UNIVERSITY OF CALIFORNIA, SAN DIEGO

The Effects of Non-Newtonian Fluids on Purcell's Scallop Theorem

A thesis submitted in partial satisfaction of the requirements for the degree Master of Science

in

Engineering Sciences (Mechanical Engineering)

by

Dor Yisrael Ashur

Committee in charge

Eric Lauga, Chair
Thomas Bewley
Nathan Delson

2010

The Thesis of Your Full Legal Name is approved and it is acceptable in quality and form for publication on microfilm and electronically:

_____

_____

_____

Chair

University of California, San Diego

2010

TABLE OF CONTENTS

List of Figures

List of Tables

List of Symbols

Symbol	Unit	Name
$\dot{\gamma}$	s^{-1}	Shear rate
λ	s	Maxwell relaxation time
μ	Pas	Dynamic viscosity
μ_o	Pas	Zero shear viscosity
ν	m^2/s	Kinematic viscosity
P	kg/m^3	Density
σ	Pa	Stress
τ	Pa	Shear Stress
ω	rad/s	Angular frequency
De	-	Deborah Number
F	N	Force
G_P	Pa	Plateau modulus
G‘	Pa	Storage modulus
G“	Pa	Loss modulus
I	$kg \cdot m^2$	Inertia
L	m	Length
M	kg	Weight/Mass
N	-	Power Law Constant
N	-	Moles
R	m	Radius
Re	-	Reynolds number
P	Pa	Pressure
T	s	Time
T	Nm	Torque
U, u	m/s	Fluid Velocity
V	m/s	Component Velocity
V	m^3	Volume

Acknowledgements

It has been a great pleasure to work with Maja Stoevhase, with whom I jointly designed and built the snail robot and autonomous swimmer described in Sections 2.1 and 3, and performed rheological tests on the corresponding fluids in those experiments. I wish her the best of luck in her endeavors.

I would like to thank the UCSD department of Mechanical Engineering Staff, who have helped me problem-solve and fabricate the components of these experiments: Tom Chalfont, Chris Cassidy, Steve Roberts. I am also grateful to the Scripps Institute of Oceanography Machine shop for access to build many of the tethered robot components.

The feedback from Professor Tom Bewley and his research group is also very much appreciated, and was key to many components of these projects

This short paragraph cannot describe how very grateful I am to Dr. Nathan Delson for the design, research, teaching and career mentoring throughout my tour at UCSD.

Finally, it has been an honor to work with my advisor, Professor Eric Lauga, who presented me with the amazing opportunity to continue my education and research at UCSD, and whose support and instruction are very much appreciated.

ABSTRACT OF THE THESIS

The Effects of Non-Newtonian Fluids on Purcell's Scallop Theorem

by

Dor Yisrael Ashur

Master of Science in Engineering Sciences (Mechanical Engineering)

University of California, San Diego, 2010

Professor Eric Lauga, Chair

Many common fluids such as blood, oils and shampoos exhibit non-Newtonian properties, and low Reynolds flow is ubiquitous in micro-scale swimmers such as flagellum and bacteria. Purcell's Scallop Theorem states that locomotion via a single time-independent degree of movement through low Reynolds flow is impossible as the effects of momentum are negligible. Experimentation was performed to prove that propulsion, via a singular rotational, or linear degree-of-freedom, is possible by employing the normal stresses and shear thinning capabilities of non-Newtonian fluids, respectively. This study also discusses non-Newtonian fluid rheometry, as well as the design and control of the swimming robots.

Introduction

One degree, reciprocal propulsion is impossible in Newtonian fluids at low Reynolds numbers, per Purcell's Scallop Theorem, which proves that a single degree, time-independent motion cannot create locomotion. This thesis posits that propulsion resulting from such motion is possible if the fluid is viscoelastic and non-Newtonian.

The physics of low Reynolds locomotion is pervasive, yet relatively unknown; micro-scale swimmers, such as flagellum and bacteria swim at scales and in thick fluids via viscous, rather than the inertial forces that drive the commonly encountered high Reynolds locomotion of fish and birds. Some everyday fluids such as blood, shampoo, Silly Putty® and some oils, exhibit elastic properties and are thus, non-Newtonian and cannot easily be modeled by the Navier-Stokes equations.

In addition to advancing the field of non-Newtonian fluid dynamics by experimentally proving the theoretical hypotheses of one degree, reciprocal locomotion, these experiments also serve to initiate the design of commercial products, which would transport sensors and payloads through such fluids and at such small scales. A medical device to transport medicine, micro tools or cameras through non-Newtonian blood could be designed, based off the normal stress experiments in this paper. Finally, a robot designed to traverse and clean or monitor the surfaces of oil spills can be modeled off the shear thinning experimentsin this thesis.

1. Theory

This section presents the relevant background information for the study of non-Newtonian fluids, low Reynolds flow, and rheology.

1.1. Newtonian Fluid Dynamics

The viscosity of a Newtonian fluid is independent of time, strain, and of any derivatives or integrals of strain over time [Harris]. The shear stress experienced by a fluid in motion can be calculated from Newton's Law of Viscosity.

$$\tau_{ij} = \left(-p + \frac{2}{3}\mu\nabla \cdot u\right)\delta_{ij} + \mu\left(\frac{\partial u_i}{\partial u_j} + \frac{\partial u_j}{\partial u_i}\right) \tag{1}$$

where δ is the Kronecker delta. For one-dimensional flow, the shear stress equation can be reduced to:

$$\tau_{yx} = \mu\left(\frac{du}{dy}\right) = \mu\dot{\gamma}_{yx} \tag{2}$$

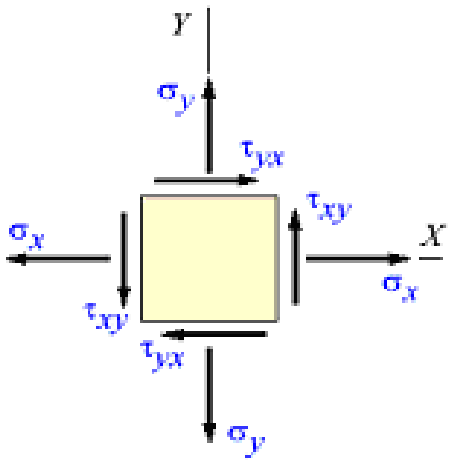

Figure 1.1: Shear and normal stress vector definitions

Incompressible Newtonian fluids also satisfy the Navier-Stokes equations [Cohen]:

$$\rho\left(\frac{\partial}{\partial t} + u\cdot\nabla\right)u = -\nabla p + \mu\nabla^2 u \tag{3}$$

$$\nabla\cdot u = 0 \tag{4}$$

1.2. Non-Newtonian Fluid Dynamics

The relationship between the shear stress and shear rate is nonlinear in Non-Newtonian fluids, and thus Newton's Law of Viscosity, and the Navier-Stokes equations, in Equations (3) and (4) above, are invalid.

1.2.1. Time-independent and Time-Dependant Fluids

The power law model is used to calculate the shear stress for time-independent one-dimensional flow a non-Newtonian fluid.

$$\tau_{yx} = k\left(\frac{du}{dy}\right)^n = k\,(\dot{\gamma}_{yx})^n \tag{5}$$

Most non-Newtonian fluids display an inverse relationship between viscosity and strain rate, as seen in Figure 1.2, where n is less than one, and are termed Pseudoplastics, or shear thinning. Conversely, fluids with an n greater than one become more viscous with rising shear rates and are termed Dilatant, or shear thickening. For n=1 the formula reduces to Equation 2 and the fluid is Newtonian [Harris].

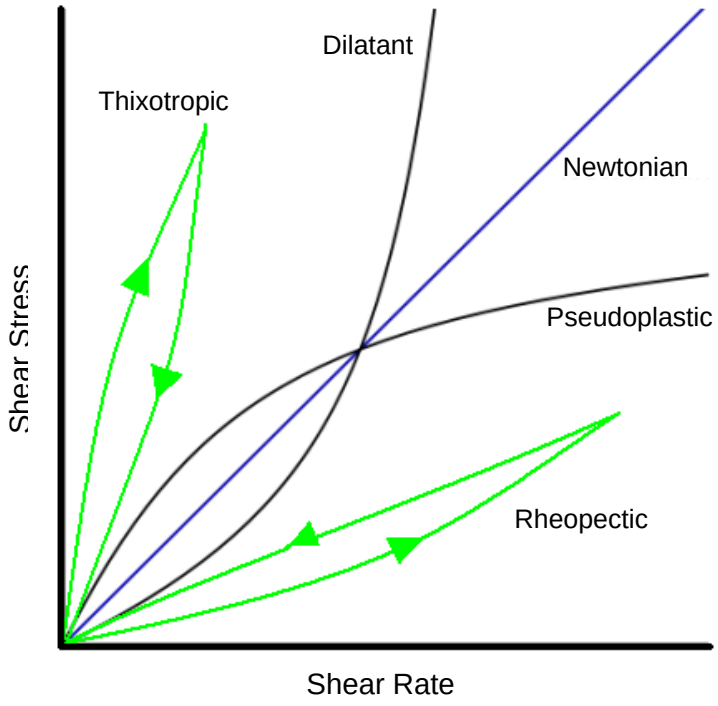

Figure 1.2: Time-independent and time-dependant non-Newtonain Fluids

Thixotropic and Rheopectic fluids, display increasing or decreasing values of n, respectively, from internal tearing or building over time [Tanner].

1.2.2.Viscoelastic Fluid Dynamics

Though all solids and liquids exhibit some, though often negligible, viscous and elastic properties, a viscoelastic, or memory fluid both flows and recoils over similarly substantial time scales.

The elasticity of a fluid can be quantified by measuring the relaxation time (λ), in which a viscoelastic fluid returns to an equilibrium geometry and stress. The non-dimensional Deborah number is the ratio of a fluid's response time to the overall process time, which quantifies the significance of elasticity to the flow, and thus the degree to which it is non-Newtonian [Chhabra].

$$De = \frac{\lambda v}{L} = \frac{\lambda}{t} \tag{6}$$

The combined viscous and elastic properties cannot be fully described by the Newtonian Navier-Stokes equations, are thus approximated as a spring and dashpot in series by the Maxwell Model [Tanner].

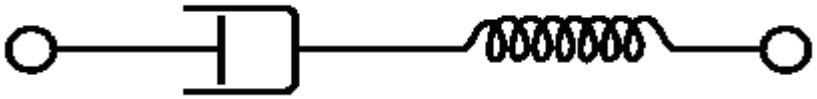

Figure 1.3: Maxwell Model with spring and dashpot

Hooke's Law describes the physics of an elastic spring in Equation (7), and Newton's Law of Viscosity in Equation (8) describes the reaction of a dashpot to stress

$$\gamma_s = \frac{-\tau}{G_p} \tag{7}$$

$$\dot{\gamma}_d = \frac{\tau}{\mu} \tag{8}$$

Adding the two in series, where the strains of the attached components are equivalent, and substituting the time constant, $\lambda=\mu/G_p$, yields:

$$\mu \dot{\gamma} = \tau + \lambda \dot{\tau} \tag{9}$$

The storage modulus (G') and loss modulus (G") quantify a fluid's stored elastic energy and viscosity dissipated heat energy, respectively, as a function of angular frequency. Purely elastic solids have a G"=0, whereas purely viscous Newtonian fluids have a G'=0.

$$G'(\omega) = G_P \frac{(\omega\lambda)^2}{(1 + (\omega\lambda)^2)} \tag{10}$$

$$G''(\omega) = G_P \frac{\omega\lambda}{(1+(\omega\lambda)^2)} \tag{11}$$

Equation (11) can be used to solve for the plateau modulus (G_P), which is equal to the maximum value of G" and the intercept of the G' and G" curves:

$$\frac{G''}{G_P} = \frac{\omega\lambda}{(1+(\omega\lambda)^2)} \tag{12}$$

Differentiating and setting equal to zero to determine the maximum yields:

$$0 = \frac{(1-(\omega\lambda)^2)}{(1+\omega\lambda^2)^2} \tag{13}$$

Solving Equation (13) above, $\omega\lambda = 1$, or $\lambda = 1/\omega$ which, when substituted into Equation (11) , shows the relationship between the plateau modulus and the zero shear viscosity [Varagnat]:

$$G_P = 2G'' = \mu_0/\lambda \tag{14}$$

A cone-plate rheometer, shown in Figure 1.4, is used to measure a fluid's response to shear strain.

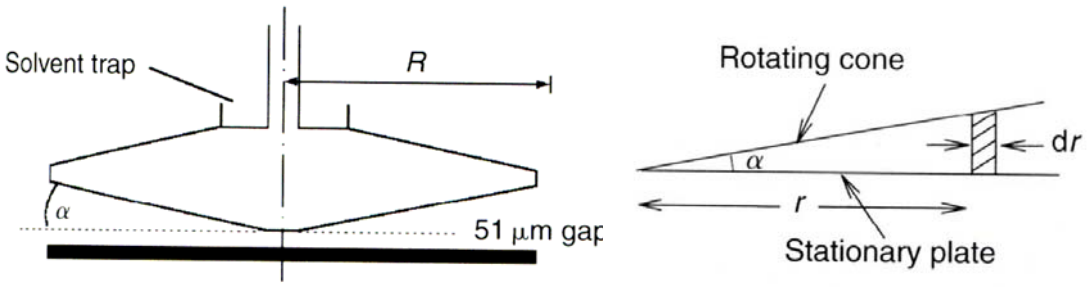

Figure 1.4: a) Cone-Plate rheometer schematic, b) Rheometer boundaries

The torque that the fluid enacts on the cone's bounding surfaces at r = r and r = r + dr, is calculated to determine the shear rate.

$$T=\int_{R}^{0} 2\pi r^2 \tau dr \tag{15}$$

For a constant shear stress value τ, the torque integrates as:

$$T = \frac{2\pi R^3 \tau}{3} \tag{16}$$

The velocity gradient between the stationary fluid, at the plate's surface, and the rotating fluid, in contact with the cone, imposes a shear rate $\dot{\gamma}$, which is independent of the radius [Chhabra]:

$$\dot{\gamma} = \frac{r\omega - 0}{r \tan\alpha} = \frac{\omega}{\tan\alpha} \tag{17}$$

1.2.3.Experimental non-Newtonian Solutions

Micellar solutions, which are currently used to alter the viscoelastic properties of commercial fluids such as paints and inks, contain hydropolar molecules called surfactants, which self arrange into spherical, lamellar or wormlike groupings, as seen in Figure 1.5, that stretch or break as they entangle, to generate elasticity. The viscosity and elasticity of non-Newtonian, micellar solutions are dependant on the concentration of micellar salts [Bhardwaj]

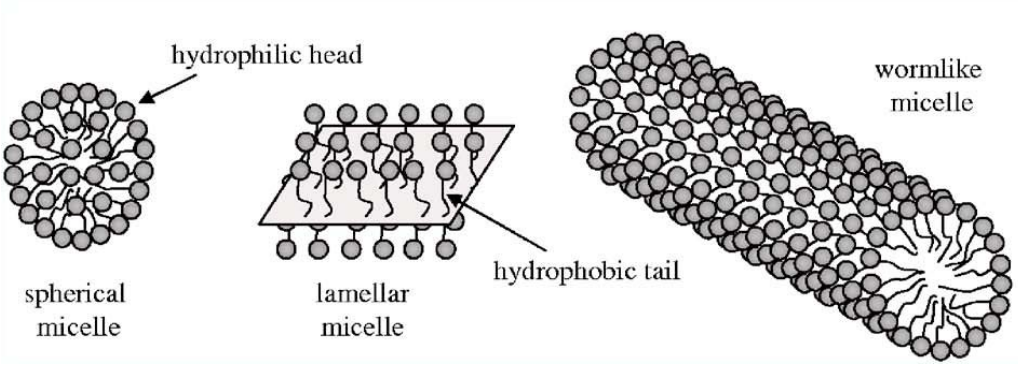

Figure 1.5: Arrangements of surfactantants in micellar solutions

Though uniform micellar solutions can be approximated by a single-mode Maxwell Model, shear-thinning shampoos, which display non-Newtonian properties as they lather, require averaging multiple relaxation times over multiple relaxation modulii per Equation (18).

$$\lambda = \frac{\sum \lambda_i G_i}{\sum G_i} \tag{18}$$

In most soaps, Sodium and Potassium salts form long chains of polar fatty acids, which organize alike surfactants micellar solutions. Similarly, the platelets, red cells and white cells in blood, interact and entangle suspension to impart elasticity.

