Appendix A

Constants and Conversions

Units presented in this dissertation were the native units of measurement by the manufacturers and instrumentation. Therefore, a chart of conversions has been provided for the reader.

Property	Symbol	Value
Density	ρ	10^3 kg/m^3
Dynamic (absolute) viscosity	μ	10^{-3} kg/(m s)
Speed of Sound	c	1500 m/s

Table A.1: Properties of water

Property		
Pressure	1 mm Hg	1 torr
Pressure	1 mm Hg	133.28 Pa
Pressure	1 mm Hg	1.934x10^{-2} PSI
Pressure	1 mm Hg	0.5353 in H$_2$O
Volume	1 L	10^3 cm^3 (cc)
Volume	1 L	10^{-3} m^3
Volume	1 L	61.02 in^3
Volume	1 in^3	16.39 cm^3 (cc)
Flow rate	1 L/min	1.67x10^{-5} m^3/sec
Flow rate	1 L/min	16.67 cm^3/sec
Flow rate	1 L/min	1.017 in^3/sec
Length	1 inch	2.54 cm

Table A.2: Unit conversions

Glycerol Sol. (% by Mass)	Kinematic Viscosity (cP)
0	1.004
0.5	1.012
1	1.022
2	1.045
3	1.068
4	1.092
5	1.116
6	1.143
7	1.171
8	1.201
9	1.232
10	1.263
12	1.330
14	1.401
16	1.480
18	1.566
20	1.661
24	1.882
28	2.136
32	2.449
36	2.839
40	3.326
44	4.005
48	4.833
52	5.895
56	7.311
60	9.264
64	11.73
68	15.70
72	23.28
76	33.88
80	49.57
84	69.18
88	119.9
92	310
96	624
98	939

Table A.3: Kinematic viscosities of glycerol solution in H_2O concentrations

Appendix B

Rotary Viscous Pump

B.1 Introduction

The limitations of the current technology in left ventricle assist devices (LVADs) led us to explore new design concepts leading to the viscous pump. The most significant improvements in this design are the prospect of not causing hemolysis, scalability, and of its capability of being flexible and thereby more compliant with the human body.

The basic mechanism of the pump consists of two primary parts: the first is a tube, open at both ends, with helical grooves on the inner surface, the second is a smooth shaft inside the tube that is spun by a motor. As the shaft rotates, it drags the fluid with it and the helical grooves guide the flow in a specified direction.

What differentiates this design from an Archimedes pump or a traditional pump is that there are no sharp surfaces moving in the fluid. We anticipate that this will minimize hemolysis. Other benefits of the design include the ability to make it flexible, in turn making it easier to implant and more compliant as an internal device.

B.2 Methods

The prototype of the pump was first made using stereolithography, a technique that creates polymer models from a computer design. Because of the limitations in the resolution of stereolithography, the design used in these experiments used a 1/2-inch

Figure B.1: Photograph of a stereolithography model of the rotary viscous pump.

shaft that was connected to an external brushless motor. The pump was incorporated into a test loop consisting of two pressure transducers located at the inlet and outlet of the pump, a transonic flow transducer, a reservoir of fluid, and an adjustable ball valve. Sections of the loop were made from Tygon 3603 tubing and connected with quick release connectors.

Two sets of data were collected for each case explored. In the first test, the ball valve was left completely open while the angular velocity of the shaft was adjusted by steps of approximately 1000 rpm. Ten samples were taken for each data point over ten seconds. Measurements were taken through a National Instruments DAQ board and recorded using LabVIEW. Data collected included the inlet pressure, outlet pressure, flow rate, and the power consumption of the motor.

In the second test, the angular speed of the shaft was set to the constant rate of 7650 rpm. In this case, however, the opening of the ball valve was adjusted in steps of 15 deg to modify the resistance of the loop. The data were collected in the same manner as the first test.

B.3   Stereolithography Results

A variety of design parameters were modified and tested. To begin, we tested the performance of our initial design with both water and 31% by volume glycerol solution corresponding to a viscosity of 3.082 g/cm-sec, similar to that of blood (figure B.2).

The pump performed better with the glycerol solution by increasing the flow rate for a given angular velocity by approximately 10%. The pressure head at a given flow rate was about 40% greater for the glycerol solution than the water.

For the experiments that followed, the pump with the best results from the previous experiment was chosen and modified. This was because the number of pumps that would have to be produced in order to collect all permutations of the design would have been too large. As a result, all the further experiments were completed with 31% glycerol solution.

The other parameters tested include the number of grooves within the pump (figure B.3), the pitch of the grooves (figure B.4), the depth of the grooves (figure B.5), the length of the pump (figure B.6), and the roughness of the surfaces both on the shaft and in the helices (figure B.7). The gap distance between the shaft and helices was maintained as constant as the accuracy of stereolithography would allow. All the parameters had dramatic effects on the performance of the pump. The grooves, to our surprise, appeared to improve the performance of the pump the shallower they were, despite the decrease in the inflow area. However, the deviation in gap distance may play a critical role in this result. Increasing the number of grooves, as well as increasing the length, improved the performance of the pump with less than linear success. The optimal pitch found was 58 deg from the axis testing at increments of 8 degrees.

