First of all, we deleted the pairs which have the counts that are less than 10. For the specific drug Avandia, there are 106 related adverse events. After stratifying by gender, there are only 39 related adverse events for female and 56 related adverse events for male. Only stratifying by age, there are 26 related adverse events left for old people while 40 for young people. Finally, we also stratified the dataset by gender and age at the same time, obtaining just 13 related adverse events for females under the age of 65, 12 for females over the age of 65, 13 for males over the age of 65, and 25 for males under the age of 65. We choose one cardiac related AEs (cardiac disorder) to show how the AE acts differently in different strata.

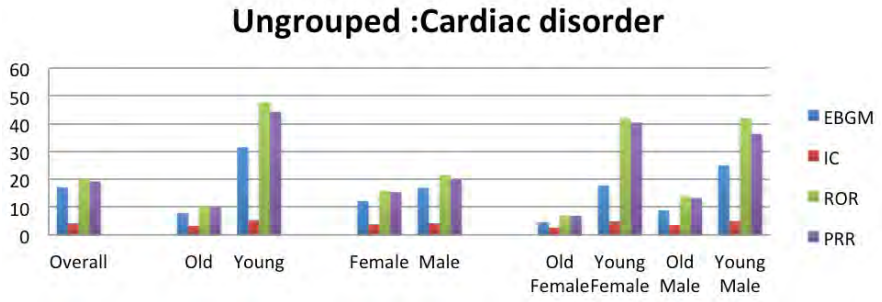


Figure 3: Avandia Use and Common cardiac disorder

Figure 3 shows that the scores calculated for people under 65 years of age are higher than those for people over 65 years old, and there is a slight difference between the male and female. The last group which is stratified by gender and age also shows that females under 65 years of age and males under 65 years of age have much larger scores than the females and males over 65 years old. The picture indicates that young people may be more prone to getting a cardiac disorder event than the older people when they take Avandia at the same time and same dose. This difference is not very clear between males and females. This stratification shows that probably an association with younger age groups is the main reason for the increase the total score as obtained from the non stratified database.

- Combination of Avandia and cardiac disorder adverse event in the grouped dataset.

For the result of the ungrouped dataset, we found out that 7 out of the top 10 high score Avandia-AE pairs are cardiac related, hence we are assuming that Avandia is more likely related to cardiac adverse events. It is reasonable for us to see how the score be changed in different strata. Therefore we chose the top 10

high Avandia related cardiac adverse events, and take them as a group named "COMMON CARDIO", and we keep the rest AEs as the same. Again, we deleted the pairs which have the counts that are less than 10. So in the grouped dataset, there are 98 adverse events in total. And stratified by gender, there are 23 related adverse events for female and 51 related adverse events for male. And stratified by age, there are 24 related adverse events for old while 25 for young. And we also stratified the dataset by gender and old, so there are 13 AEs for young female, 12 for old female, and 13 for old male and 23 related AEs for young male.

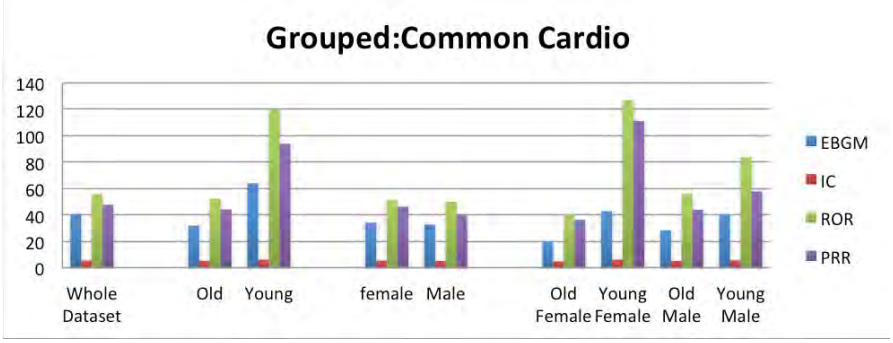


Figure 4: Avandia Use and Common cardiac disorder

For each group, we used four methods to get the score (EBGM IC ROR and PRR). We choose the "COMMON CARDIO" AE to see how the AE acts differently in different strata. Figure 4 shows that the young people have a higher score then the old people; and there is a slight difference between the gender groups; the last group which is stratified by gender and age also shows that young female and young male have much larger scores then the old female and old male. And RPR and ROR score of young female are much larger than the score for young male. The plot indicates that young people may more likely to have a cardiac disorder than the old people when they take Avandia. But the difference is not very obverse between male and female. So it probably the young people, especially young female, that increase the total score.

- Summary. From the two groups of dataset, the scores of the grouped AE dataset are higher than the scores of the ungrouped AE dataset. we get pretty much the same result, that is young people are more likely to have a cardio related adverse event compare to old people while there is little difference between male and female. Also, by comparison, the difference between EBGM, ROR, and PRR for the whole dataset group is less than the difference for the "young" group which suggests that examining the data by different demographic factors, such as age and gender, would lead to better results of detecting signals of drugs.
- Limitations and future directions: This study suffers from several limitations. First of all, we do not have the BMI for each patient. BMI may be an important factor to stratify. Second, the data is a collection of spontaneous response from physicians and patients, which may suffer from sampling bias. Third, it would be better if drugs are classified, for example, by biological component and chemical component. Studying how these two types of drugs relate adverse events would be an interesting and meaning topic. Lastly, it might be worthwhile to include all the drug-adverse event pairs in the ranking.

6.2 Time trend analysis

6.2.1 Case Study 1: Avandia and Myocardial Infarction

The quarterly AERS data allows us to analyze disproportionality measures longitudinally. Here we plot disproportionality measures for all time points preceding the issue of the FDA warning on Avandia regarding heart attacks.