1.3. Low Reynolds Number Flow

Flows defined by a Reynolds number much less than unity, are driven primarily by viscous drag, rather than inertial forces.

$$\mathrm{Re} = \frac{VL}{\nu} = \frac{\omega r^2}{\nu} \tag{19}$$

In such cases, where for example, a small object traverses slowly through a viscous fluid, the Navier-Stokes equations reduces to the time-independent Stokes equation,

$$0 = -\nabla p + \mu \nabla^2 u \tag{20}$$

For flow past a stationary body, the no slip boundary condition can be applied at the surface, where U is the free stream velocity and u_s is the surface velocity of the swimmer.

$$u_0 = \mathrm{U} + \omega(\mathrm{t}) \times x + u_\mathrm{s} \tag{21}$$

Thus, for a swimmer that deforms periodically for locomotion, a directional reversal of any motion sequence will only result in a reversal of swimming direction, as Stokes Equation is time-reversible, and thus in zero net locomotion [Lauga].

2. Normal Stress - Elastic Swimmer Robot

The "elastic swimmer", with a single rotational degree of freedom, was designed to display the effects of normal stress on locomotion though non-Newtonian fluids. A cone control surface, which rotates relative to a stationary cup, was devised to employ the normal force that results when the tangled polymer strands in non-Newtonian fluids are sheared, towards propulsion. Three robot iterations were designed and fabricated to focus on particular propulsion mechanisms and increase swimming stability.

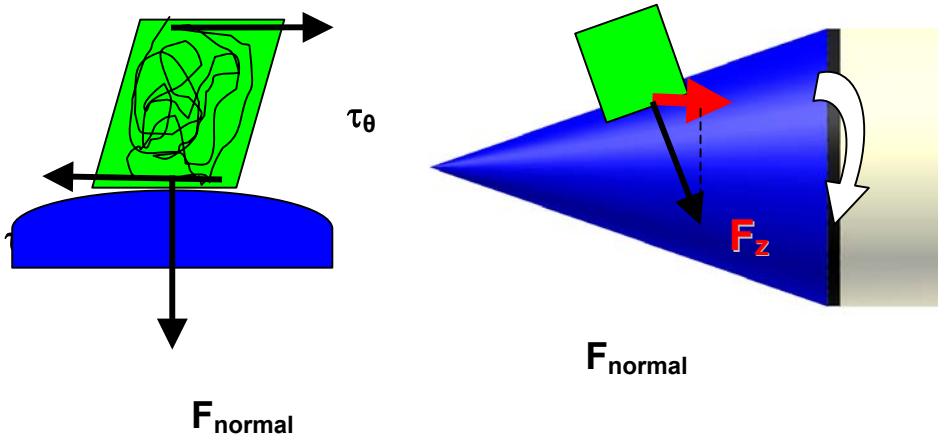

Figure 2.1: Description of normal shear propulsion

The initial, autonomous swimmer was designed to isolate the robot in its media so that no external forces from stabilization fixtures or measuring devices hinder the swimmer's movement. However, the capacious tank and fluid volume required to test such a large self-contained robot, with internal motor, power source and measuring equipment, economically restrained testing to a few inexpensive fluids. Thus, a smaller "tethered swimmer" robot was designed and built, with an external power source and transmission to reduce the volume of fluid required for each test. A second, more robust, version of the tethered robot was constructed to test the swimmer's propulsion at high speeds and torques.

2.1. Autonomous Swimmer Experiment

The autonomous swimmer is propelled by a single unconstrained rotational degree-of-freedom swimmer in Non-Newtonian fluids. The swimmer was designed in Autodesk Inventor and fabricated at the Undergraduate Machine Shop in the department of Mechanical Engineering at the University of California at San Diego. The swimmer consists of a cone propeller and a cup, which houses the motor, battery, electronics and communications devices.

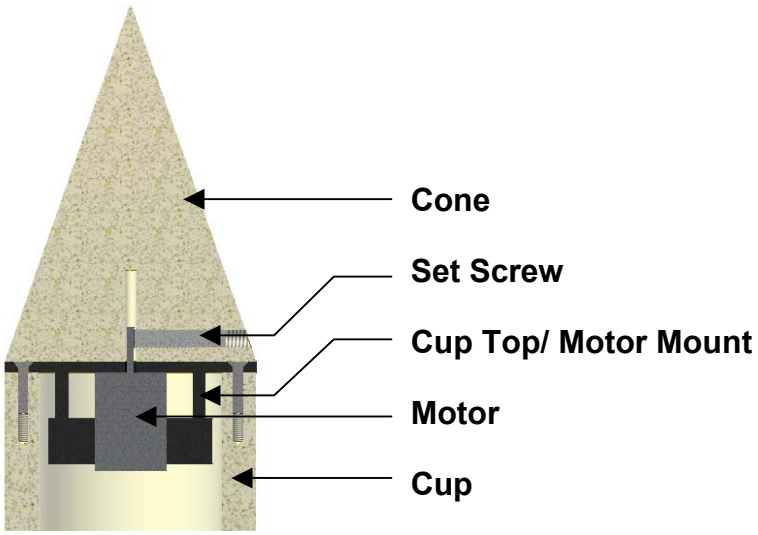

Figure 2.2: CAD model of swimmer assembly

2.1.1.Autonomous Swimmer Design

The cup, 14.5 cm high and 7 cm in diameter, was designed to house the motor, battery, motor controller, encoder, bluetooth, transmitter and ballast. The cup and cone were manufactured of Delrin plastic for its machinability, price, inertness, and availability. An o-ring imbedded into the top rim of the cup, as well as four tapped holes, seen in Figure 2.3, waterproof the compartment.

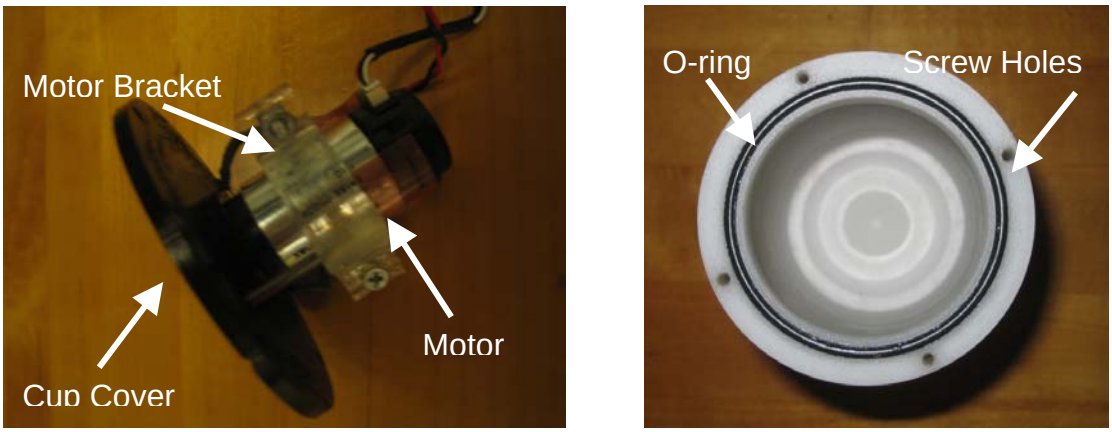

Figure 2.3: a) Motor bracket and cup cover b) Cup with O-ring

The 80-degree, 10 cm high, cone with same diameter as the cup, includes a center hole and perpendicular set screw for inserting and securing the motor‘s shaft. The motor mount plate, which was fabricated from Acrylic using the LaserCAM at the Undergraduate Design Studio, both encloses the cup and fixes the motor and accompanying electronics. The swimmer is kept neutrally buoyant and plumb through added ballast, in the form of lead weights and fasteners, and with padding, that evenly distributes the swimmer’s weight.

The 18.5cm x 18.5cm x 45.5cm 15L tank, seen in Figure 2.4, was designed to securely hold enough fluid to allow the swimmer to vertically traverse a distance of its body length, and maintain a half-diameter side clearance. The tank was built of Acrylic for its opacity, machinability, and strength and was cut using a LaserCAM. The tank was sealed using clear and black RTV Silicone to prevent leakage and reflection.

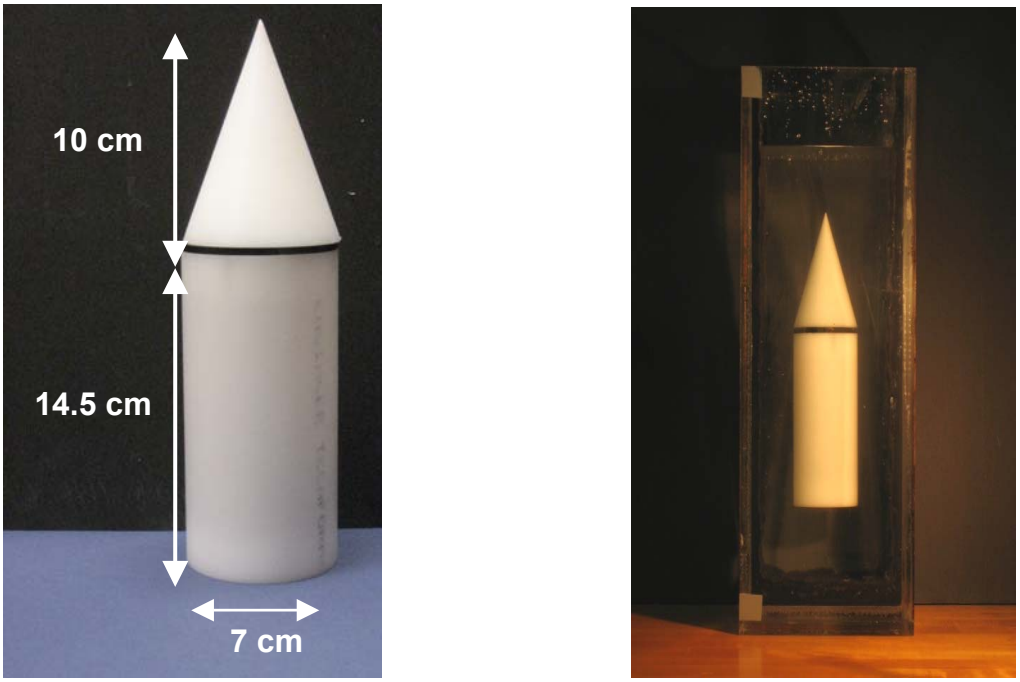

Figure 2.4: a) Cup and cone dimensions b) Swimmer in its tank

2.1.2. Motor Selection

The swimmer's motor was chosen to overcome the cone's inertia and the fluids viscosity to rotate through an order of magnitude of frequency and impart enough inertia to quickly accelerate the cone to a constant angular velocity. By assuming constant acceleration, the inertia and corresponding mechanical torque of the cone with a radius of .035 m and a mass of 183.3 g cone is determined as follows:

$$I = \frac{3}{10} m_{cone} r_{cone}^{2} = 0.067 \text{ gm}^2 \tag{22}$$

$$T_{mech} = I \alpha = \frac{I \omega}{t_{desired}} \tag{23}$$

A motor was selected with a large enough torque to make $t_{desired}$ below 1 second to reduce experimentation time.

At constant angular frequencies, the motor must only overcome viscous forces

on the axisymmetric cone. The viscous torque, that the motor must overcome to rotate the cone through a Newtonian fluid is then defined as:

$$T_{viscous} = \sqrt{2} v \omega r^3 \quad (24)$$

The motor's output torque, which is also a function of angular frequency, can be calculated by again assuming constant acceleration, using motor's speed/torque gradient of 310 RPM/mNM in Table 2.1. The appropriate motor was selected by determining the maximum theoretical frequency attainable under the given torque condition, which is represented in Figure 2.5 by the intersection of the two torque-speed curves.

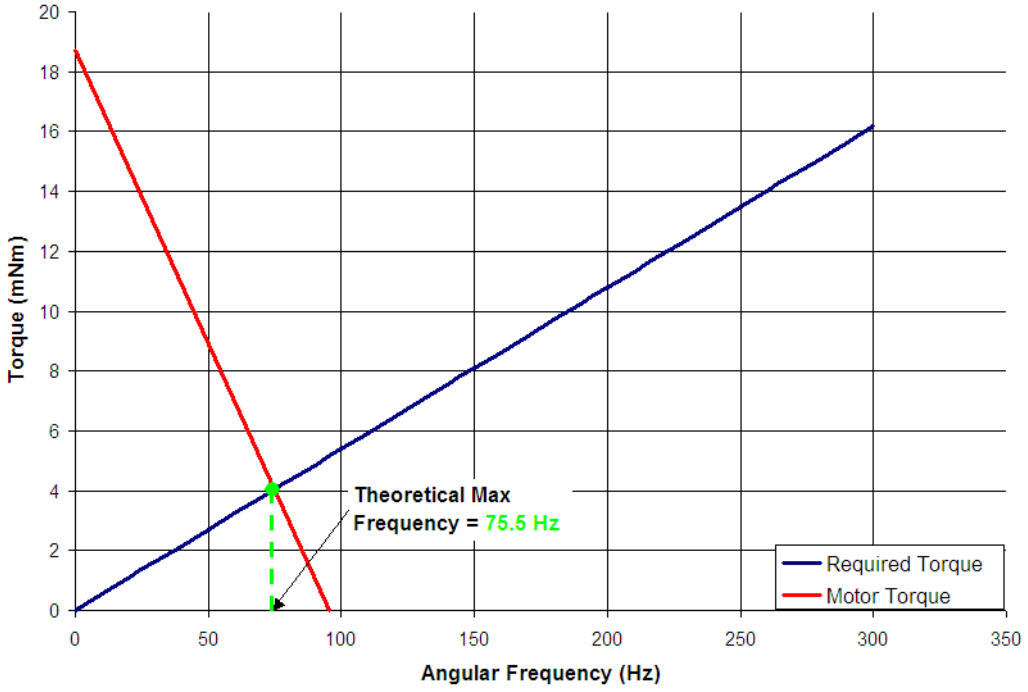

Figure 2.5: Required and availabe torque speed curves

The available and required torque-speed curves in Figure 2.5, as well as the actual motors power curve in Figure 2.6, which does not assume constant acceleration, were considered to select a motor that could rotate the cone at a two orders of magnitude range

of angular frequencies.

The motor, most suited for our available power and torque requirements was the Maxon Motor 221024 with specifications listed in Table 2.1.

Table 2.1: Maxon motor data

Motor Data	Value	Unit
No load speed	5740	Rpm
No load current	21.1	mA
Nominal speed	3810	Ron
Nominal torque	6.25	mNm
Nominal current	0.777	A
Stall torque	18.7	mNm
Starting current	2.27	A
Max. efficiency	82	%
Terminal resistance	2.20	Ω
Torque constant	8.23	mNm/A
Speed constant	1160	rpm/V
Speed/torque gradient	310	Rpm/mNm
Rotor inertia	2.18	gcm^2

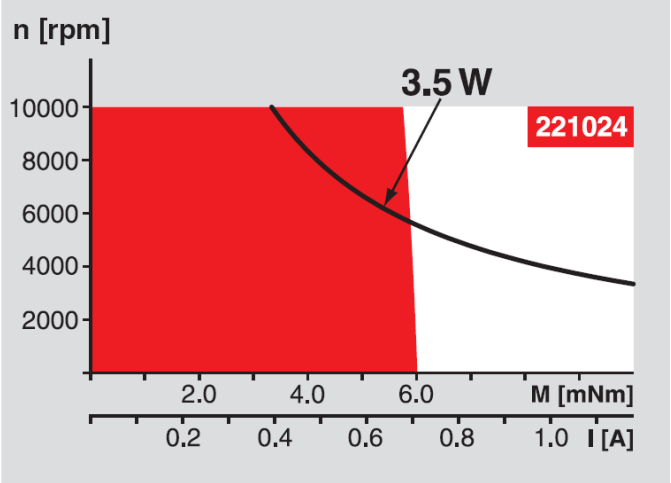

Figure 2.6: Maxon motor power curve

2.1.3. Autonomous Swimmer Control

To properly measure and control the swimmer's rotational speed, a motor controller was employed to communicate between the encoder, bluetooth chip, transmitter, and the basic stamp breadboard attached to a computer.