B.4   Dimensional Analysis and Similitude

Dimensional analysis was performed to predict the pump's performance at smaller scales. The best performing stereolithography data set was chosen for the analysis. The effective angular velocity (Ω), flow rate (Q), and pressure head (P) were determined by holding the Reynolds number [1] constant. The density and viscosity would be fixed for all size scales. The characteristic velocity and the characteristic length

---

[1] The Reynolds number is defined as $\mathrm{Re} = \frac{\rho V L}{\mu}$ where ρ = density, V = characteristic velocity, L = characteristic length, and μ = dynamic (absolute) viscosity.
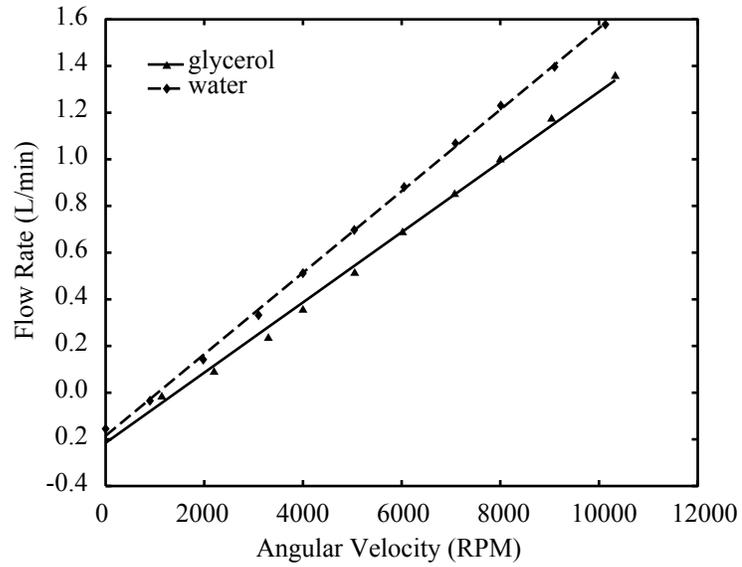
(a) Constant Resistance
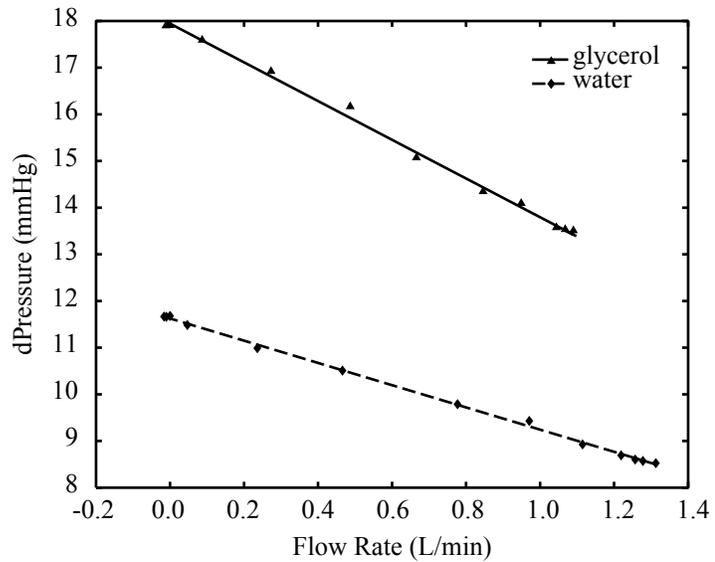
(b) Constant Angular Velocity

Figure B.2: Performance of rotary viscous pump as a function of viscosity.
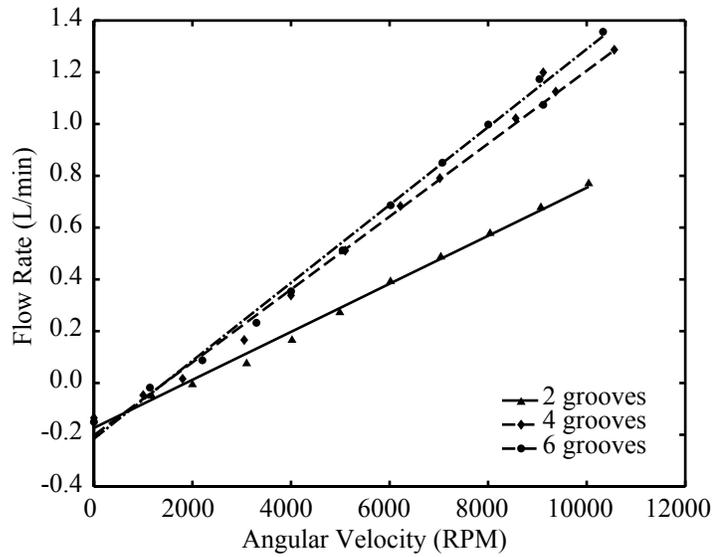
(a) Constant Resistance
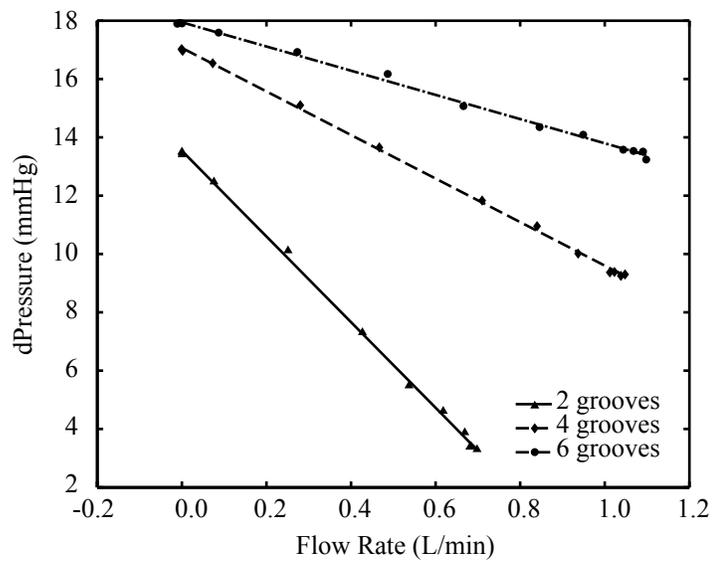
(b) Constant Angular Velocity