The four methods produce very similar results. The only noticeable difference is that he EBGM does not spike as high in 2006 Q4, perhaps because this measure is less sensitive to small changes in the number of

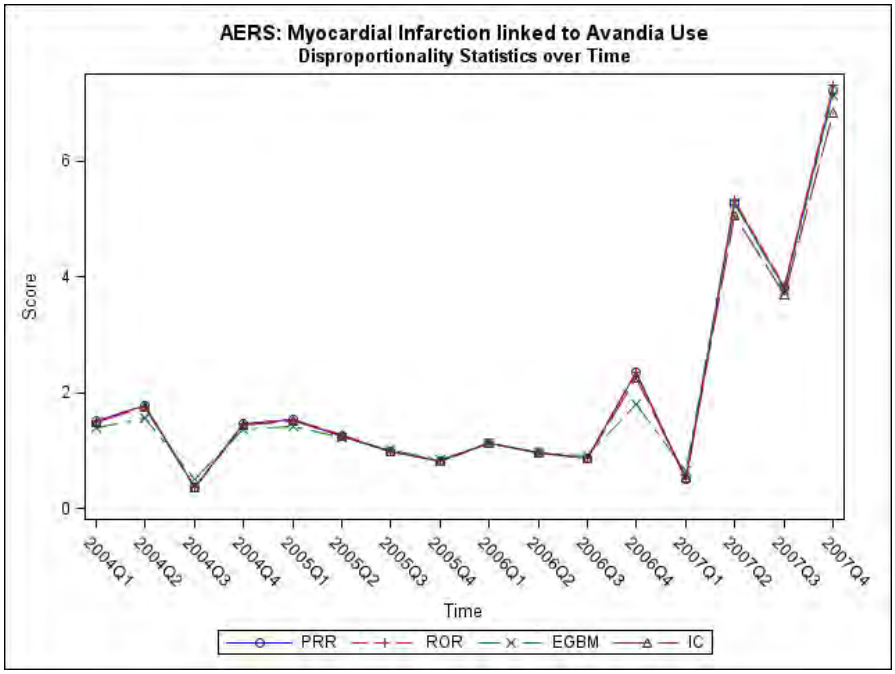


Figure 5: Myocardial Infarction linked to Avandia Use

drug/AE pairs reported (i.e. small changes in the number n_{00}).

The FDA warning was issued after the first quarter of 2007. With the benefit of hindsight we can interpret the spike in the fourth quarter of 2006 as a signal. Indeed this is consistent with what we see in the plot; before 2006 Q4, the disproportionality rates hover around 1 (no disproportionality) and then in 2006 Q4, they suddenly double.

For reasons discussed earlier, we pick the EBGM score to use in our longitudinal analysis. We employ our methods of longitudinal data analysis on EBGM score with the results in 2. The percent change and deviation from the mean methods successfully identify 2006 Q4 as the first time point to produce a signal; the non-parametric trend method does not identify a signal until 2007 Q2. Notice that all three methods produce results starting in 2005, that is, only after 4 initial time points. At least 4 time points are needed to calculate baseline statistics from which measures of change are determined.

6.2.2 Case Study 2: Propecia and Sexual Dysfunction

In the Finasteride/Sexual Dysfunction case study, the 4 disproportionality methods produce different results. The PRR and ROR scores overlap almost precisely, and they are both more sensitive to changes in number of reported drug/AE pairs, hence they produce dramatic spikes. The EBGM and IC methods follow the same trend as the frequentist methods, but produce smoother curves.

Again, we choose EBGM scores for further investigation. The plot of EBGM score over time shows a few spikes that may potentially be signals: 2008 Q3, 2010 Q1 and 2011 Q3 stand out to the naked eye. The FDA added sexual dysfunction to the Finasteride label after 2011 Q1 so we are most interested in detecting signals at 2008 Q3 and 2010 Q1.

The percent change method detected two signals prior to the one we expect to see at 2008 Q3 (see 3. The percent change is 169% at 2006 Q2 and 220% at 2006 Q4. This algorithm successfully detected 2008 Q3 (249%) and 2010 Q1 (144%), as well as the last three time points on the plot (283%, 190%, 179%).