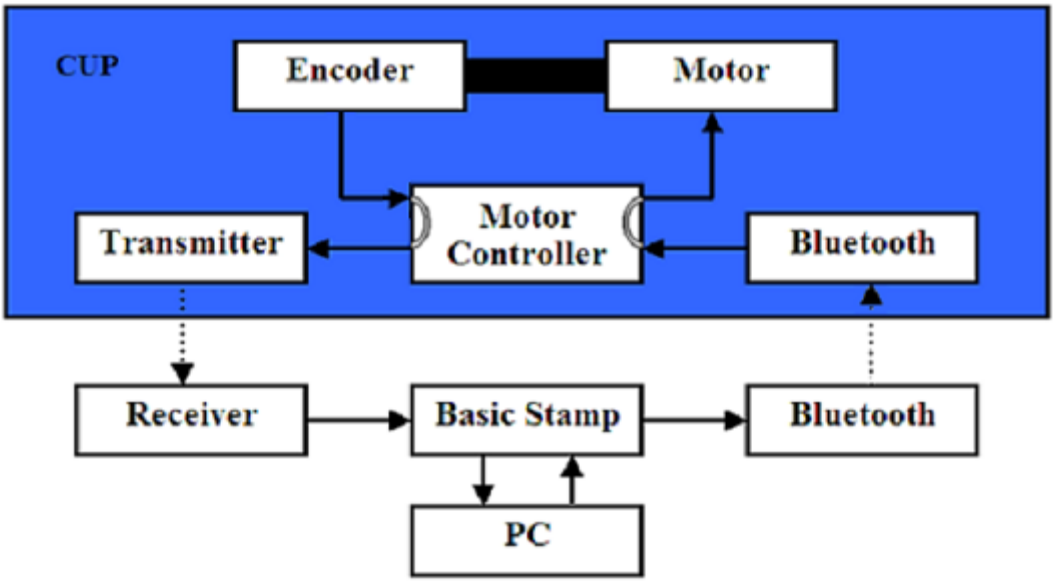

Figure 2.7: Autonomous swimmer information flow

The E4P Miniature encoder, made by US Digital, consists of a wheel, which is painted with 100 black tick marks and mounted to the back shaft of the motor, and a small photodiode, which outputs a 5V DC signal when a tick mark passes to measure the swimmer's angular frequency. The motor controller uses a PWM (Pulse Width Modulation) to output a range of voltages and allow the motor to achieve intermediate angular velocities. In addition, the input from the encoder gives the motor controller the capacity for a feedback control loop to quickly maintain a stable frequency.

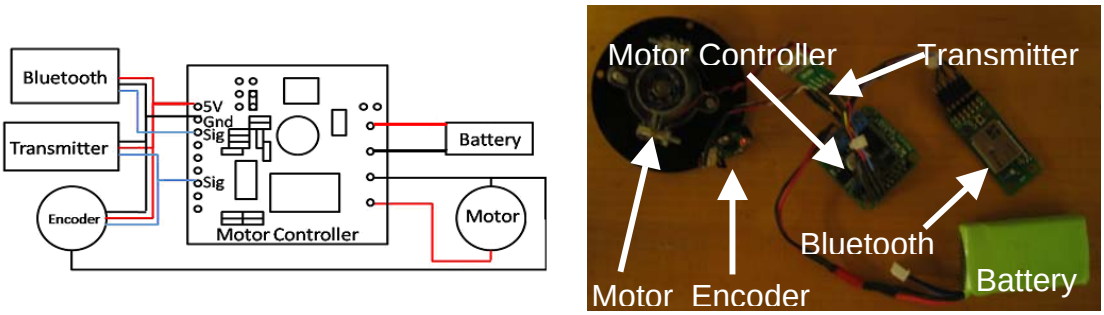

Figure 2.8: Internal swimmer wiring

The eb101 bluetooth, from the company A7, communicates wirelessly at 2.4GHz to control the motors speed, via the motor controller. The transmitter wirelessly broadcasts the encoder's output signal, at 315MHz, to the basic stamp. A small, durable and lightweight, 7.2V Lithium battery powers the motor and the swimmer's internal components.

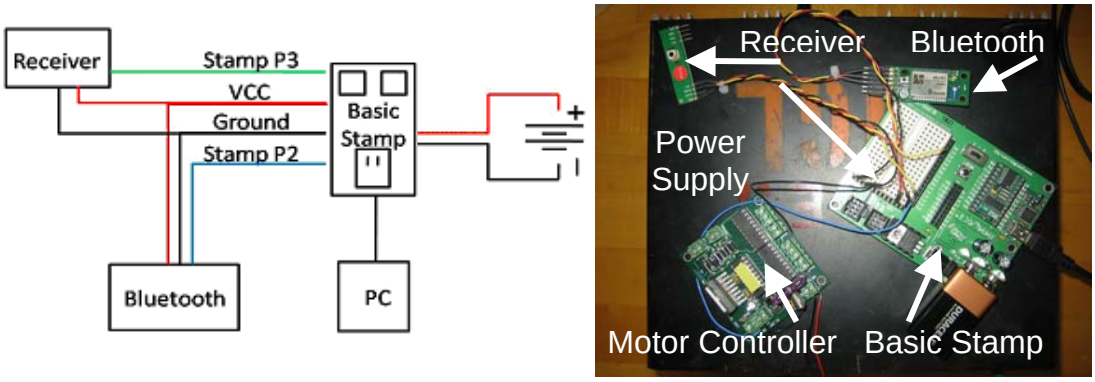

Figure 2.9: External swimmer wiring

As shown in Figure 2.9, the Parallax Basic Stamp code, found in Appendix A, reads and controls the motor's speed via the RF Link receiver and external bluetooth transmitter, respectively, and displays the encoder's output on the computer's screen.

2.1.4. Experimental Fluids

The non-Newtonian fluids used in this experiment were adopted from Bhardwaj's paper, "Filament stretching and capillary breakup extensional rheometry measurements of viscoelastic worm-like micellar solutions," which describes the rheological properties of CTAB/CpyCl and CpyCl/NaSal solutions with varying surfactant concentrations. As the required volume of CTAB would be prohibitively expensive, three solutions, with CpyCl/NaSal molar ratios of 2/1, were concocted in a 100mM brine solution, tested with a rheometer and used for the autonomous swimmer experiment.

The solute masses required for a 15L solution with CpyCl/NaSal concentrations of 50/25, were calculated using Equations (25) and (26), weighed and combined over heat with a magnetic stirring bar. After removing a small sample for rheological testing, and conducting the swimming experiments in each fluid, the necessary amounts of CpyCl and NaSal were added to form the successive, more concentrated fluid, seen in Table 2.2 until the 200/100 solution was brewed and tested. Experiments were also performed in Glycerin/Glycerol as a Newtonian control substance.

$$Molarity = \frac{N_{solute}}{V_{solution}} \tag{25}$$

$$Molar\ Mass = \frac{m}{N} \tag{26}$$

Table 2.2: Micellar Solution Concentrations

Concentration (mM)		Mass (g)			
CpyCl	**NaSal**	**CpyCl**	**NaSal**	**NaCl (Salt)**	**Water**
50	25	254.98	60.04	87.66	15,000
100	50	509.96	120.08	87.66	15,000
200	100	1019.91	240.17	87.66	15,000

The results of the steady and dynamic shear rheology tests, which were performed at room temperature (T=25°C) with the AR2000 rotational rheometer shown in Figure 2.10, are in Section 4.1. The rheometer functions by rotating a 20 mm diameter, 2° obtuse cone through the fluid, which rests on stationary lower plate, to create a constant homogenous shear rate throughout the material, as described in Section 1.2.2.

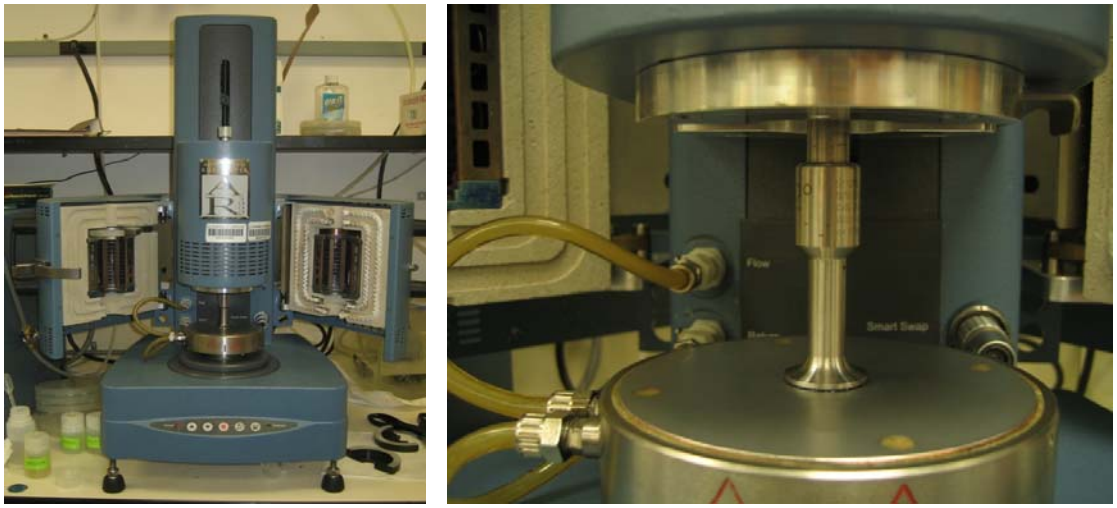

Figure 2.10: a) AR 2000 Rheometer, b) Cone-Plate Configuration

2.1.5. Autonomous Swimmer Test Procedure

This and all further experimental procedures for measuring the fluid properties, locomotion parameters and forces were designed to ensure precision, accuracy, repeatability, and expedient testing and data analysis.

Though the micellar solutions described in Section 2.1.4 were prepared for this experiment, lack of correlation between angular and translational velocity in Glycerin, shown in Figure 4.3, proved further testing to be inconsequential.

The tank was first filled with Glycerin, and was lightly shaken and rested to allow the air bubbles to settle, before the swimmer was placed, cone up for stability, towards the top of the tank. The swimmer's angular velocity was then set, via the Basic

Stamp program, and a Canon PowerShot SD630, mounted on a tripod for image stability, recorded the swimmer's motion as it traversed the tank. Finally, the video's filename and encoder's frequency output were manually recorded.

The Matlab image processing code in Appendix B, automatically tracked the swimmer's centroid, and averaged its velocity through the video's length.

2.2. Tethered Swimmer Experiment

A tethered robot was designed and built to reduce the cost and complexity of testing, by shrinking the submerged control surfaces, to reduce the required volume of test fluid. The swimmer, with an exposed motor to prevent overheating and simplify construction and control, mimics the incipient swimmer model, not discussed in this thesis, which was built of Legos for MAE 171a at UCSD, that utilized a four-wheeled cart, which rides along the edges of a small tank, to support the motor, transmission and submerged rotating cone. Although this assembly eliminates the need for neutral buoyancy, and greatly increases the stability of the robot, the wheels and wires induce external drag, which influences the swimmer's velocity. The robot was designed in Autodesk Inventor to reduce the size of the swimmer and thus the required volume of fluid, and was built of Acrylic, Delrin and Steel, at the engineering design studio and the machine shops in the department of Mechanical and Aerospace Engineering at UCSD, and at Scripps Institute of Oceanography.

2.2.1. Tethered Swimmer Design

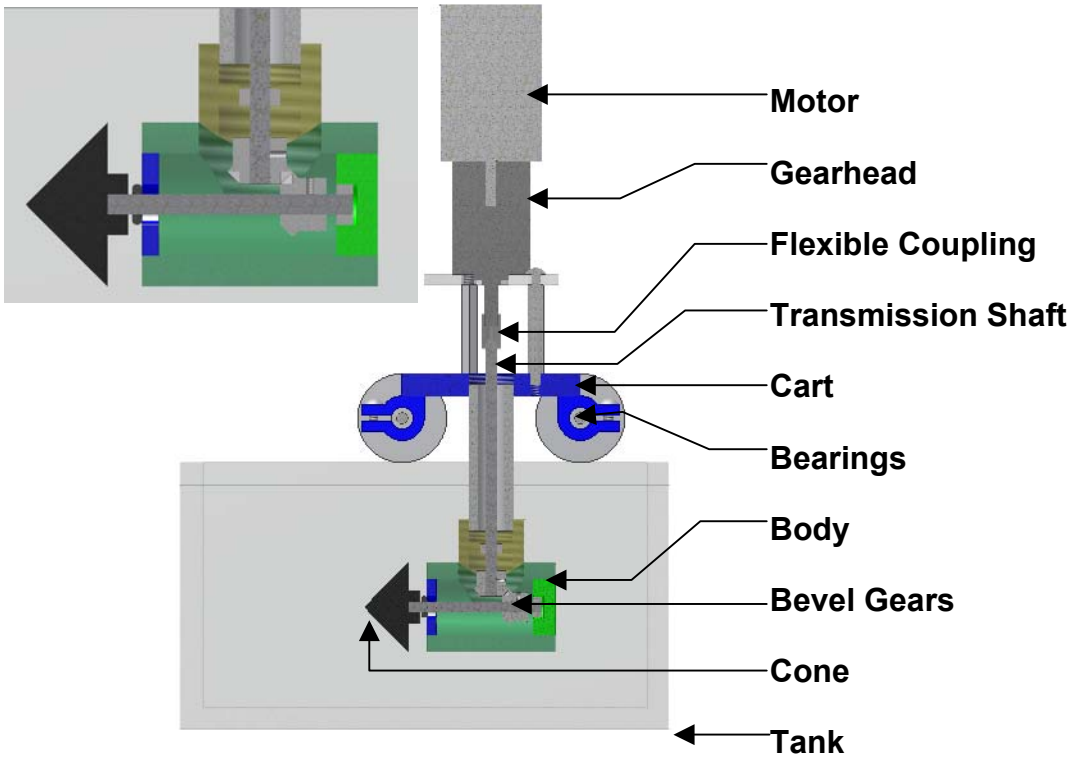

Figure 2.11: CAD drawing of swimmer transmission

Power is transmitted from the Maxon 29mm 15 W brushless motor, directly to the flexible coupling and transmission shaft, or through a 14:1 planetary gearhead for reduced frequencies. The flexible coupling, constructed of supple polypropylene tubing, easily connected to the motor's gearhead, and provided forgiveness for the misalignment between the motor or gearbox shaft and the transmission shaft. The transmission shaft, which is supported and constrained by a bearing within the body, is connected to and passes through the encoder, and terminates at a 48 pitch, molded nylon 45-degree beveled gear. A matching bevel gear transmits the torque to the cone's rotation shaft, which is supported by bearings at both ends. The swimmer's cone was machined of Delrin for its resilience and machinability, and was painted with two stripes of sliver reflective paint to allow for frequency detection by a digital non-contact tachometer. The body was

machined of acrylic, for its opacity and inertness, in two separate pieces, to allow for proper assembly and cleaning.

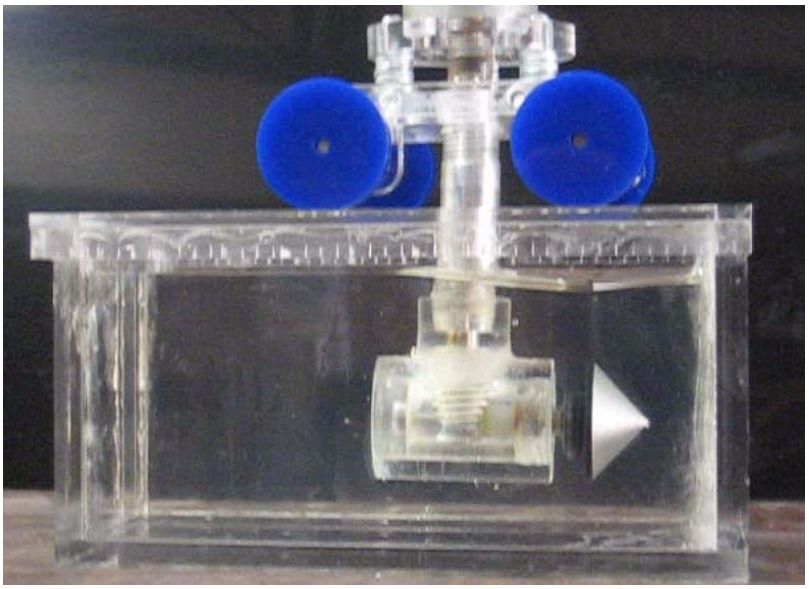

Figure 2.12: Tethered swimmer in silicon oil

As the swimmer's motor is not submerged, a cable umbilical replaced the autonomous swimmer's wireless bluetooth and transmitter connections, as seen in Figure 2.13, and a motor controller is used instead of a bluetooth chip to modify the speed of the motor. The Basic Stamp program in Appendix A sets the motor's speed and counts the encoder's output signals to determine the swimmer's the angular frequency.

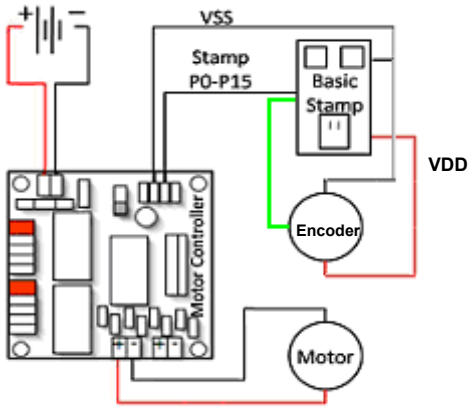

Figure 2.13: Tethered Swimmer Wiring

A second model of the tethered swimmer, replaced the previously separable and crumbling acrylic body and attachment shaft, with a solid, press-fit aluminum assembly, to eliminate the gear misalignment during post-cleaning reassembly, and substituted copper gears for the worn plastic miter gears as seen in Figure 2.15. As the tube coupling severed or slipped at high speeds and torques, a Ruland Flexbeam clamp-style flexible coupling, was purchased and re-machined to accept the motor's gearhead. The gearhead was not used for this experiment as it's additional weight impeded the swimmer's velocity, as seen in Section 4.4. In addition, a new encoder, with a thick, sheathed cable for protection from the motor's EMF, replaced the tachometer for more accurate frequency measurement.