Figure B.3: Performance of rotary viscous pump as a function of number of grooves.
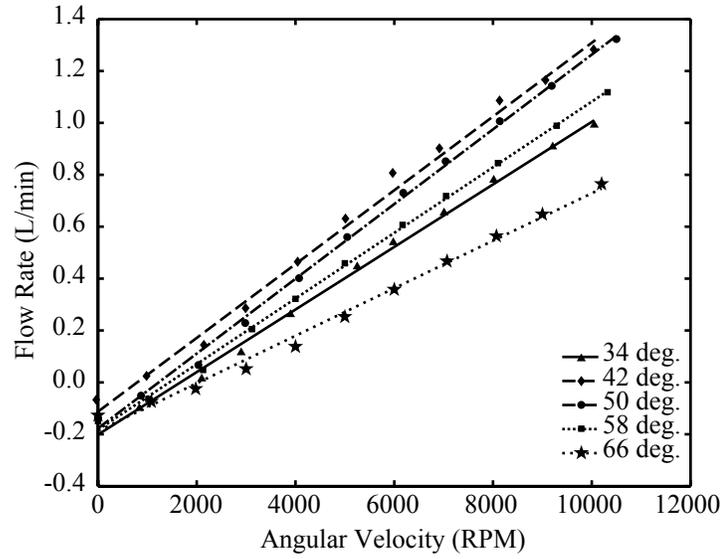
(a) Constant Resistance
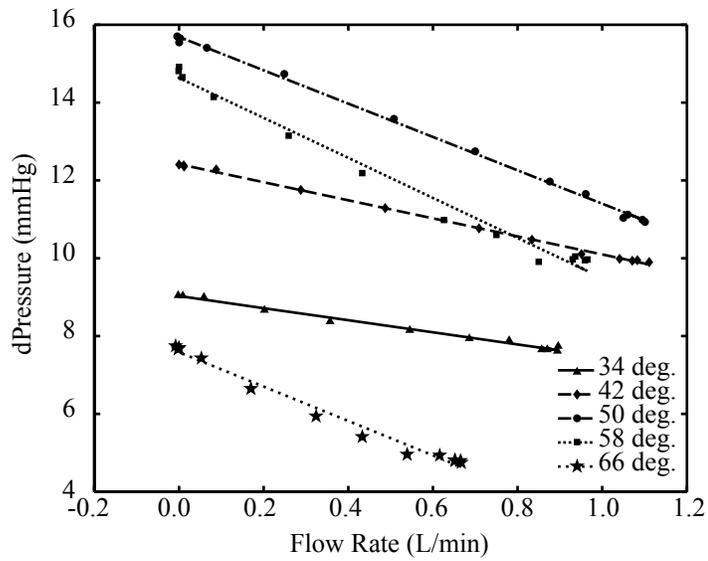
(b) Constant Angular Velocity