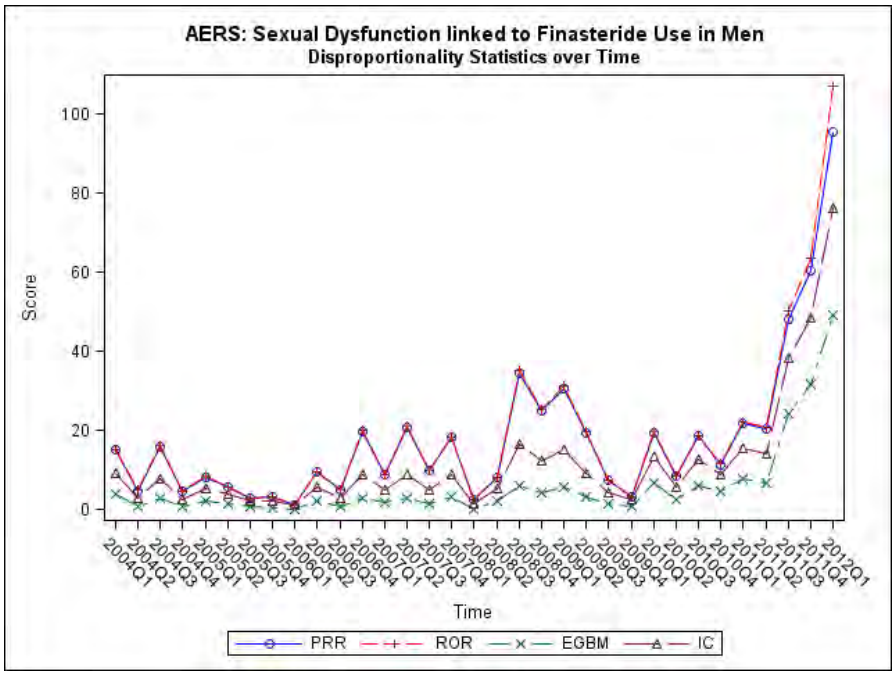


Figure 6: Sexual Dysfunction linked to Finsteride Use

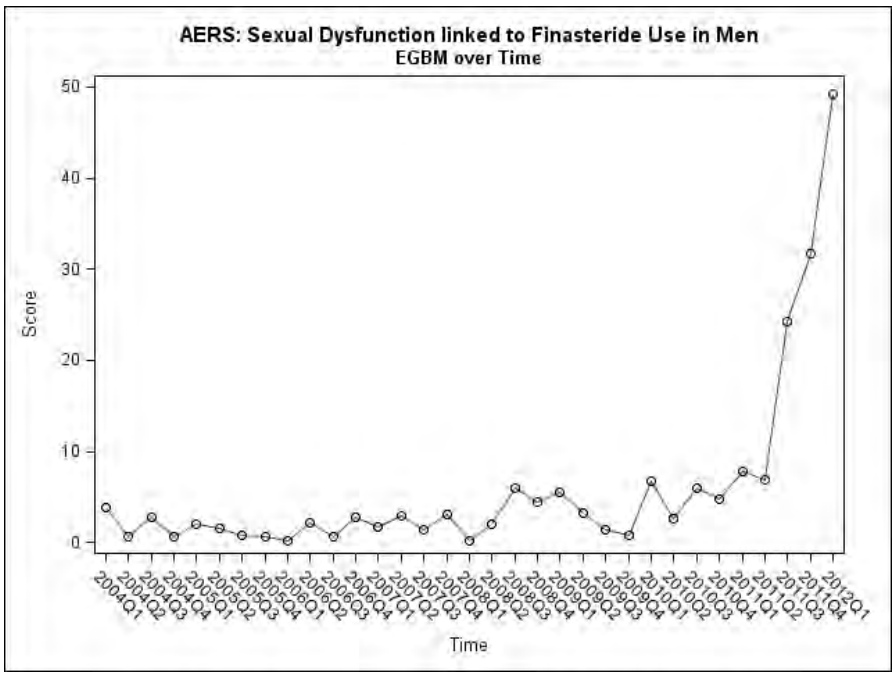


Figure 7: Sexual Dysfunction linked to Finasteride Use

Time	Method 1 (% change)	Method 2 (p-value)	Method 3 (p-value)
2005Q1	17%	0.66	0.98
2005Q2	0%	0.94	0.98
2005Q3	-11%	0.52	0.76
2005Q4	-32%	0.31	0.55
2006Q1	0%	0.91	0.52
2006Q2	-7%	0.56	0.42
2006Q3	-8%	0.46	0.23
2006Q4	**88%**	**0.02**	0.88
2007Q1	-47%	0.13	0.48
2007Q2	386%	<0.001	<**0.001**

Table 2: Avandia/Myocardial Infarction: Longitudinal signal detection results (EBGM)

The parametric method successfully detected 2008 Q3 and 2010 Q1 as well as the last three time points ($p<0.001$). It did not detect any other time points. Finally, the non-parametric trend method only detected a signal in 2009 Q1, two time points after the initial spike in 2008.

To summarize, in this case we see that, while the percent change method successfully detected the signals we expected, it was too sensitive to small changes in score and detected two signals erroneously. The non-parametric trend test lagged behind and only detected a signal two time points after one occurred and the parametric method successfully detected the signals we expected and no others.

6.2.3 Case Study 3: Thiazolidinediones and Macular Edema

In the Thiazolindinedione/Macular Edema case study, we see again that while the 4 disproportionality methods follow the same trends, they produce different results that are due to the PRR and ROR scores being more sensitive to small changes. Overall, we see more variability over time in this drug/AE pair, with scores suddenly spiking, even on the smoothest curve (EBGM). This drug/AE pair presents the most challenging scenario of signal detection of the three cases.

Recall that evidence of the link between Thiazolidinediones and Macular Edema was published very recently and the FDA has not made a recommendation nor added a warning regarding this link. Any signals detected in this data could potentially be worth investigation. To the naked eye, 2006 Q1, 2010 Q1 and 2010 Q3 stand out as potential signals. 2004 Q2 is also a spike, but will not be detected by our methods as it in the set of first 4 measures and must be used to calculate baseline statistics.

2006 Q1 is detected as a signal by the percent change method (678%) and the parametric method ($p<0.001$), but not by the non-parametric trend method. The non-parametric trend method picks up 2006 Q4 as a signal, again a few time points behind.

We note here that the non-parametric trend method, while not good at picking up single time point spikes, works well to detect signals where several time points in a row have an elevated score compared to the past, as in 2006 Q1-Q4 here.

2010 Q3 is detected by both the percent change method (235%) and the parametric method ($p<0.001$). 2009 Q4 is only detected by the percent change method (166%). For this time point, we cannot judge whether the sensitivities of percent change and deviation from the mean are too high or too low, since we can only speculate as to whether this time point represents a 'real' signal.