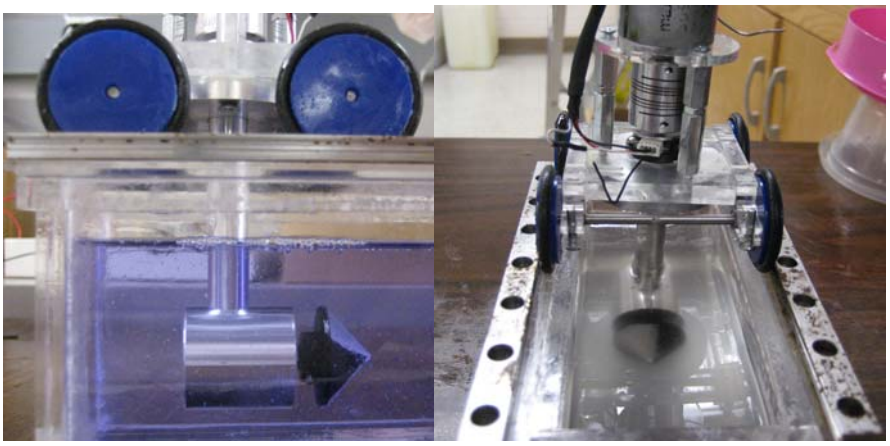

Figure 2.14: Final elastic swimmer robot

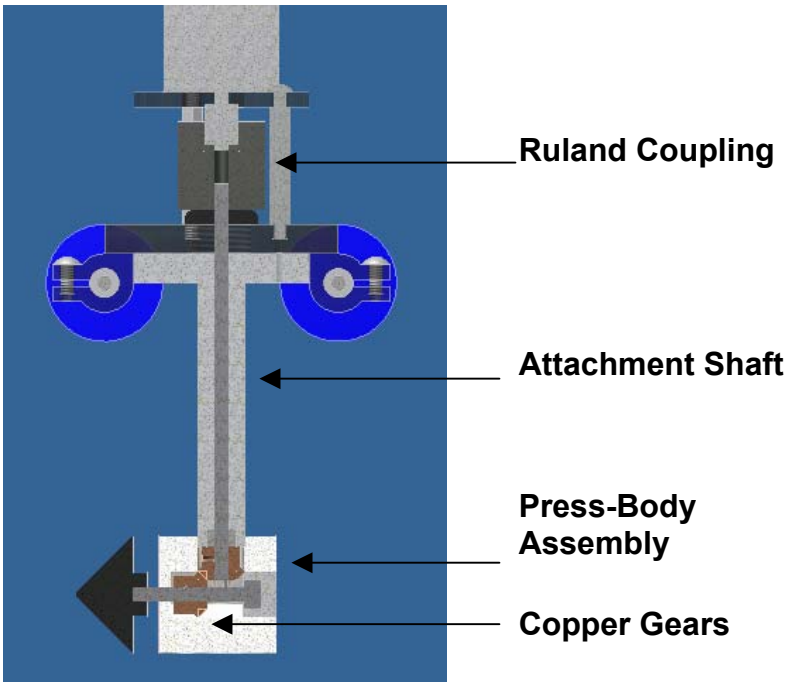

Figure 2.15: CAD model of final elastic swimmer robot

As the cart, whose mass was increased to 788 grams began to roll at a $.26^{O}$ incline, the cart's wheels' coefficient of friction can be calculated as:

$$\mu_s = \frac{F_f}{F_n} = \tan(\theta) = .266 \qquad (27)$$

2.2.2. Experimental Fluids

The choice of Non-Newtonian fluids for this test were inspired by Matthiew Vargagnat's paper, "Instabilities of Jets of Non-Newtonian Fluids Impacting a Plate," which describes the rheological properties of two commercial shampoos: Pantene Pro-V "Ice Shine" and Herbal Essence "Totally Twisted", both of which are transparent, inexpensive, stable and have a appropriate relaxation times. The results of the rheometry testing of the shampoos with the cone-plate rheometer described in Section 2.1.4, are in

Section 4.2. Polydimethylsiloxanes, Trimethylsiloxy Terminated, or Silicon Oil, with a kinematic viscosity of 10,000cSt was used as a control.

2.2.3. Tethered Swimmer Test Procedure

After filling and agitating the tank, to remove air bubbles that might disrupt the flow, the swimmer was slowly inserted into the tank, with the four wheels riding on the tank's edge, as seen in Figure 2.14. The swimmer was tested over 10 sets of 10 increasing frequencies. The counterclockwise angular frequencies were measured either by the encoder for 30 seconds, for the low frequency tests, or by the digital non-contact tachometer for 6 seconds, for the high frequency tests.

The final tethered swimmer ran for three sets of 10 frequencies at both 10V and 20V for 6 seconds and 30 seconds per test, respectively.

Figure 2.16: High speed camera on positioning boom arm

The experimenter tracks the swimmer's position, using the MATLAB code in Appendix B, by clicking on a prominent feature of the swimmer in 3 successive displayed pictures, which were captured at a time delay by a high speed Pixelink PL-B741F Monochrome Machine Vision Camera. The three velocity measurements,

calculated from the three position points and the camera's time delay are averaged to determine the swimmer's speed.

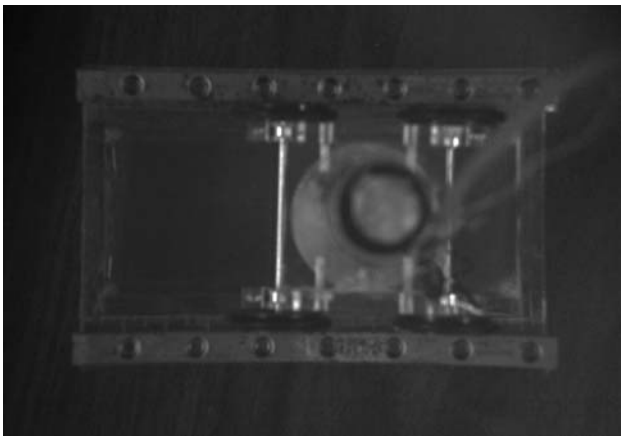

Figure 2.17: Sample of test picture used for velocity measurement

3. Shear Thinning - Snail Robot

The shear thinning effects of linear one degree-of-freedom propulsion on the surface of a non-Newtonian fluid are shown through the design of the "Snail Robot," which mimics the locomotion of snails in nature, by swimming across the surface of an elastic, non-Newtonian, fluid through asymmetrical phases of contraction and expansion.

3.1. Snail Robot Design

The snail robot, designed in Autodesk Inventor, consists of two unequally sized Styrofoam buoys, attached by a linear slider, which separate and contract based off the position of a cam.

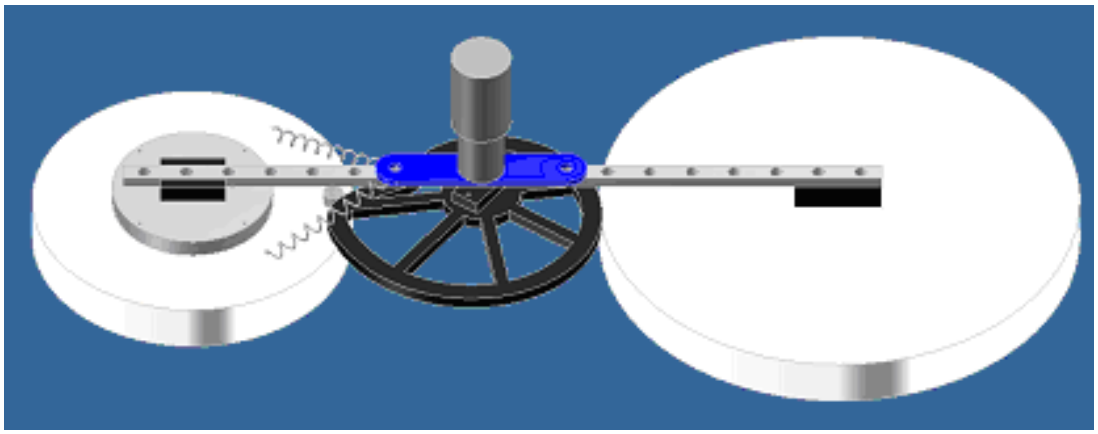

Figure 3.1: CAD model of snail robot

The, 15.24cm and 25.4cm diameter, buoys are wrapped in cellophane to prevent fluid absorption and reduce deterioration, and are capped with an acrylic plate to firmly mount to the linear slide. The motor and, 14:1 or 53:1, gear head, which are mounted to the linear slide rail via a mounting plate, turn a cam that remains in contact with a bearing on the small buoy through springs. To shift the center of mass towards the midpoint between the buoys centers, 4oz weights were added to the large buoy as seen in Figure 3.2

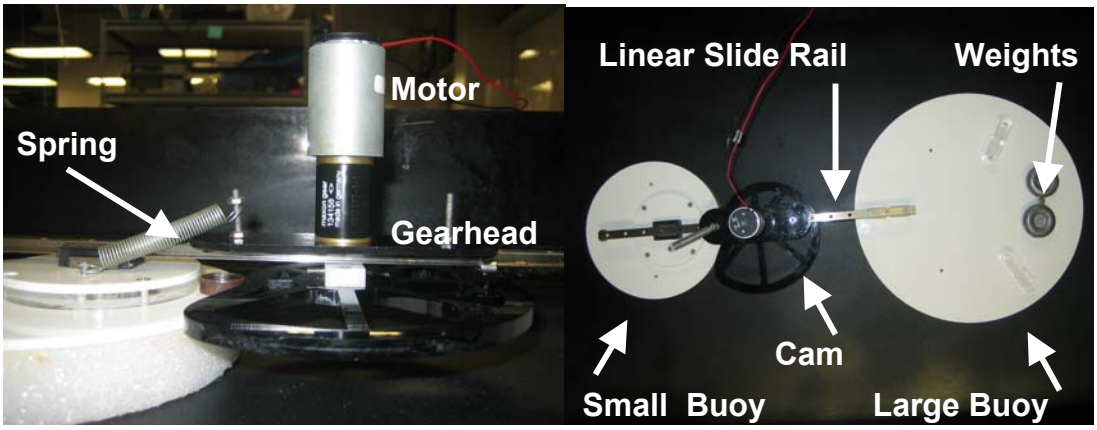

Figure 3.2: Schematic snail setup

A 7th degree polynomial cam design was performed by Autodesk Inventor to allow for a separation amplitude of and minimum radius of 1.5in, and a ratio of expansion rate to contraction rate of 5:1. The cam's pressure angle was set at 52.46 degrees to minimize the contraction phase while preventing stall and loss of contact with the bearing. As the initial cam, with a glued acrylic shaft insert, could not withstand the high torques imposed by the fluid's viscosity at high speeds, a new cam, in Figure 3.3b, was manufactured with a mounted Delrin shaft housing with a flat to reduce backlash.

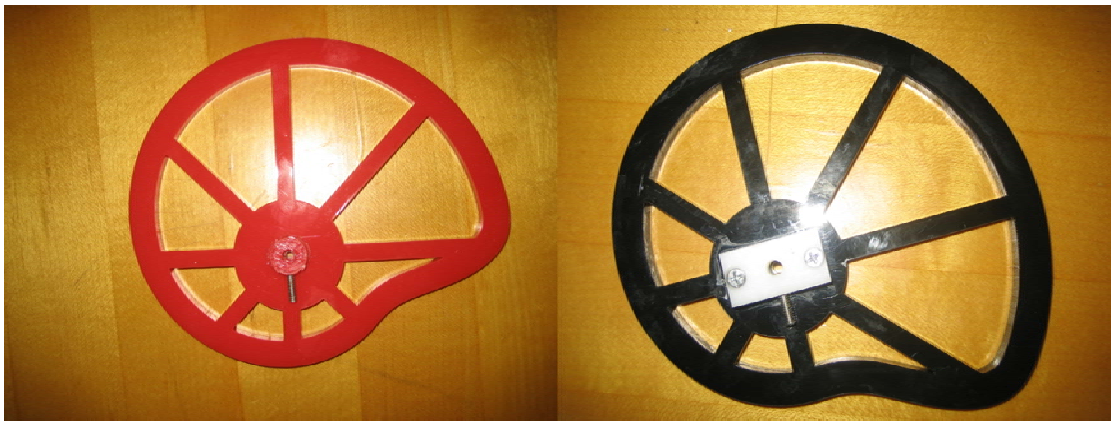

Figure 3.3: a) Initial low frequency cam, b) High frequency cam

The 40" x 22" x 5" acrylic tank, painted black for improved image processing contrast, allows the robot to swim its body length while maintaining clearance on each

side, holds enough liquid to allow the snail to float, and has splash guards to prevent fluid loss. The tank was cut using a LaserCAM and sealed with black RTV Silicone.

3.2. Snail Experiment Procedure

The snail was placed towards one side of the tank, after the poured fluid was agitated and allowed to settle. The image processing software described in Section 3.3 utilized the experiment videos, recorded at 30 frames per second by a Canon PowerShot SD630 digital camera, to calculate the snail's velocity. Experiments were performed three times each at frequency for 50 revolutions in the micellar solutions, described in Section 2.1.4, and in the control fluid, glycerin.

The motor speed control program found in Appendix A, similar to that of the tethered elastic swimmer, employs the PWM function in the Basic Stamp Circuit board to parcel the power supply's 10V or 20V, to, in combination with the 2 available gear heads, allow for a large, discrete frequency and velocity range.

3.3. Snail Velocity Measurement

The program seen in Appendix B was developed to automatically measure the velocity of the irregular and periodically spasmodic swimmer. The code was written to extract the velocity of the snail's centroid, while ignoring the snail's undulations, and satisfying the Nyquist Criterion to prevent aliasing.

The code first determines the scaling factor between image pixels and physical distance by recording, the locations of two manually clicked points of known distance, placed far enough apart to reduce scaling error. The images are then cropped, converted to grayscale and then to black and white, and are finally filtered, using a threshold pixel value, to further distinguish between black and white features. The clumps of black and

white pixels are then dilated, labeled and sorted by size. The centroids of the two largest clusters of white dots, which represent the buoys, are then determined for each frame and thus the instantaneous velocity of each buoy, in the X and Y directions, is measured. Although the snail's X velocity is cardinal, the X component of a single buoy's velocity, alone, is misrepresentative as rotation about the Z-axis occurred and would be indistinguishable from translation. Thus the snail's motion was characterized as the velocity of it's centroid in the Q axis, which is defined by the linear slide rail or by the vector between the buoys' centroids, seen in Figure 3.4.

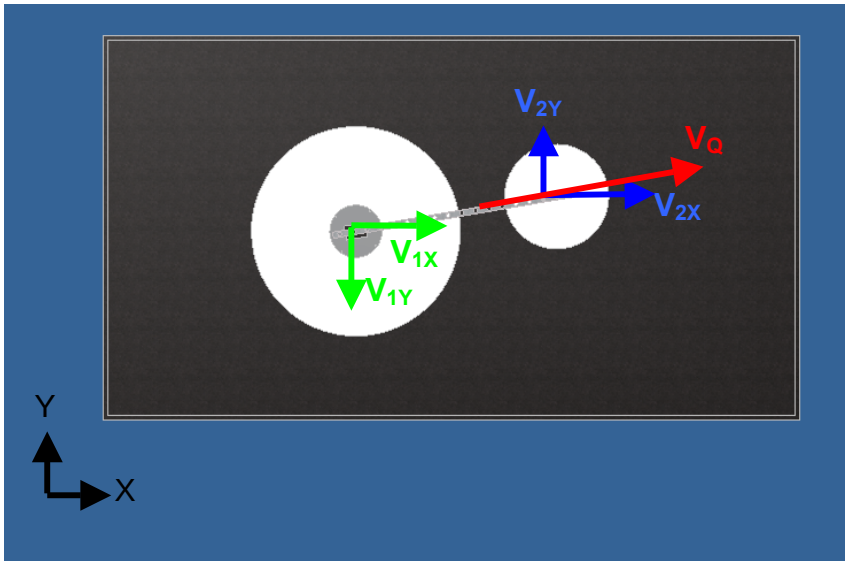

Figure 3.4: Velocity components of the snail robot

By defining the instantaneous centers of the buoys as (X_1 , Y_1) and (X_2 , Y_2), the instantaneous vector Q can be calculated as:

$$Q = [(X_2 - X_1) , (Y_2 - Y_1)] \quad (28)$$

As the snail is rigid, when time averaged, the buoys' mean velocities in the Q direction are thus equal and equivalent to that of the snail's centroid. Thus the instantaneous velocity of the snail in the Q direction, V_Q, is calculated by:

$$V_Q = [V_{1X} , V_{1Y}] \bullet Q' \quad (29)$$

The snail's frequency and speed can be determined by counting the velocity's sign switches over a set time, and by calculating the slope of a linear fit of the data, respectively, for the resulting sinusoidal curve in Figure 3.5.