Figure B.4: Performance of rotary viscous pump as a function of groove angle.
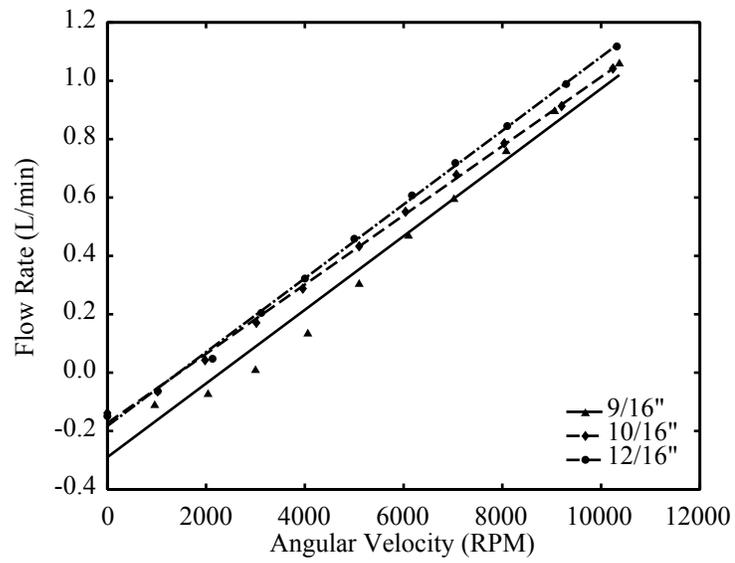
(a) Constant Resistance
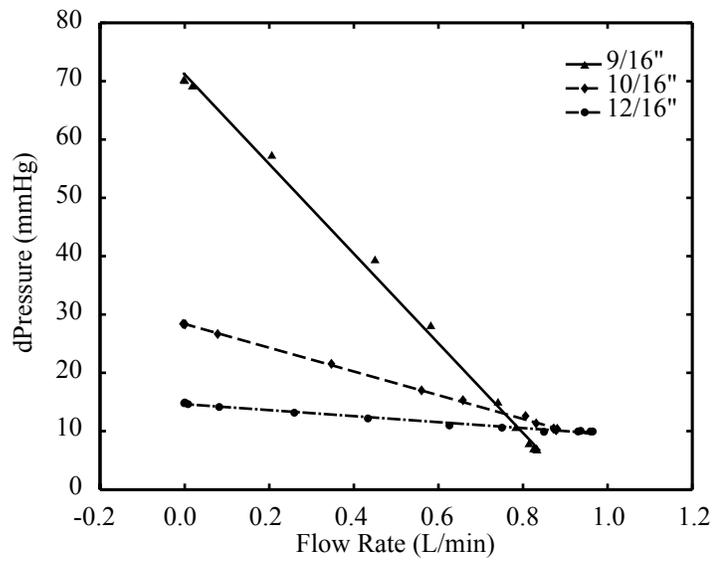
(b) Constant Angular Velocity

Figure B.5: Performance of rotary viscous pump as a function of depth of grooves.
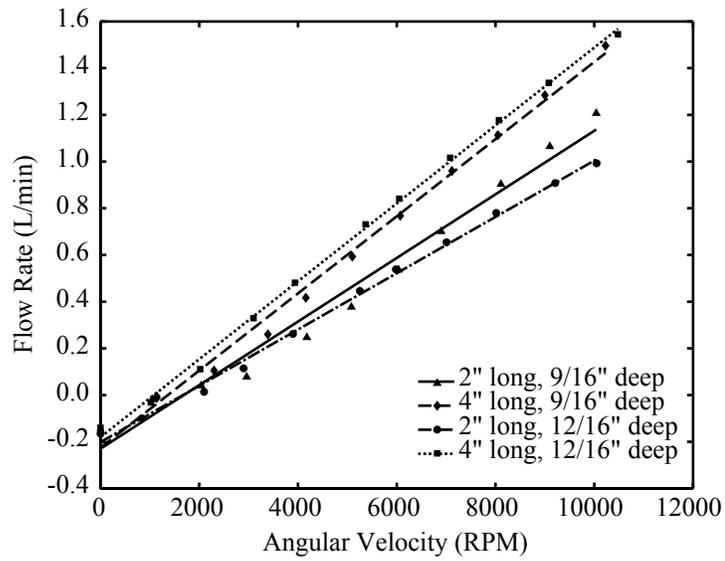
(a) Constant Resistance
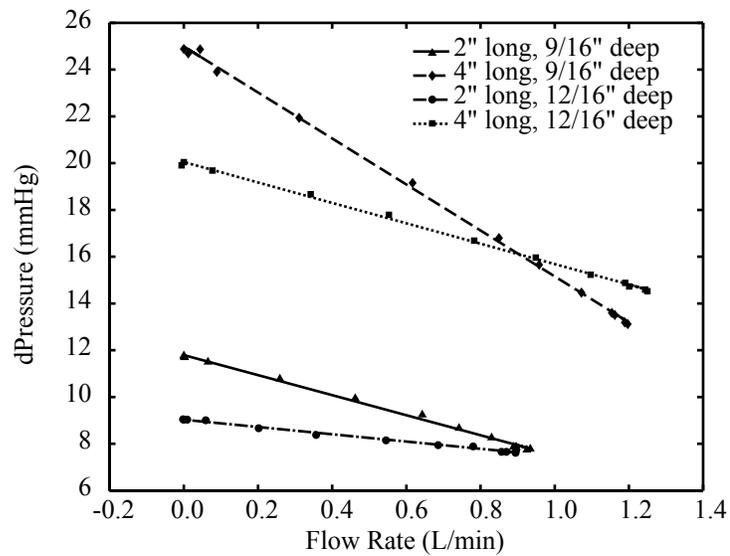
(b) Constant Angular Velocity