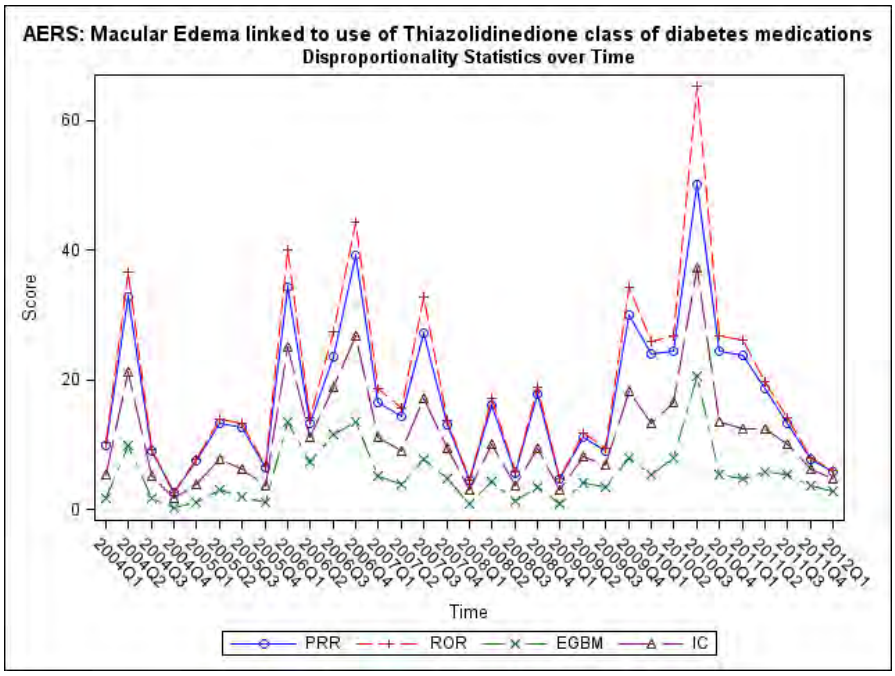


Figure 8: Macular Edema linked to Thiazolidinedione Use

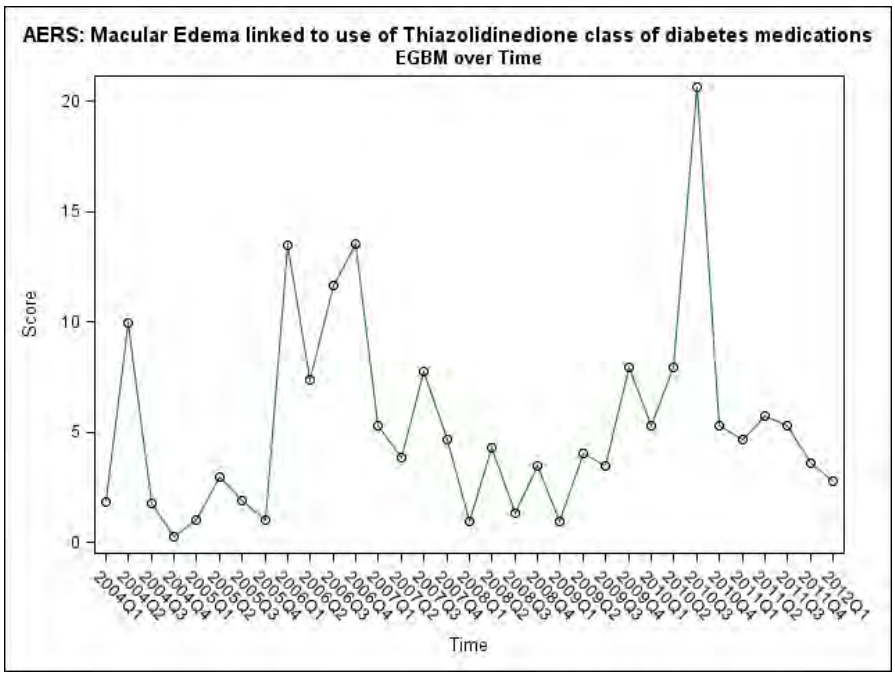


Figure 9: Macular Edema linked to Thiazolidinedione Use

Time	Method 1 (% change)	Method 2 (p-value)	Method 3 (p-value)
2005Q1	5%	0.95	0.46
2005Q2	6%	0.78	0.58
2005Q3	-53%	0.39	0.43
2005Q4	-56%	0.33	0.27
2006Q1	84%	0.25	0.14
2006Q2	**169%**	0.58	0.27
2006Q3	33%	0.45	0.24
2006Q4	**220%**	0.23	0.53
2007Q1	23%	0.85	0.62
2007Q2	55%	0.26	0.86
2007Q3	-27%	0.86	0.91
2007Q4	41%	0.17	0.69
2008Q1	-90%	0.18	0.88
2008Q2	6%	0.73	0.95
2008Q3	**249%**	**<0.001**	0.17
2008Q4	53%	0.09	0.11
2009Q1	75%	0.02	**0.02**
2009Q2	-29%	0.56	
2009Q3	-71%	0.61	
2009Q4	-79%	0.41	
2010Q1	**144%**	**<0.001**	
2010Q2	-15%	0.89	
2010Q3	106%	0.04	
2010Q4	19%	0.22	
2011Q1	56%	0.01	
2011Q2	29%	0.05	
2011Q3	**283%**	**<0.001**	
2011Q4	**190%**	**<0.001**	
2012Q1	**179%**	**<0.001**	

Table 3: Finasteride/Sexual Dysfunction: Longitudinal signal detection results (EBGM)

By carefully observing the time point at which a signal was detected, we find interestingly that the bootstrap approach does not detect a signal as fast as the parametric approach. This is mainly due to the nature of non-parametric bootstrap: by having fewer assumptions, we lose efficiency. On the other hand, this conservativeness gives more reliable results. The parametric approach, however, detects a signal quickly, but it suffers when the assumption may not be valid and is more likely to give false signals than the bootstrap approach.

6.3 Simulation

In this section, we show attempt numerical experiments of the approaches listed above in Section 3 using simulated data. Because the uncertainty and complexity of the real data described previously is very challenging, a more clear insight could potentially be obtained through simulations where the truth is known. This is especially important because there is no gold standard for signal detection techniques in the literature. In addition, little work has been conducted regarding simulation in the pharmacovigilance field. Three recent papers proposed simulation of data generation processes but with very different philosophies [13], [14], [3]. In our case, we address this issue in a different way with many interesting outcomes.

Time	Method 1 (% change)	Method 2 (p-value)	Method 3 (p-value)
2005Q1	-70%	0.58	0.66
2005Q2	-9%	1.00	0.75
2005Q3	28%	0.77	0.6
2005Q4	-34%	0.58	0.55
2006Q1	**678%**	**<0.001**	0.52
2006Q2	53%	0.44	0.42
2006Q3	96%	0.10	0.15
2006Q4	61%	0.07	**0.04**
2007Q1	-54%	0.96	
2007Q2	-59%	0.74	
2007Q3	10%	0.64	
2007Q4	-39%	0.85	
2008Q1	-83%	0.31	
2008Q2	-1%	0.83	
2008Q3	-70%	0.38	
2008Q4	24%	0.73	
2009Q1	-62%	0.35	
2009Q2	61%	0.87	
2009Q3	44%	0.77	
2009Q4	**166%**	0.42	
2010Q1	30%	0.90	
2010Q2	53%	0.43	
2010Q3	**235%**	**< 0.001**	
2010Q4	-49%	0.97	
2011Q1	-52%	0.86	
2011Q2	-41%	0.96	
2011Q3	-42%	0.97	
2011Q4	-32%	0.68	
2012Q1	-43%	0.56	

Table 4: Thiazolidinedione/Macular Edema: Longitudinal signal detection results (EBGM)

6.3.1 Data generation

Unlike previous work by Ahmed, et al. [3], which suggests generating data in terms of number of events, our starting point is to generate data in terms of the patient's reporting mechanism.