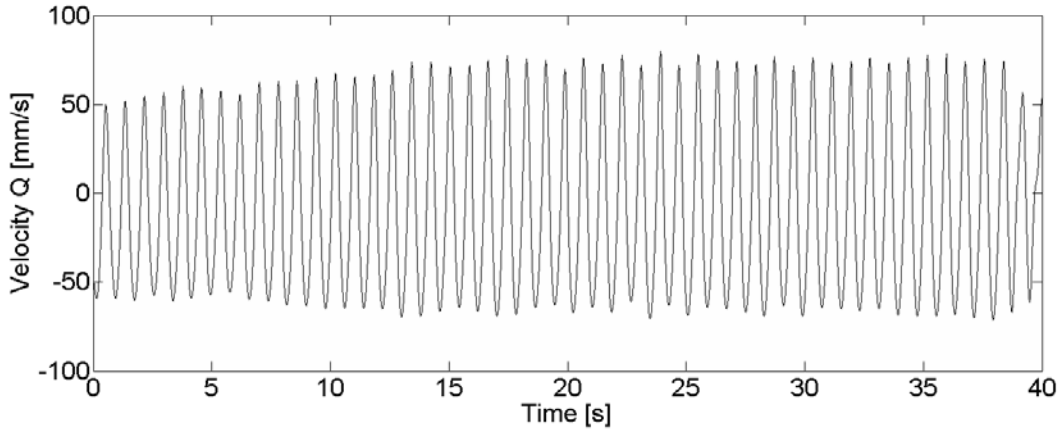

Figure 3.5: Velocity-time graph of snail

4. Experimental Results

The following chapter discusses the measured rheological properties of the micellar and shampoo fluids, as well as the test results of the autonomous and tethered normal stress experiments and the shear thinning experiment.

4.1. Micellar Rheology Results

The storage and bulk modulus of the micellar solutions, for the shear stress and autonomous elastic swimmer experiments, and the shampoos, for the tethered swimmer experiment, were measured over a range of frequencies using a rotational rheometer described in Section 2.1.4. The theoretical modulus, predicted by the Maxwell Model in Section 1.2.2, displays a decent fit to the measured laboratory data as seen in Figure 4.1. The rheometry results confirm that the micellar fluids are shear thinning, that the storage and bulk modulus increase with surfactant concentration, and that the control substance, Glycerin, is Newtonian.

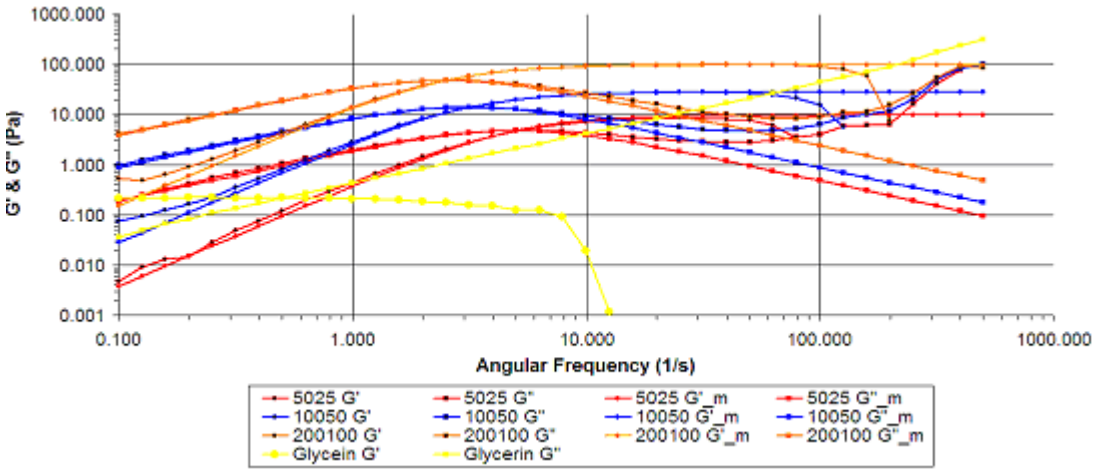

Figure 4.1: Bulk and storage modulus of micellar solutions

Figure 4.2, which displays the micellar fluid's viscosities as a function of shear

rate, shows that higher concentrated solutions are in fact denser, and further proves that glycerin is Newtonian. The Maxwell Model equations, described in Section 1.2.2 were used to accurately calculate the micellars' rheological characteristics displayed in

Table 4.1, though the zero shear viscosity of the 50/20 micellar solution disagreed with the viscosity results in Figure 4.2, which reaches 1.91 Pas.

Table 4.1: Viscoelastic properties of the micellar solutions

Micellar Solution	Zero Shear Viscosity η_0 (Pas)		Plateau Modulus G_P (Pa)		Maxwell Relaxation Time λ (s)	
	Experiment	Published	Experiment	Published	Experiment	Published
50/25	1.91	1	9.57	4.2	0.20	0.27
100/50	8.82	11	27.84	27	0.31	0.50
200/100	38.19	69	95.70	104	0.40	0.59

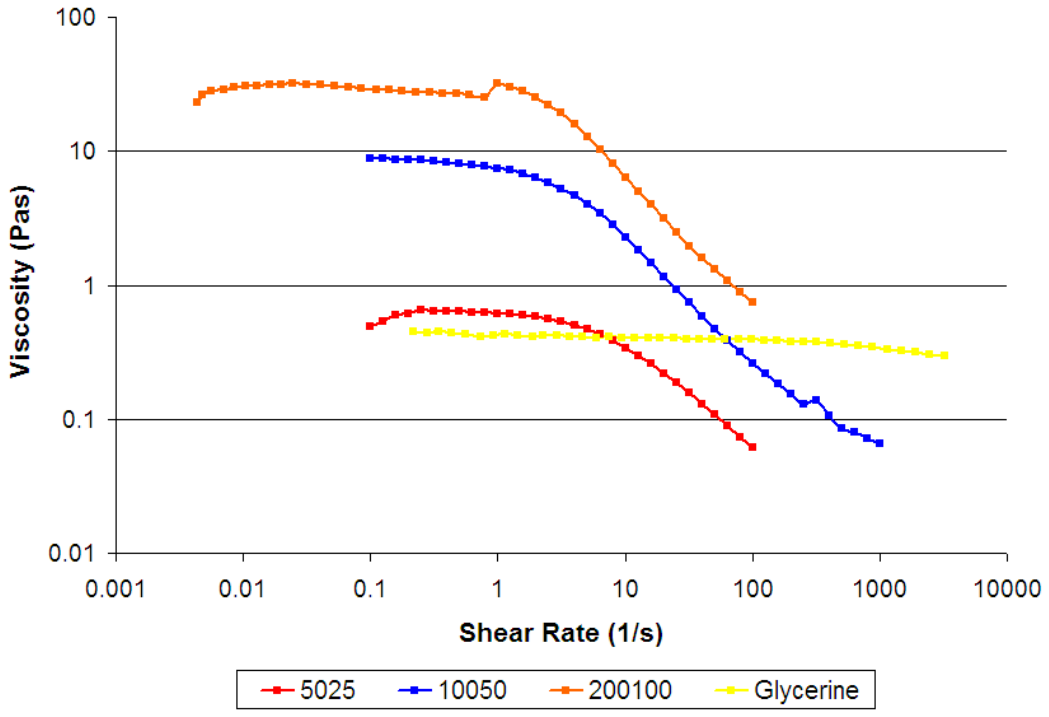

Figure 4.2: Viscosity of micellar solutions as a function of shear rate

As the high viscosity 200/100 micellar solution began to evaporate during testing, two separate rheometry tests were required, causing the kink seen in the viscosity measurement above.

4.2. Shampoo Rheology Results

The results of the rheometry testing, along with the Maxwell Model curve fits, of the two shampoos, shown in Figure 4.3 were obtained using the same equipment and procedure as for the micellar solutions.

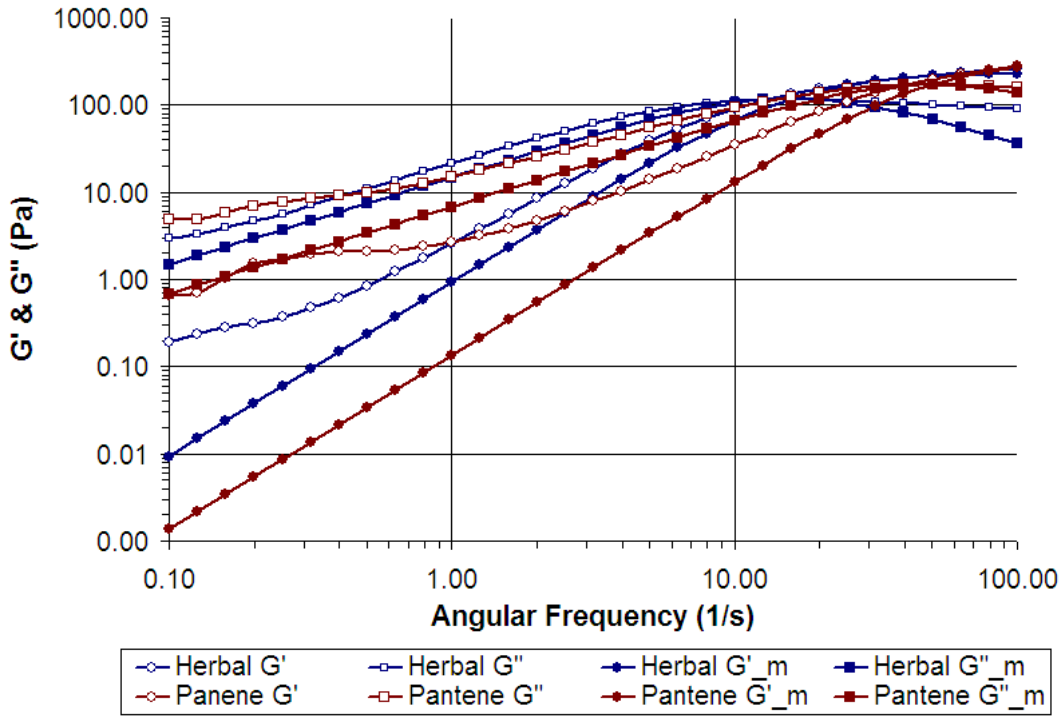

Figure 4.3: Bulk and storage modulus of shampoos

The published and experimental rheometry data, shown in Table 4.2, are much more congruent for the Herbal Essence "Totally Twisted" shampoo, than for the Pantene Pro-V "Ice Shine" shampoo, as the specific brand of Pantene shampoo Varagnat's

experiment was no longer distributed.

Table 4.2: Viscoelastic properties of shampoos

Shampoo	Zero Shear Viscosity η_0 (Pas)		Plateau Modulus G_P (Pa)		Relaxation Time λ (s)	
	Experiment	Published	Experiment	Published	Experiment	Published
Pantene	6.82	12.8	341.8	257.5	.020	0.0497
Herbal	14.85	16.7	235.4	236.2	.063	0.0707

In addition, the calculated zero shear viscosity of the Pantene Pro-V shampoo in Table 4.2 disagrees with the viscosity testing in Figure 4.4, that shows a nominal viscosity of around 25, not 6.82, Pas.

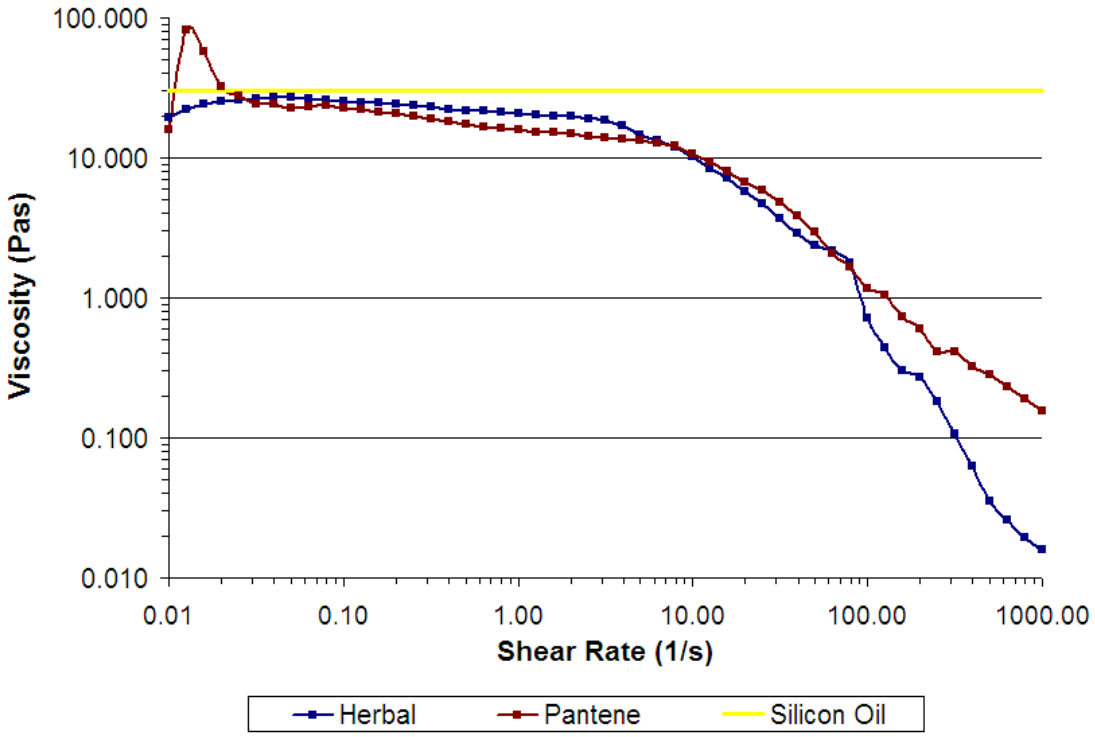

Figure 4.4: Viscosity of shampoos and silicon oil as a function of shear rate

4.3. Autonomous Swimmer Results

Testing was only performed in Newtonian Glycerin, as the confined motor overheated, and the swimmer's non-zero velocity was decoupled from its angular velocity, as seen in Figure 4.5. As the swimmer, with a radius of .035m, rotating at 100Hz though Glycerin with a kinematic viscosity of 3.17×10^{-4} m^2/s (or 3.17 cSt), displays a Reynolds number of 386.18, per Equation (19), that is much larger than unity, the swimmer motion is momentum induced and thus cannot fully display the non-Newtonian effects on Purcell's Scallop Theorem.