Figure B.6: Performance of rotary viscous pump as a function of length of pump.
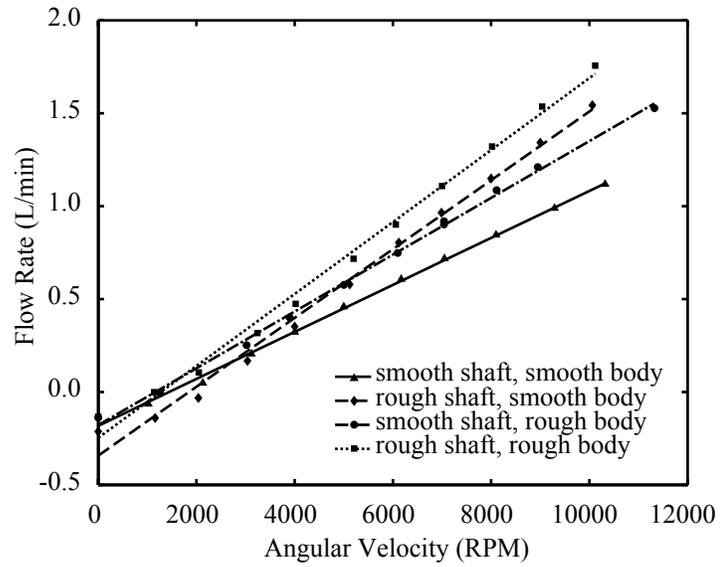
(a) Constant Resistance
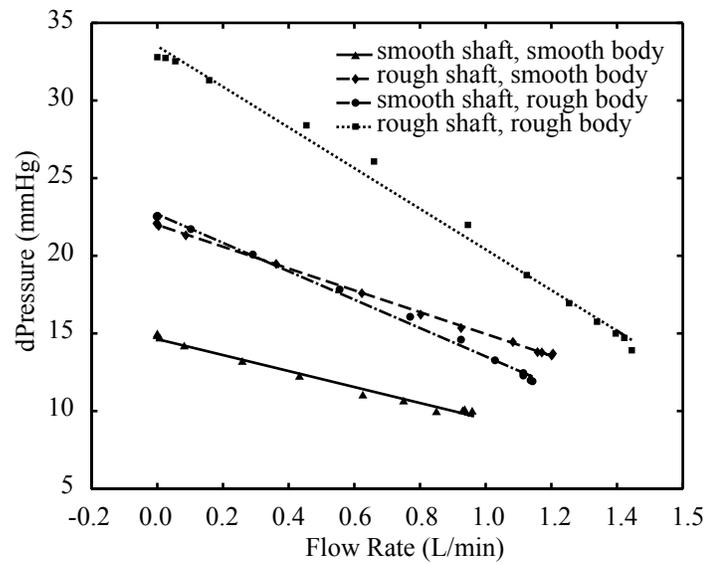
(b) Constant Angular Velocity

Figure B.7: Performance of rotary viscous pump as a function of surface roughness.

chosen were the surface speed and the radius of the internal shaft, respectively. If we define α as the characteristic length of the predicted pump a to that of the known pump b ($\alpha = \frac{L_a}{L_b}$) we find that the effective angular velocity is

$$\Omega_a = \frac{\Omega_b}{\alpha^2}$$

Assuming that the flow rate will be identically proportional to L^2V the effective flow rate becomes

$$Q_a = \alpha Q_b$$

The pressure is determined by

$$P_a = \frac{P_b}{\alpha^2}$$

B.5   Other Manufacturing Techniques

Given the encouraging results of the first round of testing we decided to pursue the scaling of the design to a biologically relevant size that could be implanted through the femoral artery. Because stereolithography does not provide sufficient accuracy, other manufacturing techniques had to be considered. Among those were thermal plastics extrusion, latex dipping, and silicone molding. Thermal plastics extrusion proved to be too expensive to try. Latex dipping was only partially successful. The part could be made, but the latex material was sticky and inaccurate. Also, it took two days to complete one part. Eventually, we used a two-part silicone molding technique that could reliably produce a part in 24 hours.

B.6   Silicone Designs

The first step to the silicone molding process is to design the mold. Our mold was machined from aluminum so that it would be accurate and reusable. The longest length that the interior mold could be made while maintaining a maximum diameter of 1/4 inch was 4 inches. The silicone used was a two part formula, General Electric's
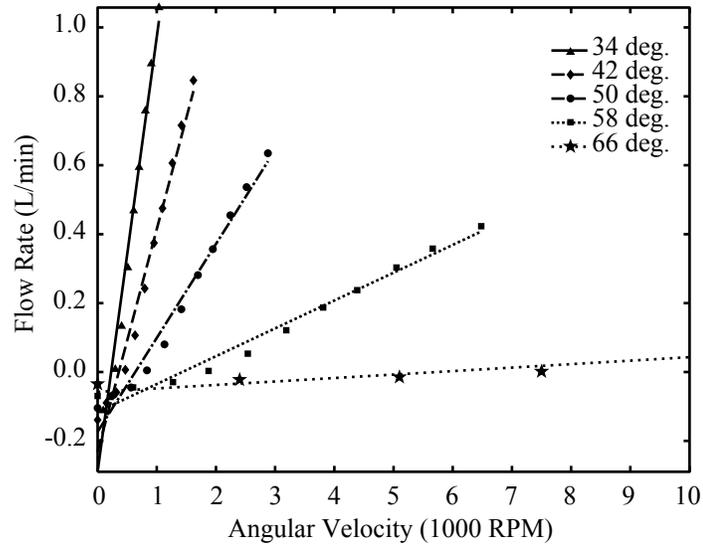
(a) Constant Resistance
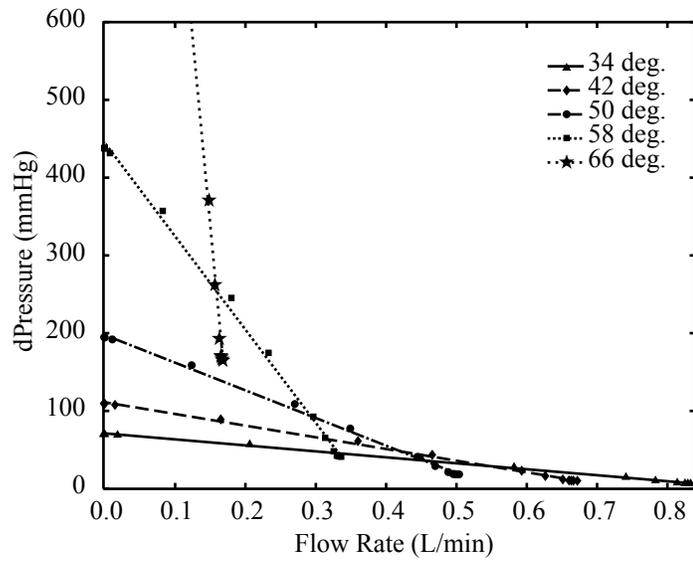
(b) Constant Angular Velocity