We assume the distribution of one adverse event $AE_j,\ j \in \{1, \cdots, J\}$ from a patient's report follows a Bernoulli distribution with success probability p_{AE_j}. The probability p_{AE_j} in the Bernoulli distribution for patient N is determined by the following equation:

$$p_{AE_j}(N) = Prob(AE_j|patient\ N) = \frac{1}{e^{-(\beta_0+\beta_1 x_1+\ldots+\beta_I x_I)}+1} \tag{1}$$

where $x_i \in \{0, 1\},\ i = 1, \ldots, I$ is an indicator for whether the patient is using drug i. Let $\beta_i, i \in \{1, \cdots, I\}$ denote the coefficients of the drug effect to a particular AE. The larger value of β_i, the higher the effect drug i has on AE_j. Let β_0 denote the constant, which could be viewed as the background noise in the simulation.

Instead of directly generating counts for each drug-event combination, we first generate the patient reports, each with a number of drugs and AEs. The counts of each drug-event combination will then be calculated

from the patient's report.

For all simulations, we consider the number of drugs to be $J = 4$, and $I = 4$ the number of AEs. We will generate information for $10,000$ patients. Our simulation study consists of the following scenarios:

CASE 1. The reporting of adverse events are completely random, and independent with any possible factors. Without loss of generality, we assume $AE_j,\ j \in \{1, \cdots, J\}$ has 50% probability to appear in a patient's report.

CASE 2. The chance of appearance of $AE_j, j \in \{1, \cdots, J\}$ depends on the Bernoulli distribution described in (1). In CASE 2, we fix $\beta_0 = -2.2$, and the other coefficients are given in Table 5. In this setup, AE_1 is mainly dominated by $drug_1$, AE_2 is mainly dominated by $drug_2$, AE_3 is mainly dominated by $drug_1, drug_2$ and $drug_3$, and AE_4 is dominated by $drug_1, drug_2, drug_3$ and $drug_4$.

	β_1	β_2	β_3	β_4
AE_1	2.00	0.00	0.00	0.00
AE_2	0.00	2.00	0.00	0.00
AE_3	1.50	1.50	2.00	0.00
AE_4	1.50	1.50	1.50	2.00

Table 5: Coeffients in simulation CASE II.

CASE 3. In this case, we increase the first cell coefficient in the above table, and others remain the same (Table 6). In this case, the influence of AE_1 by $drug_1$ increases from 37.75% to 78.58%. Compared to CASE

	β_1	β_2	β_3	β_4
AE_1	3.80	0.00	0.00	0.00
AE_2	0.00	2.00	0.00	0.00
AE_3	1.50	1.50	2.00	0.00
AE_4	1.50	1.50	1.50	2.00

Table 6: Coeffients in simulation CASE III.

II, in CASE III the probability of the drug-event combination $drug_1 - AE_1$ is increased.

CASE 4. In this case, we increase every cell from Table 6, which means the counts of every drug-event combination increases significantly (Table 7). In CASE IV, the probabilities of all counts have a very large

	β_1	β_2	β_3	β_4
AE_1	3.80	0.00	0.00	0.00
AE_2	0.00	3.80	0.00	0.00
AE_3	2.50	2.50	3.80	0.00
AE_4	2.50	2.50	2.50	3.80

Table 7: Coeffients in simulation CASE IV.

jump.

For each individual case from the setup above, we generated 50 datasets. Then for each dataset we used the four disproportionality methods described in Section 3 to obtain the PRR, ROR, EBGM, and BCPNN scores for each drug-AE combination, and finally we calculated the mean and variance across the 50 datasets (to account for simulation variability).

6.3.2 Simulation results

Tables 8, 9, 10 and 11 in the Appendix show the results for CASE I, II, III, and IV, respectively. Every drug-event combination is listed in the results table for the four drugs and four AEs under consideration. Columns N00 through N11 are the corresponding cells in the 2×2 contingency table pictured in Table 1. The columns EBGM, IC, PRR, and ROR correspond to the measures of disproportionality calculated for each drug-adverse event combination through each of those four methods, and are shown here together with their standard deviation.

In Table 8, since the assumption for the simulation was that adverse events and drugs are reported completely at random and independently, the values of EBGM all equal one for every drug-event combination. Similar results occur for the other three methodologies: IC, PRR, and ROR. There is no sign of disproportionality in our simulation CASE I, as expected.

In Table 9, the probability of $drug1 - AE1$ is around 30%. This is reflected from N00 in the first row. However, the EBGM is still very close to 1, and PRR and ROR do not change significantly. In Table 10, the EBGM, PRR and ROR are still close to 1, which indicates that all existing algorithms have difficulty detecting the signal change, even when the probability changes dramatically. In Table 11, we obtained similar results for EBGM, PRR and ROR, although they have a slightly higher volitality than the results before. Although the counts in N00 have increased significantly compared to Table 9, the signal detection approaches have failed to find these changes.

Through this simulation, we found some potential issues to look into regarding those existing approaches.

- None of these approaches take drug-drug interactions into consideration. The only difference between CASE II and CASE III is that the probabilty of AE_1 caused by $drug1$ changes from 37.75% to 78.58%. In Table 10 and 11, the N00 value for drug1-AE1 changes from 1378.54 to 2273.06, which is an expected increase since we increase the probability of occurrence. However, the N10 value for drug1-AE1 also changes from 4072.8 to 7051.20, which is a big change and dilutes the signal. The reason behind this phenomenon is that some patients take other drugs besides drug1; however, once they have an AE1 event, this event will also be counted into the "other drugs" effect even if the real reason is only drug1.