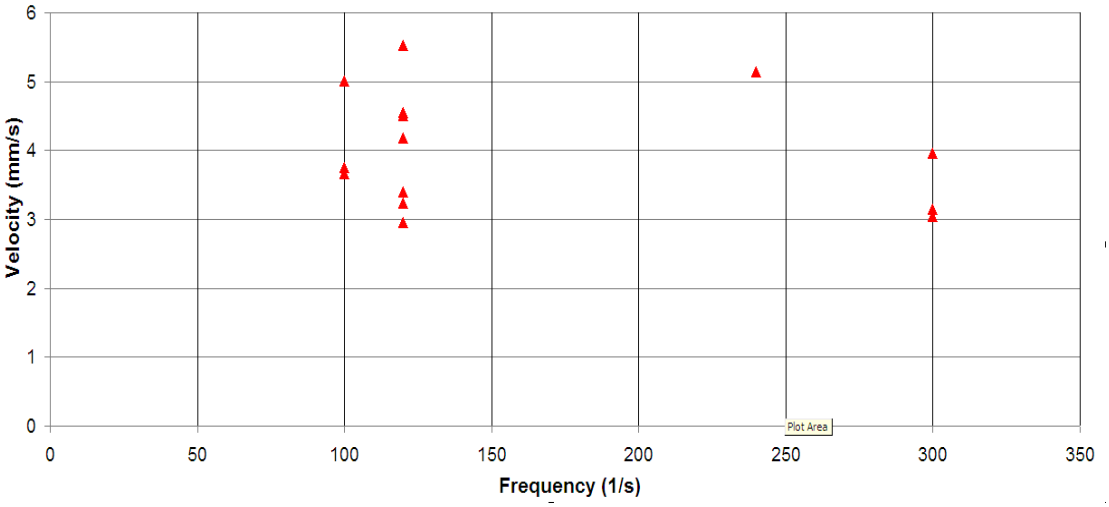

Figure 4.5: Autonomous swimmer velocity in glycerin

4.4. Initial Tethered Swimmer Results

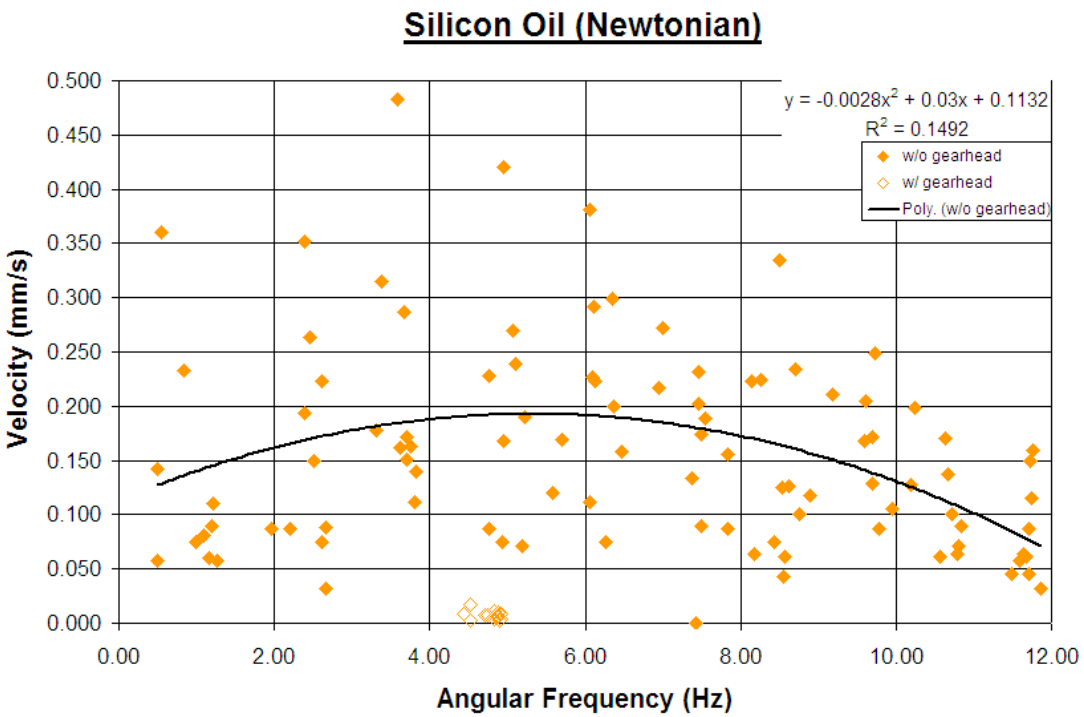

Figure 4.6: Initial tethered swimmer velocity in Silicon oil

The results for the initial tethered swimmer experiment, in Figure 4.6 showed little correlation between angular and linear velocity, with a low R^2 value of .15 to a 2nd-order polynomial fit. The geared down robot proved sluggish, from the added gear head weight, and displayed an angular frequency range too small to extend the data set, proving the experiment inconclusive and disparate with the non-geared results. As the 0.0254m swimmer's minimum and maximum frequencies through 10,000cSt Silicon oil were .49 and 12 Hz, respectively, the minimum and maximum Reynolds numbers, calculated using Equation (19), are.032 and 0.774, and are less than unity, but far from the "Low Reynolds" regime.

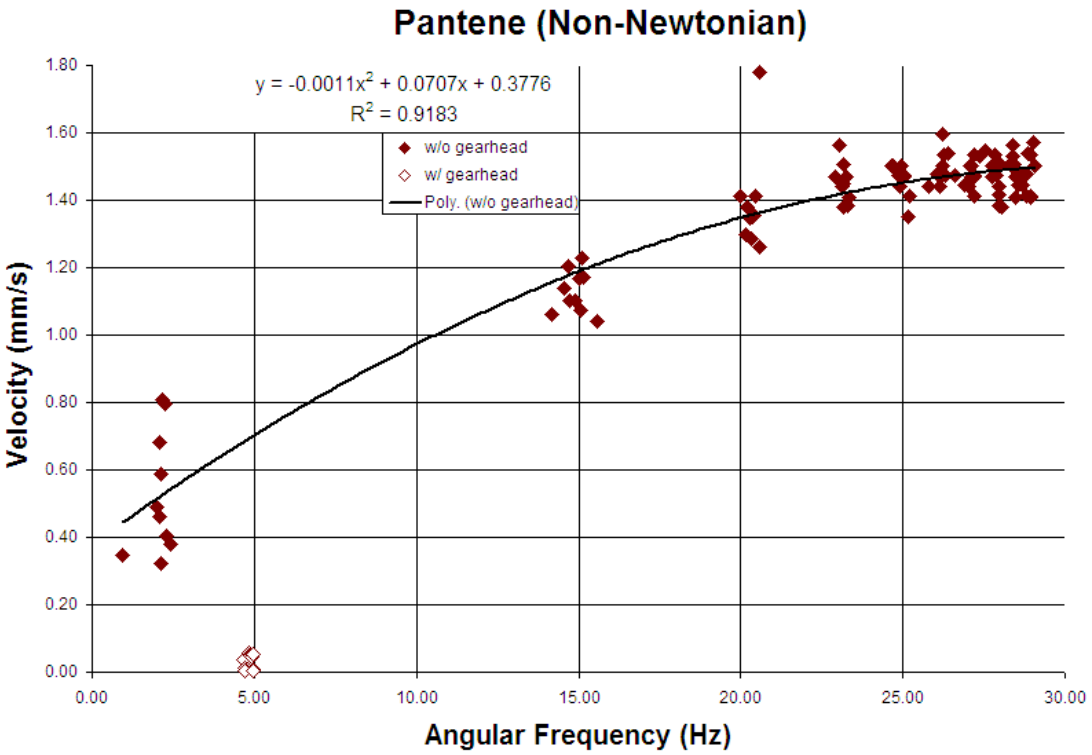

Figure 4.7: Initial tethered swimmer velocity in Pantene Pro-V shampoo

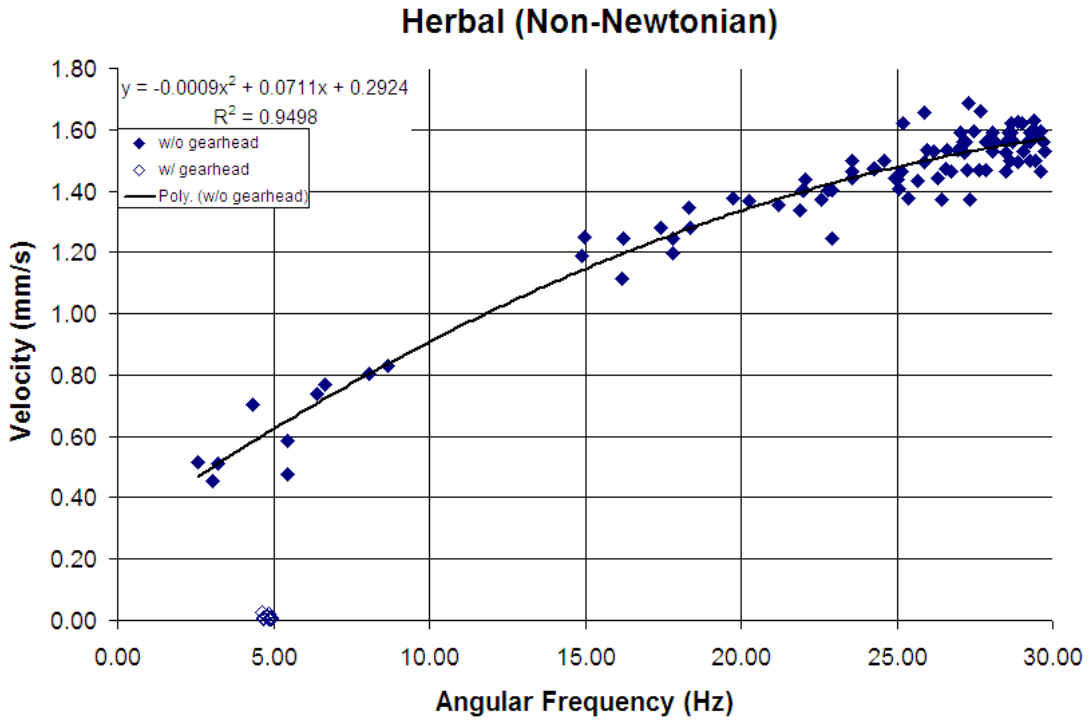

Figure 4.8: Initial tethered swimmer velocity in Herbal Essence shampoo

The experiment results for the Pantene and Herbal Essence shampoos proved much more coherent, with high R^2 values of .92 and .95 in Figure 4.7 and Figure 4.8, respectively. Both experiments, however, contain large frequency gaps and do not display the fully expected inverse parabolic relationship between angular frequency and linear velocity.

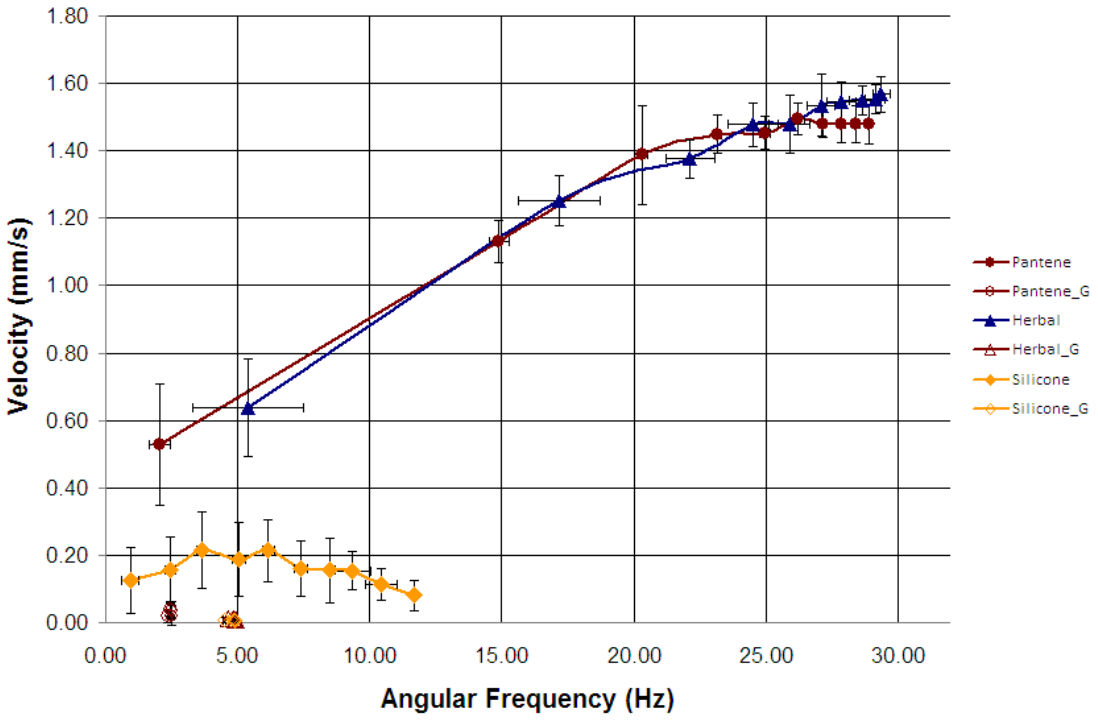

Figure 4.9: Initial tethered swimmer grouped by voltage input

The combined experimental results grouped and averaged by input voltage in Figure 4.9 above, displays larger standard deviations at low frequencies and velocities, though changes in the motor's temperature, and in the fluid may have altered the available and required power, respectively, at a given voltage. The error in the velocity measurement was calculated as ± .085 mm/s, by performing the MATLAB procedure described in Section 2.2.3, with several copies of the same image, which is larger than the ± .077 mm/s average standard deviation of the grouped data.

Although the robot translated congruently and faster though the two shampoos than through the silicon oil control, as seen in Figure 4.10, the missing range of control frequencies in silicon oil prevents the experiment from fully disproving Purcell's Scallop Theorem.

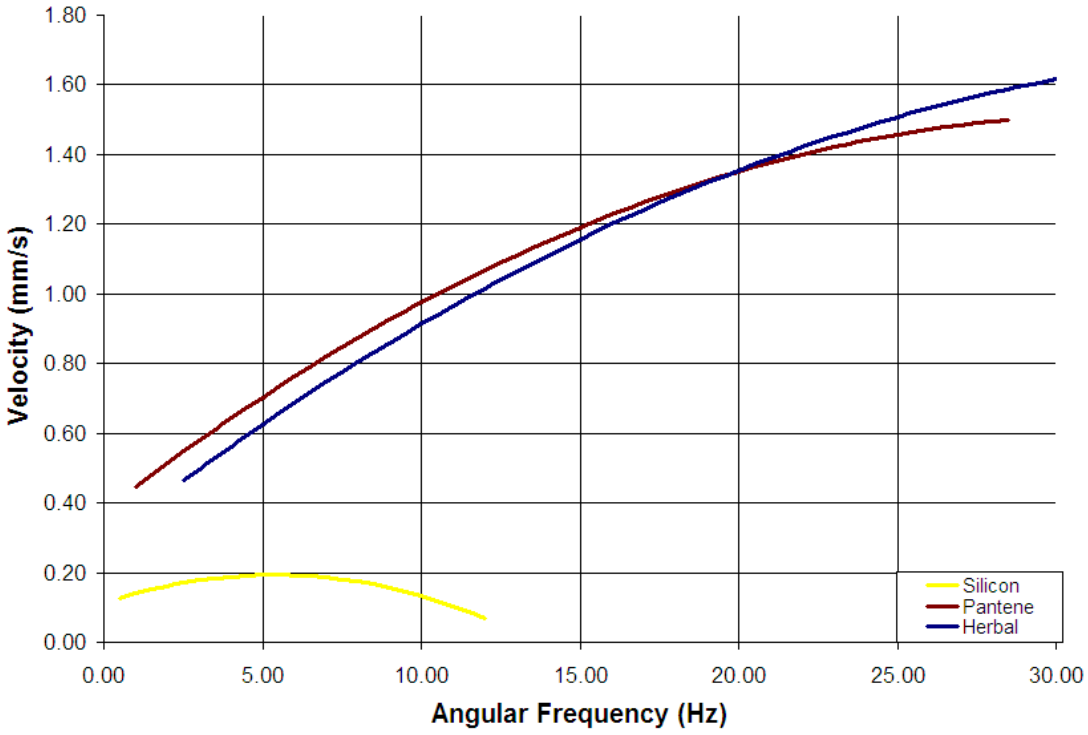

Figure 4.10: Combined Initial tethered swimmer velocity trend lines

4.5. Final Tethered Swimmer Results

The relationship between angular frequency and velocity for the new tethered swimmer were much less accurate, with R^2 values of .6369, .3037 and .0376 in the Pantene and Herbal shampoos and in the Silicon oil respectively.

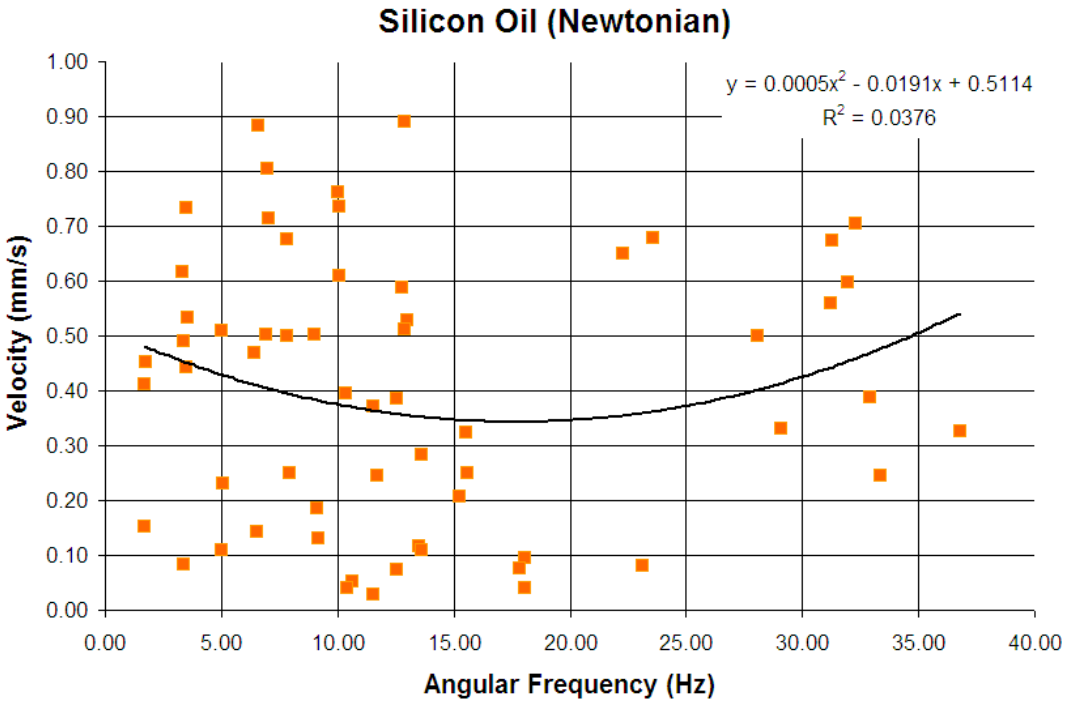

Figure 4.11: Final tethered swimmer velocity in Silicon oil

Although the swimmer's motion through Silicon oil, in Figure 4.11 is inconclusive, the frequency range is more continuous than the easily distinguishable 10V and 20V test regions in the Pantene and Herbal Essence results in Figure 4.12 and Figure 4.13. The final swimmer, with minimum and maximum angular frequencies of 1.69 and 37 Hz, respectively, has a Reynolds number range from .109 to 2.38, which again is far from the "Low Reynolds" regime.