Figure B.8: Similitude results based on perfomance of rotary viscous pump.

RTV615. This particular silicone had the necessary combination of characteristics. It was rigid, easily poured before setting, and bio-compatible. In order to create a smooth and consistent texture to the mold, the silicone mixture, before setting, needed to be vacuum-pumped to remove the air bubbles. In addition, a large surface area exposed to the air and a shallow mold help the irregularities settle out during the setting period.

The experiments performed with the new silicone design are analogous to those done on the stereolithography prototypes. First the pump was tested at various rotational speeds up to 10,000 rpm with minimal loop resistance. The inlet and outlet pressures were measured, as well as the flow rate. A second test was performed where the rotational speed of the shaft was held constant and the loop was shut by pinching the Tygon tube closed.

B.7   Silicone Results

The experiments performed on the silicone rubber design showed that scaling of the pump is possible. The geometry for the mold was chosen such that the resulting pump matched as closely as possible to the best performing stereolithography pump. This corresponded to a pitch of 58 degrees and a groove depth of 1/32 inches.

The silicone pump had a flow rate that was linearly proportional to the rotational velocity. The flow rate corresponding to a constant surface velocity of the shaft for the silicone pump was comparable to that of the analogous stereolithography pump. The resistance of the loop was minimal, creating an immeasurable pressure gradient across the pump. However, when the loop was closed, the pressure increased up to 20 mm Hg.

B.8   Acknowledgements

continued collaboration following. In addition, I would like to acknowledge Barbara Hirtz and Ng Kuang (Nathaniel) Chern for their continued work on the rotary viscous pump. Barbara worked with the silicone molded design. Nathaniel used a machined Delrin pump. Both their efforts focused on the hemolysis testing for blood use.

Appendix C

Flow Visualization

C.1 Digital Particle Image Velocimetry

Digital particle image velocimetry (DPIV) is a flow imaging technique that can provide quantitative data about a flow field in two dimensions. The fluid of interest is seeded with small particles that can fluoresce in laser light. A sheet of laser light is shown through the plane of interest and photographed at a known frame rate. The images are then divided into small regions. By computing a cross-correlation between these regions in consecutive frames, the net velocity of that region can be calculated. A number of methods can be used to improve upon the resolution of this technique. However, it remains limited to flows in which the fluid is optically clear and accessible.

C.2 Ultrasound Doppler

Traditional ultrasound images are captured by measuring the time for a longitudinal ultrasound wave to reflect from a density interface. Flow velocities can then be determined by measuring the Doppler shift of the reflected signal [21]. This technique can be used to measure flows only in the radial direction from the ultrasound probe (figure C.1(a)).

(a) ultrasound Doppler

(b) digital ultrasound speckle image velocimetry
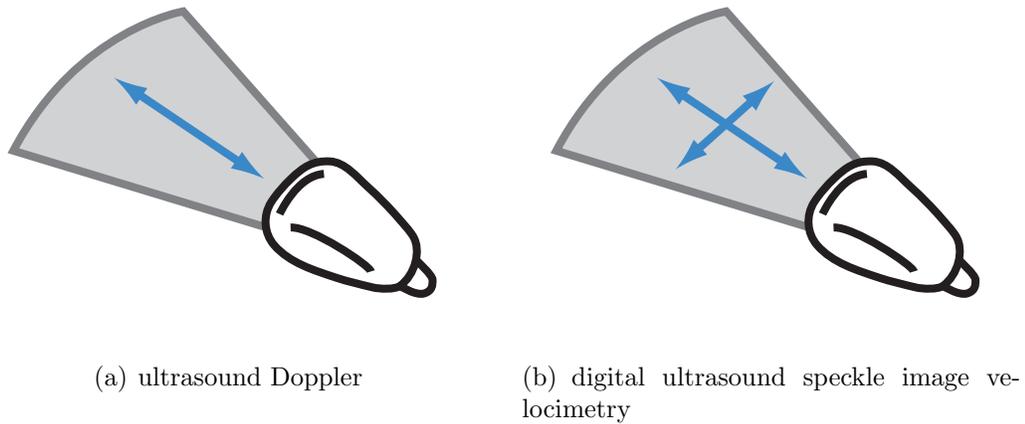
Figure C.1: Ultrasound Doppler and DUSIV velocity directions

C.3   Digital Speckle Image Velocimetry

Digital ultrasound speckle image velocimetry (DUSIV), a hybrid between ultrasound imaging and digital particle image velocimetry, is a technique for obtaining two-dimensional flow field data. The ultrasound can be used to take images of a flow through an optically opaque body. By seeding the flow with particles that reflect ultrasound waves, one can obtain images similar to those used for DPIV (figure C.1(b)).