- The pool of all drugs and all AEs will affect the results greatly. In our simulation study, we only considered 4 drugs and 4 AEs. In this case, we have the same scale of values from N00 to N11. However, in the AERS data, the total number of drugs and AEs are extremely large, and the consequence of this is that the N00 number can be very small compared to N10, N01, and N11. For example, N00 could be just 100, whereas N10, N01, and N11 are all in the millions. This case actually raises a very important question, which is how to choose the pool of drugs and AEs against which to make the comparisons. Is it more reasonable to try to include as many drugs and AEs as possible, or should we only select those drugs and AEs that are known to be related (say because of their chemical content or because of the known classification of events by system-organ class)?

- The probability of AEs appearing as a consequence of taking a specific drug in our simulation ranges from 10% to 90%. However, in the AERS dataset the probability of AEs appearing is very low, say 1%. All the existing approaches we studied fail to detect the disproportion in our simulation when the probability is high. We should direct more attention on how to assess the performance of those approaches and how they compare to each other. The high chance of certain events appearing should be reflected by those methods even in special situations, such as those fabricated in our simulation.

6.3.3 Conclusions from simulation

The main purpose of our simulation was to try to compare and contrast the results that are provided by each of the different methods. Since all the methods have different approaches and modeling techniques, we were hoping that the simulation would shed some light into what those differences are, and maybe some of the advantages and limitations of each. This could provide guidelines on what could be improved or modified to get more significant results.

A comparison of all the disproportionality methods currently in use can be a valuable exercise, given some sort of gold standard.We propose that in order to validate any results that may be obtained using data mining techniques, a reference database with known drug-adverse event combinations is needed. The information needed to construct this database could be obtained from studies published in the literature and from information obtained through clinical trials. To the best of our knowledge, only a few examples exist in the literature of databases constructed for this purpose (e.g. see [9]), and we strongly feel that this approach should have a more widespread use.

7 Conclusions

In this project, we investigated the four most popular signal detection methods in the current literature: PRR, ROR, MGPS, and BCPNN, and highlighted some of their advantages and limitations, as well as points for improvement.

We also developed three novel algorithms for signal detection that incorporate the time factor into the analysis, allowing for a sequential determination of "importance of association" between specific drug-adverse event pairs: one based in percent changes, a parametric approach, and a non-parametric approach. We conducted analyses for three drug-adverse event combinations, namely Avandia & myocardial infarction, Finasteride & sexual dysfunction, and Thiazolidinedione & macular edema. We found that the parametric approach is the fastest to identify a potential signal, but it might be likely to produce more false positives. On the other hand, the nonparametric approach seems to be more conservative (more evidence needs to accumulate in order for it to detect a signal) which may make it more reliable, but it is slower than the parametric approach.

Additionally, we analyzed the Avandia & myocardial infarction pair for a specific period of time, highlighting the importance of stratification on the demographic characteristics of the individuals reporting adverse events. Finally, we did a simulation study to gain insight into the existing methods and to try to provide a starting point for future studies, since we believe that validation of the performance of any data mining algorithm is essential.

8 Future work and recommendations

There are many prospective lines of future work related to the signal detection problem. Some of them we have already outlined in the paper, but here we make some additional recommendations.

The reliability of the dataset is a very important issue to keep in mind. To this respect, we believe that much work can be put into homogenizing the names of the drugs and adverse events, and this can potentially be addressed via language processing techniques. In addition, due to the importance of conducting stratified analyses, more demographic covariates should be collected, as well as information related to the drug doses and exposure times, which could be very valuable in dismissing spurious signals. We also propose including other sources of information, such as results of clinical trials, epidemiological tracking information from the CDC, or drug labels, in order to introduce some level of validation to the voluntary reporting system.

Other directions for future work lie in how to model interactions: both drug-drug interactions, and adverse events with others. Some efforts have been made to introduce drug-drug interactions into existing methods

by expanding the definition of "drug" to drug combinations [5]. There have also been attempts to use logistic regression to discover associations between drugs [12]. But these seem to suffer from the issue of high dimensionality, so maybe clustering methods could be explored to this respect to reduce the dimensionality of the data. In terms of the event associations, a possibility would be to include system organ class (SOC) information into a hierarchical model as prior information.

A promising area of exploration lies in how to incorporate historical information into the modeling, since a lot is learned in each time period and could be potentially used to estimate the background noise for future time points.

Two final thoughts include taking into consideration the way in which the reports are submitted. First, it could be promising to try to model the reporting mechanism and include it as prior information in a Bayesian model, since in this way we take into account the uncertainty regarding the number of exposed individuals. Secondly, weights could be added depending on the reliability of the agent submitting the report (clinicians, patients, manufacturers), since we could have more confidence on the information provided by different sources.

In our opinion, there are several issues that remain problematic. One of them is the absence of a gold standard against which to evaluate the performance of data mining and signal detection methods. Another point for concern is the lack of validation and comparisons of the different methods.

Since there are no best practices or golden standard for signal detection, a great area of opportunity arises for the development of analytical tools, but also for the misinterpretation of their results. It is important to always keep in mind that no algorithm can replace the role of trained physicians, since signal detection requires clinical judgement and knowledge of thresholds, but the methods can serve as initial indicators of the possibility of associations between drugs and adverse events.