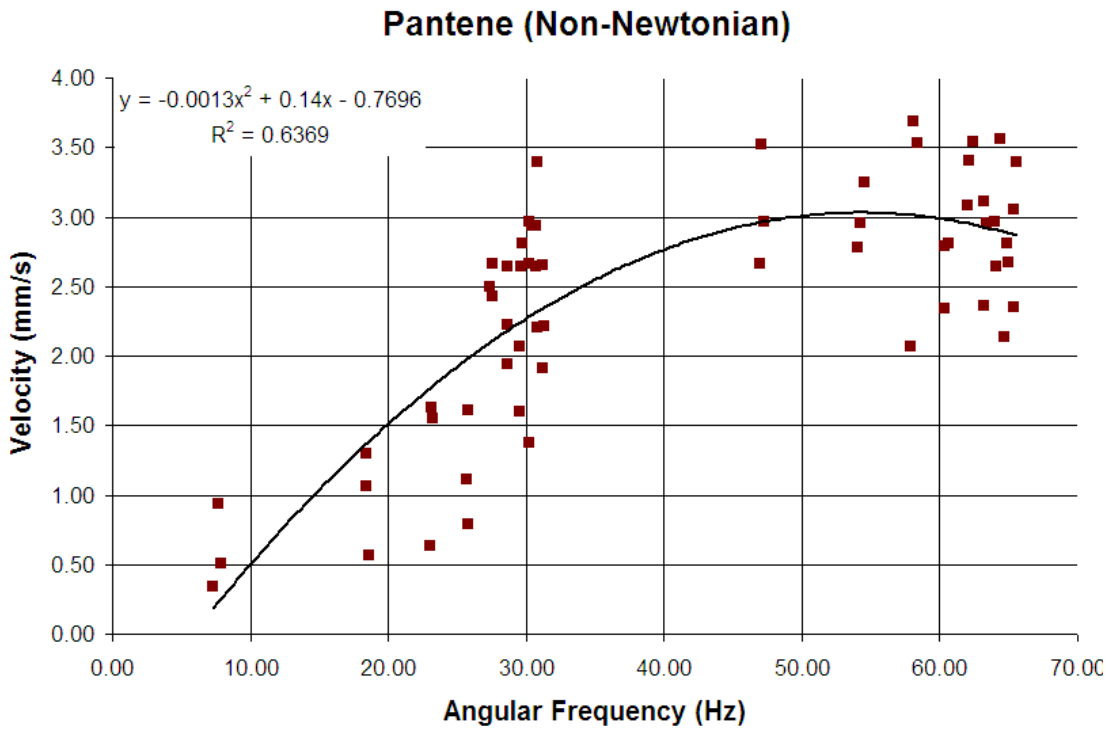

Figure 4.12: Final tethered swimmer velocity in Pantene Pro-V shampoo

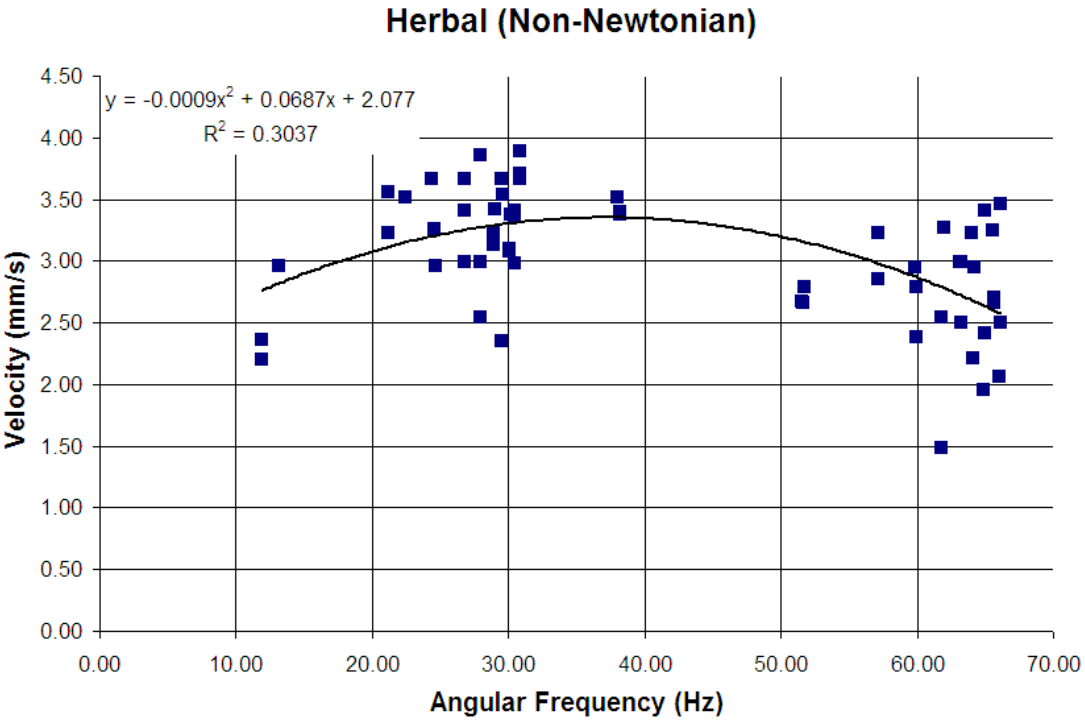

Figure 4.13: Final tethered swimmer velocity in Herbal Essence Shampoo

Grouping and averaging the experimental results by input voltage in Figure 4.14, again displays large standard deviations at low frequencies and velocities, though grouping the results by voltage can be misleading as described in the previous experiment's results. As opposed to the initial tethered swimmer experiment, the ± .320 mm/s average standard deviation of the grouped data is much larger than the MATLAB image processing induced error of ± .085 mm/s. Although the thickened encoder cable was strung vertically away from the swimmer to reduce tugging, the cable's rigidity was most likely the main source of error in this experiment.

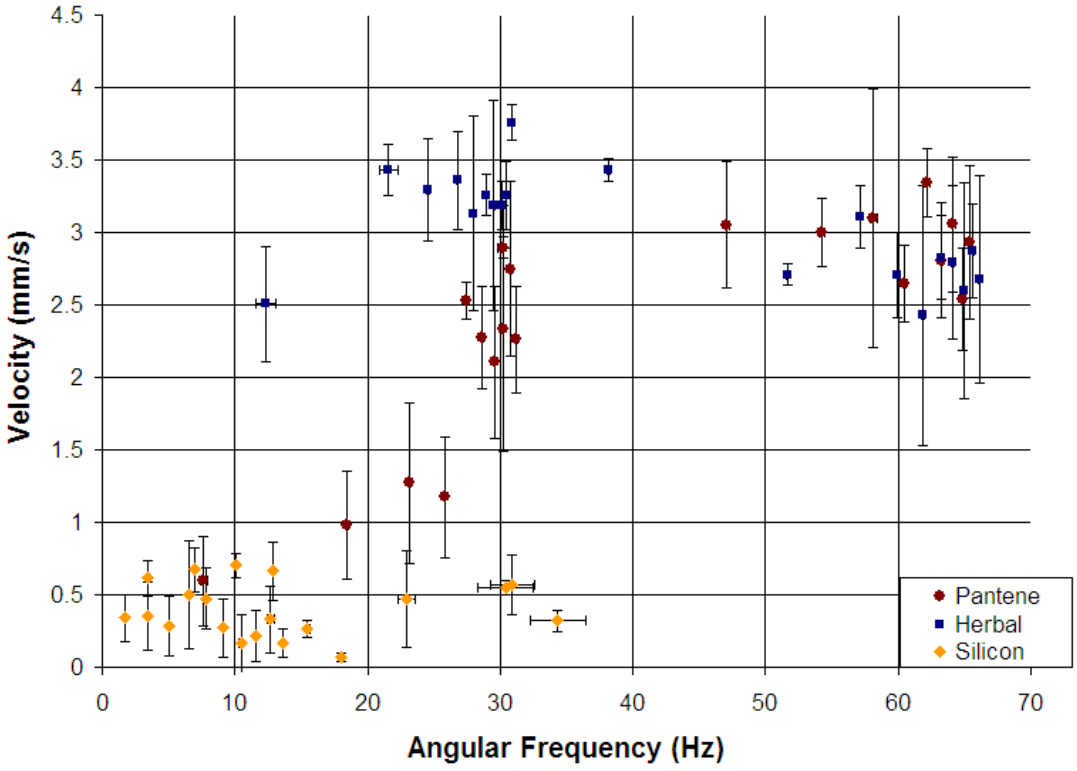

Figure 4.14: Final tethered swimmer grouped by voltage input

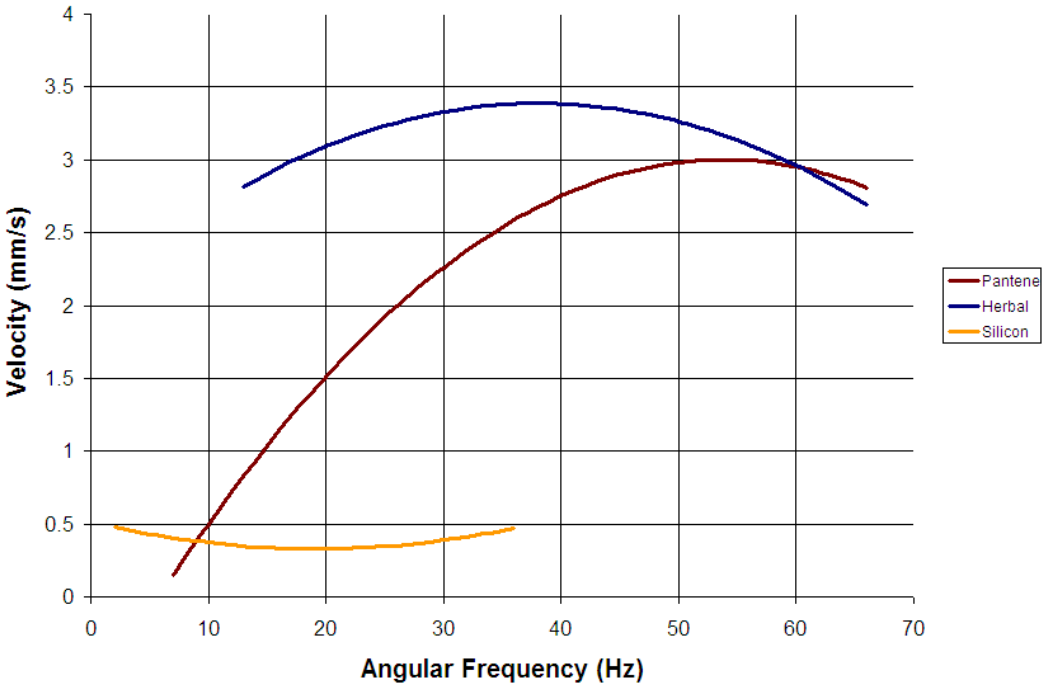

Figure 4.15: Combined final tethered swimmer velocity trend lines

Combining the trend lines in Figure 4.15 shows that the swimmer does in fact propel faster through non-Newtonian fluids, though the gap of control data, and the high standard deviations also prove this experiment ineffectual to confirm definite propulsion through non-Newtonian fluids.

4.6. Snail Results

The snail was tested in the three micellar solutions, whose rheometry is described in Section 2.1.4, and in Glycerin as a control, from 10 to 275 Hz. Though the snail's velocity through Glycerin was expected to be negligible, Figure 4.16 shows nominal locomotion throughout the range of frequencies, most likely due to vibration, irregularities in the fluid and fluid depth, an imperfect distribution of weight between the buoys, and possible velocity measurement before the swimmer reached equilibrium.

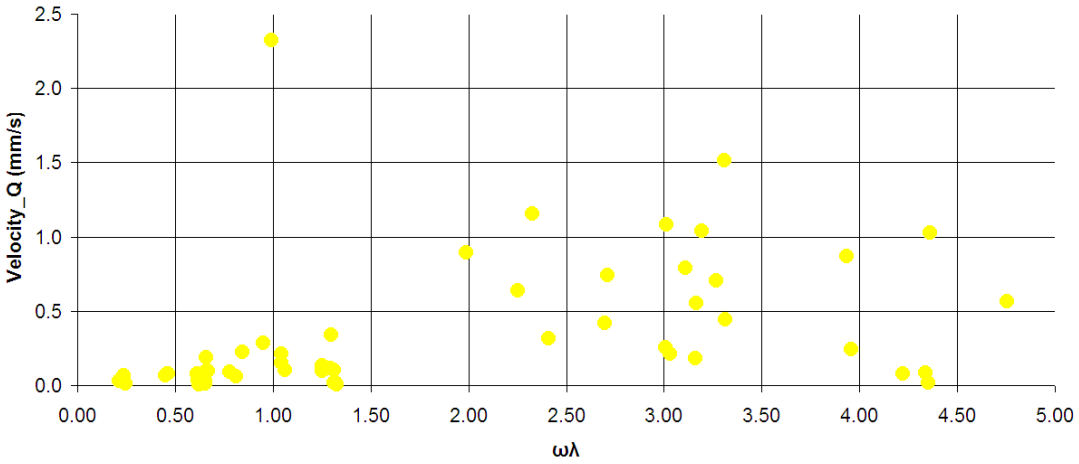

Figure 4.16: Snail velocity in Glycerin

The correctly predicted parabolic relationship between frequency and velocity in the 50/25 micellar solution, shown in Figure 4.17, displays a Deborah number of 14 at its maximum velocity and proves that the scallop theorem does not hold up in non-Newtonian fluids.

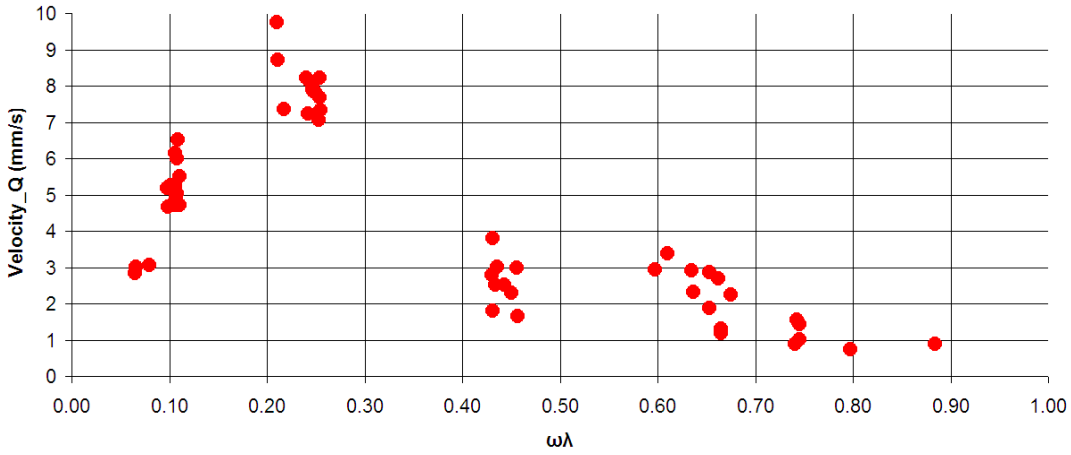

Figure 4.17: Snail velocity in the 50/25 micellar solution

Though the test results in the 100/50 micellar solution, shown in Figure 4.18 displays a semi-parabolic relationship between the frequency and velocity, translation rates were small and inconsistent, proving the results inconclusive.