There remain a number of critical differences in ultrasound images that distinguish them from DPIV:

- The images acquired by ultrasound are captured in polar coordinates. This results in non-uniform resolution of the image when transformed into Cartesian coordinates.

- The individual particles are not resolved in the ultrasound images. Instead, regions in the flow where the density of particles is higher collectively reflect the ultrasound and are realized as speckles in the image.

- The pixels of the image are captured sequentially. This means that the time at which one pixel of the image was taken is not the same as the time another pixel was taken. However, the time step between any pixel in one frame and the same pixel in another frame should be constant for all pixels.

Though the implementation of DPIV and DUSIV may vary from software package to package, the math is essentially the same. Two images are acquired at a time difference of τ. The image is broken down into smaller interrogation regions. The corresponding interrogation regions between the two images are compared by computing the two-dimensional cross correlation of the intensity. This is equivalent to computing the convolution of the two images in frequency space:

$$\Phi_{fg}(x, y) = \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} F(\xi + x, \zeta + y)G(\xi, \zeta)d\xi d\zeta$$

Here, $\Phi_{fg}(x, y)$ represents the cross-correlation function of the intensities of the regions f and g whose Fourier transforms are F and G. In reality, the images are finite in size (a, b), so the cross correlation need only be computed for the width and height of the region. The maximum of the cross-correlation function is then determined.

By knowing the time τ between the images captured and the maximum correlation of the two regions, a local velocity can be inferred. This is performed for regions covering the entire image and results in a velocity vector field.

DUSIV can be used to image flow through any material that transmits ultrasound. However, the probe must be in sonic contact with the body that it is imaging. This can be accomplished by using a transmitting gel, or by submerging the object of interest and the probe underwater.

Appendix D

Edge Detection Documentation

I have written, with the support of Dave Benson, a program to detect the walls of the elastic tube as imaged by the ultrasound machine. The software relies of a set of assumptions about the image to simplify the task:

- There will be two distinct walls, approximately horizontal in the images.

- The walls appear brighter than the background.

- The only other artifacts in the image are noise and speckles.

D.1   Algorithm Overview

- The original, grayscale, polar coordinate image is read from a file into an array containing the pixel data (figure D.1(a)).

- The image is box-blurred in the vertical direction to reduce noise (figure D.1(b)).

- The blurred image is searched for local maxima. These points have intensity values higher than the points directly above and below and have an overall intensity at least half the maximum intensity in the image (figure D.1(c)).

- These local maxima are linked into chains by their nearest neighbor within a search area (figure D.1(d)).

- The number of chains above a minimum length is counted.

(a)　　(b)　　(c)　　(d)　　(e)　　(f)
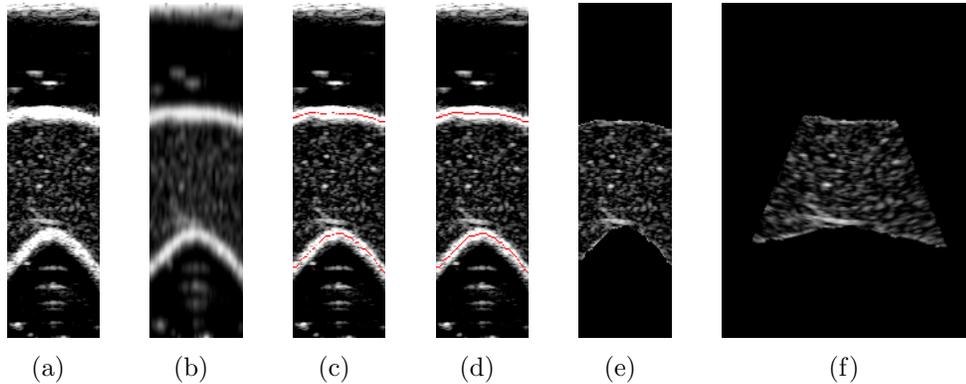
Figure D.1: Processing steps of the edge-detection software (left to right: original image, box-blurred, maxima points located, maxima points linked into a chain, original image masked, coordinates converted from polar to Cartesian)

- The process is repeated by successively blurring the image until there are only two chains that match the minimum length criteria. Those chains represent the edges of the tube.

- The original image is then masked on the walls and outside the tube based on the edge locations found (figure D.1(e)).

- The masked image is transformed from polar coordinates to Cartesian coordinates for use with the DUSIV software (figure D.1(f)).