9 Appendix

Drug	Event	N00	N10	N01	N11	EBGM	std	IC	std	PRR	std	ROR	std
drug1	AE1	2512.92	7505.46	7560.38	22496.18	1.00	0.01	-0.00	0.01	1.00	0.01	1.00	0.01
drug1	AE2	2503.66	7507.38	7522.04	22541.86	1.00	0.01	-0.00	0.01	1.00	0.01	1.00	0.01
drug1	AE3	2504.28	7509.60	7521.42	22539.64	1.00	0.01	-0.00	0.01	1.00	0.01	1.00	0.01
drug1	AE4	2506.58	7525.06	7519.12	22524.18	1.00	0.01	-0.00	0.01	1.00	0.01	1.00	0.01
drug2	AE1	2500.60	7517.78	7471.34	22585.22	1.00	0.01	0.00	0.01	1.00	0.01	1.01	0.02
drug2	AE2	2488.56	7522.48	7483.38	22580.52	1.00	0.01	-0.00	0.01	1.00	0.01	1.00	0.01
drug2	AE3	2487.06	7526.82	7484.88	22576.18	1.00	0.01	-0.00	0.01	1.00	0.01	1.00	0.01
drug2	AE4	2495.72	7535.92	7476.22	22567.08	1.00	0.01	-0.00	0.01	1.00	0.01	1.00	0.01
drug3	AE1	2516.66	7494.38	7556.64	22507.26	1.00	0.01	0.00	0.01	1.00	0.01	1.00	0.01
drug3	AE2	2521.66	7492.22	7551.64	22509.42	1.00	0.01	0.00	0.01	1.00	0.01	1.00	0.01
drug3	AE3	2522.06	7509.58	7551.24	22492.06	1.00	0.01	0.00	0.01	1.00	0.01	1.00	0.01
drug3	AE4	2493.68	7524.70	7510.32	22546.24	1.00	0.01	-0.00	0.01	1.00	0.01	0.99	0.01
drug4	AE1	2502.16	7508.88	7501.84	22562.06	1.00	0.01	0.00	0.01	1.00	0.01	1.00	0.01
drug4	AE2	2500.88	7513.00	7503.12	22557.94	1.00	0.01	0.00	0.01	1.00	0.01	1.00	0.01
drug4	AE3	2507.28	7524.36	7496.72	22546.58	1.00	0.01	0.00	0.01	1.00	0.01	1.00	0.01
drug4	AE4	2511.18	7507.20	7514.52	22542.04	1.00	0.01	0.00	0.01	1.00	0.01	1.00	0.01

Table 8: Simulation results for CASE I.

Drug	Event	N00	N10	N01	N11	EBGM	std	IC	std	PRR	std	ROR	std
drug1	AE1	1378.54	4072.80	7474.06	22023.86	1.00	0.01	-0.00	0.02	1.00	0.01	1.00	0.02
drug1	AE2	1357.76	4088.02	7349.08	22154.40	1.00	0.01	0.00	0.02	1.00	0.01	1.00	0.02
drug1	AE3	2688.20	8148.98	6018.64	18093.44	0.99	0.01	-0.01	0.01	0.99	0.01	0.99	0.01
drug1	AE4	3299.66	9915.30	5407.18	16327.12	1.00	0.00	0.00	0.01	1.00	0.01	1.00	0.01
drug2	AE1	1352.06	4099.28	7331.90	22166.02	1.00	0.01	-0.00	0.01	1.00	0.01	1.00	0.02
drug2	AE2	1344.94	4100.84	7339.02	22164.46	0.99	0.01	-0.01	0.02	0.99	0.01	0.99	0.02
drug2	AE3	2698.84	8138.34	5985.12	18126.96	1.00	0.01	0.00	0.01	1.00	0.01	1.00	0.01
drug2	AE4	3288.12	9926.84	5395.84	16338.46	1.00	0.01	0.00	0.01	1.00	0.01	1.00	0.01
drug3	AE1	1381.24	4064.54	7471.36	22032.12	1.00	0.01	0.00	0.02	1.00	0.01	1.00	0.02
drug3	AE2	2744.24	8092.94	6108.36	18003.72	1.00	0.01	-0.00	0.01	1.00	0.01	1.00	0.01
drug3	AE3	3348.58	9866.38	5504.02	16230.28	1.00	0.00	0.00	0.01	1.00	0.01	1.00	0.01
drug3	AE4	1359.52	4091.82	7346.34	22151.58	1.00	0.01	0.00	0.02	1.00	0.01	1.00	0.02
drug4	AE1	1361.84	4083.94	7344.02	22159.46	1.00	0.01	0.01	0.02	1.00	0.01	1.01	0.02
drug4	AE2	2705.90	8131.28	5999.96	18112.12	1.00	0.01	0.00	0.01	1.00	0.01	1.00	0.01
drug4	AE3	3278.60	9936.36	5427.26	16307.04	1.00	0.00	-0.01	0.01	0.99	0.01	0.99	0.01
drug4	AE4	1361.22	4090.12	7345.62	22152.30	1.00	0.01	0.00	0.02	1.00	0.01	1.00	0.02

Table 9: Simulation results for CASE II.

Drug	Event	N00	N10	N01	N11	EBGM	std	IC	std	PRR	std	ROR	std
drug1	AE1	2273.06	7051.20	7292.50	22568.32	1.00	0.01	-0.00	0.01	1.00	0.01	1.00	0.01
drug1	AE2	1379.28	4105.66	8489.42	25210.72	1.00	0.01	-0.00	0.02	1.00	0.02	1.00	0.02
drug1	AE3	2765.98	8230.90	7102.72	21085.48	1.00	0.01	-0.00	0.01	1.00	0.01	1.00	0.01
drug1	AE4	3359.40	10019.60	6509.30	19296.78	1.00	0.00	-0.00	0.01	1.00	0.01	0.99	0.01
drug2	AE1	2373.54	6950.72	7566.10	22294.72	1.00	0.01	0.01	0.01	1.00	0.01	1.01	0.01
drug2	AE2	1398.18	4086.76	8541.46	25158.68	1.00	0.01	0.01	0.02	1.01	0.01	1.01	0.02
drug2	AE3	2784.62	8212.26	7155.02	21033.18	1.00	0.00	-0.00	0.01	1.00	0.01	1.00	0.01
drug2	AE4	3383.30	9995.70	6556.34	19249.74	1.00	0.00	-0.00	0.01	1.00	0.01	0.99	0.01
drug3	AE1	1332.50	4152.44	8233.06	25467.08	0.99	0.01	-0.01	0.02	0.99	0.01	0.99	0.02
drug3	AE2	2690.52	8306.36	6875.04	21313.16	1.00	0.01	0.00	0.01	1.00	0.01	1.00	0.01
drug3	AE3	3269.48	10109.52	6296.08	19510.00	1.00	0.00	0.00	0.01	1.00	0.01	1.00	0.01
drug3	AE4	2313.62	7010.64	7497.56	22363.26	0.99	0.01	-0.01	0.01	0.99	0.01	0.98	0.01
drug4	AE1	1374.98	4109.96	8436.20	25263.94	1.00	0.01	0.00	0.02	1.00	0.01	1.00	0.02
drug4	AE2	2755.76	8241.12	7055.42	21132.78	1.00	0.00	0.00	0.01	1.00	0.01	1.00	0.01
drug4	AE3	3366.82	10012.18	6444.36	19361.72	1.00	0.00	0.01	0.01	1.01	0.01	1.01	0.01
drug4	AE4	2364.04	6960.22	7504.66	22356.16	1.01	0.01	0.01	0.01	1.01	0.01	1.01	0.01

Table 10: Simulation results for CASE III.