D.2 Using the Program

To compile the program, one must have a C compiler, glib 1.2, and a few other C libraries. Use the command:

```
prompt: gcc -o edgefind edgefind.c `glib-config --cflags --libs` -lm
```

Once compiled the program can be run in a shell by typing:

```
prompt: edgefind < input.pgm > output.ppm
```

The program will use the file, `input.pgm` as its input and `output.ppm` as the output file. In order to perform this task on many bitmap files, one can use the

shell script `do-all` on any computer running a bourne shell. This script requires imagemagick to convert BMP to PGM files. It then executes the ultrasound program on all the PGM files and creates PPM output files with the same name. Finally, it executes the program `ppm2raw` to convert the ppm files to raw files with the width and height information appended to the file name. This prepares the file for use with the DPIV software.

D.3   Global Constants and Variables

`CHAIN_SEARCH_WIDTH` is the width to the right or left of a pixel in which to search for another chain pixel.

`CHAIN_SEARCH_HEIGHT` is the the height to the top or bottom of a pixel in which to search for another chain pixel.

`CHAIN_LENGTH_MIN` is the minimum length a chain of points must be before it can be considered as an edge.

`HALFTHICK` is half the thickness of an edge, in pixels, to erase.

`N_PRE_GUASSIANS` is the number of times to box-blur the original image before looking for any chains.

`ANGLE_INC` is the angle, in radians, that each column represents.

`BLACK` is the color black defined in RGB, in octal.

`WHITE` is the color white defined in RGB, in octal.

`int width, height` is the width and height of the original image in pixels.

`int w_conv, h_conv` is the width and height of the image after it is converted from polar to cartesian coordinates.

`guint8 *orig` is a one-dimensional array containing the intensity values of all the pixels in the original image.

`guint8 *data` is a one-dimensional array corresponding in size to the original image as it is being blurred and manipulated.

`int *chain_data` is a one-dimensional array corresponding in size to the original image of the points marked as chains. When the value is -1, the point belongs to an unknown chain. When the value is 0, the point does not belong to any chain. When the value is positive, the point belongs to the chain of that value.

`guint8 *conv_data` is a one-dimensional array containing the intensity values of all the pixels in the image converted from polar to Cartesian coordinates.

D.4   Function Overview

`static void box_columns()` will box-blur an image in the vertical direction by averaging each pixel with its upper and lower neighbor.

`static void locate_maxima ()` will locate intensity maxima based on the following criteria: it must have an intensity at least half the maximum intensity in the image, its lower and upper neighbors must have a lower intensity. All the points that are found meeting this criteria are stored in `int *chain_data`.

`static int find_closest_nonzero (int *r, int cx, int cy, int w, int h)` searches an array `*r` containing chain data around a pixel located at x=`int cx`, y=`int cy` for values that are non-zero in a search area that is $2w + 1$ wide and $2h + 1$ tall.

`static gboolean find_chains ()` will search through `int *chain_data` for maxima points, connect and label them as chains based on the algorithm of `find_closest_nonzero`. It will also determine the length of each chain. It will return true if there are exactly two chains that are equal to or longer than `CHAIN_LENGTH_MIN`.

`static void link_chain()` joins the nearest horizontal points within a chain with straight lines to form a contiguous chain of pixels.

`static void mask_data ()` determines the upper and lower chain by averaging the vertical position of all the points in the chain. It then turns all the points above the top chain and all the points below the bottom chain of the original image black. It also turns `HALFTHICK` number of points below the top chain and above the bottom chain of the original image black to mask out the thickness of the elastic tube edges.

`static void scan_convert(double theta)` converts the masked, polar coordinate image into Cartesian coordinates. The pixel height of the image remains the same while the width is adjusted based on the total angle that was scanned. To do this, the position of each pixel in the new converted image in the old coordinates is calculated. A linear interpolation of the intensities of the surrounding pixels of the polar coordinate image is used as the intensity for the new pixel.

`int main ()` is, as its name suggests, the main part of the program.

Bibliography

[1] AUERBACH, D., MOEHRING, W., and MOSER, M., "An analytical approach to the Liebau problem of valveless pumping," *Cardiovascular Engineering: An International Journal*, vol. 4, no. 2, pp. 201–207, 2004.

[2] BORZI, A. and PROPST, G., "Numerical investigation of the Liebau phenomenon," *Zeitschrift für angewandte Mathematik und Physik*, vol. 54, no. 6, pp. 1050–1072, 2003.

[3] FIELD, S. and DRZEWIECKI, G. M., "Dynamic response of the collapsible blood vessel," in *Analysis and Assessment of Cardiovascular Function*, pp. 277–296, Springer, 1998.

[4] FISHMAN, M., "Fashioning the vertebrate heart: Earliest embryonic decisions," 1997.

[5] FOROUHAR, A., HICKERSON, A., LIEBLING, M., FRASER, S., DICKINSON, M., and GHARIB, M., "Personal communication: The embryonic zebrafish heart as an impedance pump," 2005.

[6] GROTBERG, J. B. and JENSEN, O. E., "Biofluid mechanics in flexible tubes," *Annual Review of Fluid Mechanics*, vol. 36, pp. 121–147, 2004.

[7] HANSEN, F., MANGELL, P., SONESSON, B., and LANNE, T., "Diameter and compliance in the human common carotid-artery — variations with age and sex," *Ultrasound in Medicine and Biology*, vol. 21, no. 1, pp. 1–9, 1995.