References

[1] Institute of Medicine (US) Forum on Drug Discovery, Development, and Translation. Emerging Safety Science: Workshop Summary. Washington (DC): National Academies Press (US); 2008. 8, Pharmacovigilance. Available from: http://www.ncbi.nlm.nih.gov/books/NBK4056/

[2] FDA, The Sentinel Initiative: National Strategy for Monitoring Medical Product Safety.

[3] Ahmed, I., et al. (2009). *Bayesian pharmacovigilance signal detection methods revisited in a multiple comparison setting*. Statistics in Medicine **28**, 17741792.

[4] Bate A. (2007). *Bayesian confidence propagation neural network*. Drug Safety **30**, 623625.

Drug	Event	N00	N10	N01	N11	EBGM	std	IC	std	PRR	std	ROR	std
drug1	AE1	2258.06	6999.36	10031.94	30693.88	0.99	0.01	-0.01	0.01	0.99	0.01	0.99	0.01
drug1	AE2	2261.30	6972.52	10076.18	30673.24	0.99	0.01	-0.01	0.01	0.99	0.01	0.99	0.01
drug1	AE3	3650.80	11154.56	8686.68	26491.20	1.00	0.00	-0.00	0.01	1.00	0.01	1.00	0.01
drug1	AE4	4131.48	12555.16	8206.00	25090.60	1.00	0.00	0.00	0.00	1.00	0.00	1.01	0.01
drug2	AE1	2327.12	6930.30	10274.32	30451.50	1.00	0.01	-0.00	0.01	1.00	0.01	1.00	0.01
drug2	AE2	2341.78	6892.04	10259.66	30489.76	1.00	0.01	0.01	0.01	1.01	0.01	1.01	0.01
drug2	AE3	3738.36	11067.00	8863.08	26314.80	1.00	0.00	0.00	0.01	1.00	0.00	1.00	0.01
drug2	AE4	4194.18	12492.46	8407.26	24889.34	1.00	0.00	-0.00	0.00	1.00	0.00	0.99	0.01
drug3	AE1	2284.50	6949.32	10005.50	30743.92	1.00	0.01	0.01	0.01	1.01	0.01	1.01	0.01
drug3	AE2	3627.80	11177.56	8662.20	26515.68	1.00	0.00	-0.00	0.00	1.00	0.00	0.99	0.01
drug3	AE3	4119.64	12567.00	8170.36	25126.24	1.00	0.00	0.01	0.00	1.01	0.00	1.01	0.01
drug3	AE4	2378.34	6879.08	10375.98	30349.84	1.01	0.01	0.01	0.01	1.01	0.01	1.01	0.01
drug4	AE1	2346.24	6887.58	10408.08	30341.34	0.99	0.01	-0.01	0.01	0.99	0.01	0.99	0.01
drug4	AE2	3788.40	11016.96	8965.92	26211.96	1.00	0.00	0.00	0.00	1.00	0.00	1.01	0.01
drug4	AE3	4241.34	12445.30	8512.98	24783.62	1.00	0.00	-0.01	0.00	0.99	0.00	0.99	0.00
drug4	AE4	2293.90	6963.52	10043.58	30682.24	1.00	0.01	0.01	0.01	1.00	0.01	1.01	0.01

Table 11: Simulation results for CASE IV.

[5] DuMouchel W. (1999). *Bayesian data mining in large frequency tables, with an application to the FDA Spontaneous Reporting System*. Am. Stat. **53**, 17790.

[6] Gould, A.L. (2003). *Practical pharmacovigilance analysis strategies*. Pharmacoepidemiology and Drug Safety **12**, 559-574.

[7] Hauben, M. (2003). *A Brief Primer on Automated Signal Detection*. The Annals of Pharmacotherapy **37(7/8)**, 1117-1123.

[8] Hauben, M., Zhou, X. (2003). *Quantitative Methods in Pharmacovigilance*. Drug Safety **26-3**, 159-186.

[9] Hochberg, A.M., et al. (2009). *An Evaluation of Three Signal-Detection Algorithms Using a Highly Inclusive Reference Event Database*. Drug Safety **32 (6)**, 509-525.

[10] Idris, I., et al. (2012). *Association Between Thiazolidinedione Treatment and Risk of Macular Edema Among Patients With Type 2 Diabetes*. Arch Intern Med. **172 (13)**, 1005-1011.

[11] Irwig, M. S. (2012), *Persistent Sexual Side Effects of Finasteride: Could They Be Permanent?* Journal of Sexual Medicine **9 (7)**.

[12] Madigan, D., et al. (2010). *Bayesian methods in pharmacovigilance*. In: J. M. Bernardo, et al. (eds), Bayesian Statistics **9**, Oxford University Press.

[13] Rolka H., et al. (2005). *Using simulation to assess the sensitivity and specificity of a signal detection tool for multidimensional public health surveillance data*. Statistics in Medicine **24(4)**, 551562.

[14] Roux E., et al. (2005). *Evaluation of statistical association measures for the automatic signal generation in pharmacovigilance*. IEEE Transactions on Information Technology in Biomedicine **9(4)**, 518527.

[15] *Avandia (Drug)*, New York Times 'Health' Section article. From http://topics.nytimes.com/top/news/health/diseasesconditionsandhealthtopics/avandiadrug/index.html.