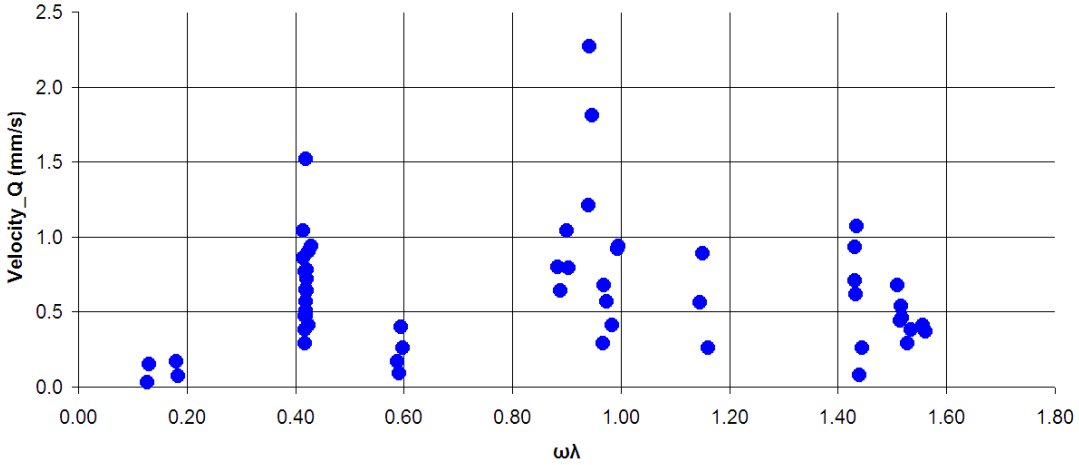

Figure 4.18: Snail velocity in the 100/50 micellar solution

The complete lack of correlation between angular and linear velocity of the snail in the 200/100 micellar solution, displayed in Figure 4.19, proves that experiment completely inconclusive.

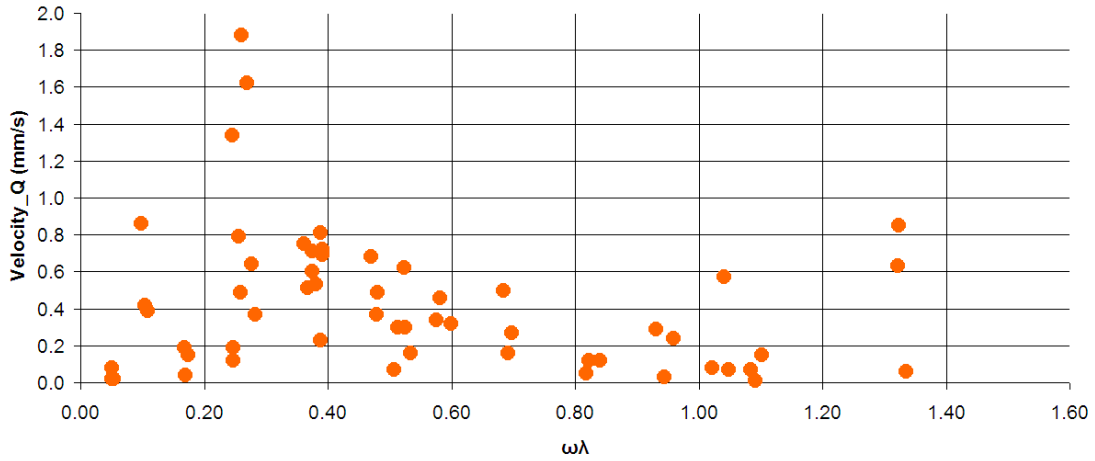

Figure 4.19: Snail velocity in the 200/100 micellar solution

Combining the above three experiments, onto Figure 4.20 shows the overall inconsistencies of the snail's linear and angular velocity motion and measurement in addition to the differences between the micellar solutions. This graph confirms the negative validity assessment of the 100/50 and 200/100 experiments, and thus, though the

50/25 micellar tests showed decent parabolic form, the combination of results and lack of further data prove the experiments inconclusive to prove that shear thinning locomotion is possible in non-Newtonian fluids.

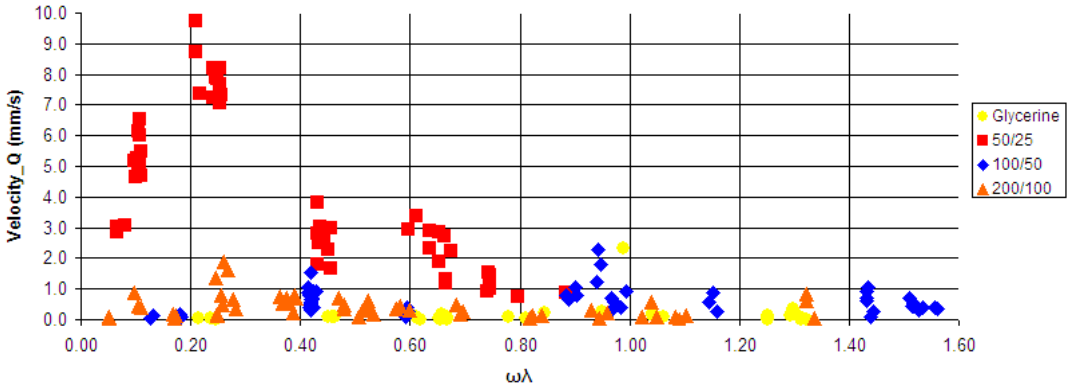

Figure 4.20: Snail velocity graph of all micellar solutions

5. Conclusion and Outlook

Although the autonomous elastic swimmer fulfilled its design requirement of rotating a mechanically isolated control surface through a test fluid, its confined motor could not dissipate heat and thus proved the robot incapable of performing long enough to record any experimentation. A heat sink or coolant to prevent motor burn out, or a stronger motor, which would work much below its capacity, could be implemented into future models of this robot. Reducing the weight of the cone, by hollowing it out, could also reduce the strain on the motor and would increase stability and buoyancy as well. Finally, use of a more viscous control fluid is advised to reduce the currently Reynolds number.

A tethered swimmer was designed as a replacement, which would simplify testing by exposing the motor, but risk biased by the frictional effects of mechanical contact. A second model was designed and fabricated to reduce friction and extend the frequency range closer to two orders of magnitude. Although both robots performed as intended, their limited frequency ranges, and the insubstantial correlation between angular frequency and velocity, deemed the experiment inconclusive to display elastic locomotion.

It is recommended that further test be performed at much lower and higher voltages to increase the swimmer's frequency range, though the motor did heat up substantially at 20V and may require coolant or cooling fans. In order to increase the repeatability and accuracy of the velocity testing, it is suggested that the thick encoder cables be replaced by thin wires or by a receiver and transmitter, as they may have tugged at the swimmer during testing. Finally, as the swimmer's frequency was much smaller in Silicon oil than in the shampoos, it is recommended that a less viscous control substance

be used.

Despite the fact that the snail robot, in the shear thinning experiments, did not translate through glycerin, velocity testing in the micellar solutions proved inconclusive as little correlation was found between frequency and the near-zero velocity.

As the Styrofoam buoys may have bulked over time when the test fluids managed to soak through the cellophane wrapping, the depth of fluid below the snail may have decreased or disappeared, slowing and impairing locomotion. Future designs could benefit from hollow impenetrable molded plastic buoys, which could also increase buoyancy and stability. In addition, fixing the center rail to the center of the buoys could prevent the unequal weight distribution with possibly lead to a depth difference between the two buoys. A linear actuator could also sooth the snails vibrations from by the heavy rotating cam. Although the manufactured Delrin stilts, to evenly set the fluid's thickness below the snail, did not stick to the Styrofoam, were unstable, and were thus not used in the experiments, it is recommended that future plastic buoys do employ such a feature. In addition, a buoy diameter ratio, larger that the 5:3 configuration used in this experiment is suggested to accentuate the snail's locomotion. Finally, testing in thicker fluids, at lower speeds, to reduce bubbling and thus fluid inconsistencies could have eliminated further error

References

[1] Bhardwaj A, Miller E, Rothstein J: "Filament stretching and capillary breakup extensional rheometry measurements of viscoelastic wormlike micelle solutions". J. Rheology, 51(4), 2007, pp. 693-719.

[2] Cohen I, Kundu P: Fluid Mechanics. 4th edition, Amsterdam; Boston; Academic Press, 2008.

[3] Chhabra R, Richardson J: Non-Newtonian Flow and Applied Rheology. 2 Edition, Amsterdam; Oxford: Butterworth-Heinemann, 2008.

[4] Fox R, McDonald A, Pritchard P: Introduction to Fluid Mechanics. 6 Edition, Hoboken, N.J.: Wiley, 2004

[5] Harris, J: Rheology and Non-Newtonian Flow, London; New York: Longman, 1977

[6] Lauga E, Powers T: "The hydrodynamics of swimming microorganisms", Reports on Progress in Physics *Rep. Prog. Phys.* 72, 096601, 2009.

[7] Metzger, T: The Rheology Handbook. 2. Edition, Hannover; Vincenz Network, 2006.

[8] Morrison, F.A.: Understanding Rheology, Oxford University Press, New York, 2001.

[9] Purcell, E.M.: "Life at Low Reynolds Number". American Journal of Physics, Vol. 45, No.1, 1977, pp. 3-11.

[10] Tanner, R: Engineering Rheology, Oxford: Claredon Press; New York: Oxford University Press, 1988

[11] Varagnat, M: "Instabilities of jets of non-Newtonian fluids impacting a plate, Massachusetts. Masters Thesis, Massachusetts Institute of Technology

Appendix

A.1. Appendix A: Basic Stamp Code for Control

A.1.1.Code for the autonomous normal stress experiment

```
' {$STAMP BS2}
' {$PBASIC 2.5}
Speed                                   VAR Byte
Direction                               VAR Byte
N                                       VAR Word
RotationSpeed                           VAR Word

    SEROUT 2,84,[$80,2,0,0]             ‘define motor controller "0"
    SEROUT 2,84,[$80,2,1,21]            'ERRORMULT
    SEROUT 2,84,[$80,2,2,1]             'ERRORDIV
    SEROUT 2,84,[$80,2,3,3]             'INTEGMULT
    SEROUT 2,84,[$80,2,4,4]             'INTEGDIV
    SEROUT 2,84,[$80,2,5,16]            'DERIVMULT
    SEROUT 2,84,[$80,2,6,1]             'DERIVDIV
    SEROUT 2,84,[$80,2,7,72]            'PIDRATE

again:
    DEBUG CR,"Speed (0-127):   "        'enter speed (0-127)
    DEBUGIN DEC Speed
    DEBUG DEC Speed,CR
    DEBUG "Direction (0/1): "           'enter direction (0 or 1)
    DEBUGIN DEC Direction
    DEBUG DEC Direction,CR
    SEROUT 2,84,[$80,0,Direction,Speed]     'send the command, 9600 baud

    PAUSE 2000
    COUNT 3,1000,N              'measure number of counts on pin 3 in 10000 mS

    RotationSpeed = N/100*60        'encoder wheel has 100 slots for 1 revolution
```

```
    DEBUG  "N:  ",DEC N,CR                          'max of N has to be less than 65535
    DEBUG  "RotSpeed:  ",DEC RotationSpeed,CR       'display speed in rpm

GOTO again
```

A.1.2.Code for the tethered normal stress experiment

```
' {$STAMP BS2}
' {$PBASIC 2.5}

Speed           VAR Byte
N               VAR Word
RotationSpeed VAR Word

again:
    DEBUG CR,"Speed (0-64):"          'enter speed (0-64)
    DEBUGIN DEC Speed
    DEBUG DEC Speed,CR

    SEROUT 14,84,[1,"f",Speed]        'outputs the speed

    'PAUSE 1000

    'COUNT 1,1000,N              'measure number of counts on pin 1 in 1000 mS

    'RotationSpeed =N/100        'encoder wheel has 100 slots for 1 revolution

    'DEBUG  "N:  ",DEC N,CR                 'max of N has to be leass than 65535
    'DEBUG  "RotSpeed:  ",DEC RotationSpeed,CR       'display speed in rpm

GOTO again
```

A.1.3. Code for shear stress experiment

```
' {$STAMP BS2}
' {$PBASIC 2.5}
```

```
Speed VAR Byte
again:
DEBUG CR,"Speed (0-64): " 'enter speed (0-64)
DEBUGIN DEC Speed
DEBUG DEC Speed,CR
SEROUT 14,84,[$80,1,"f",Speed]
GOTO again
```

A.2. Appendix B: MATLAB Code for Image Processing

A.2.1. Code for the normal stress experiment

```
clear all
close all                                   %clears all previous data

numpics=3;                                  %number of pictures per experiment

dirName = 'C:\Documents and Settings\Dor Ashur\My Documents\Research\New
Elastic Swimmer\Velocity test pictures';
fnames = dir(dirName);
fnames = fnames(~[fnames.isdir]);
nfids = length(fnames);
vals = cell(nfids, 1);

for K = 1:nfids
 vals{K} = imread(fnames(K).name);  %notice not K+2
end

numtests=nfids/numpics;
a=(1:nfids);
a=reshape(a,numpics,[]);

for n=1:numtests
   scrsz = get(0,'ScreenSize');
   figure('OuterPosition',[1 1 scrsz(3) scrsz(4)])
   title(['Test Number ',num2str(n),])
```

```
   subplot(1,3,1), subimage(vals{a(1,n)}) % makes a 3x1 matrix of images

   axis off
   title('Picture 1')
   subplot(1,3,2), subimage(vals{a(2,n)})
   axis off
   title('Picture 2')
   subplot(1,3,3), subimage(vals{a(3,n)})
   axis off
   title('Picture 3')
   truesize(.35*size(vals{a(1,n)}))

% scaling and locating

[x_click, y_click] = ginput(5);
close(1);

% distance between those last two points in pixels
pixel_distance = sqrt((x_click(2)-x_click(1))^2 + (y_click(2)-y_click(1))^2);
actual_distance = 152;              %distance btw selected points in mm.
scale_factor = actual_distance/pixel_distance;

v(1)= sqrt((x_click(4)-x_click(3))^2 + (y_click(4)-y_click(3))^2);
v(2)= sqrt((x_click(5)-x_click(4))^2 + (y_click(5)-y_click(4))^2);
v(3)= sqrt((x_click(5)-x_click(3))^2 + (y_click(5)-y_click(3))^2);
velocity(n)= mean(v)*scale_factor;
velocity_error(n)=std(v);
end
save new_elastic_swimmer.mat
```

A.2.2.Code for the shear stress experiment:

```
clear all
close all
```

```
%--------------------------------------------------------------------------
% PARAMETERS - These are the values to be changed to get
% this to work with the video.

npoints = 2;          %The number of points to track.

thresh = 180;         %Threshold for converting grayscale image to black and white.
Each
        % pixel in a grayscale image has a value from 0-255. The threshold
        %command, to be used later, converts any pixel with a value above
        %'thresh' to pure white (255), any value below that to pure black (0).

animate = 1;          %Set to 1 to show the video, 0 to skip the animation and make the
        %program run faster

%--------------------------------------------------------------------------

% load video file as .avi
disp('Loading video file...')
obj = mmreader('MVI_1180.avi');
disp('Loading complete.')
s = read(obj, 1);                                   % read frame one

% determinine framerate
nframes = get(obj, 'numberOfFrames');               %number of frames in video
info = mmfileinfo(sprintf ('MVI_1180.avi'));
fps = nframes/info.Duration;                        %calculates frames per sec
dt  = 1/fps;                                        %time between each frame

% cropping video
disp('Opening figure window...')
imshow(s,1);                                        %display the 1st image
title('Crop Video and Scale')
xlabel('Click 2 points to crop (top to bottom) followed by 2 points of known distance
for scaling');
```

```
[x_crop, y_crop] = ginput(4);                    %records 4 clicked pixel
locations.
close(1);

% scaling from pixels to mm
pixel_distance = sqrt((x_crop(4)-x_crop(3))^2 + (y_crop(4)-y_crop(3))^2);
actual_distance = 560;                           %physical distance between
selected points in mm.
scale_factor = actual_distance/pixel_distance;

% loop through the rest of the video to calculate velocity
for i = 1:nframes



%editing the image to make it discernable
   time(i) = dt*i;                               %increment time vector
   im1 = read(obj,i);                            %reads all frames
   im1 = im1(int16(y_crop(1)):int16(y_crop(2)),:,:);          %crops frames
   im2 = rgb2gray(im1);                          %converts frames to grayscale
   im2 = im2>thresh;                             %set a threshold value
   se = strel('disk',1);
   im3 = imerode(im2,se);                %erodes buoys in all frames to get rid of
loose pixels
   se = strel('disk',10);
   im4 = imdilate(im3,se);                %dilates buoys in all frames to make them
circular
   [im5,num] = bwlabel(im4,8);             %labels the now b&w image as
"background" and
         %"objects". Each pixel of the black background is a
         % 0. each white blob will have a corresponding
         %number for its pixels
   %finding the areas of the blobs
   props = regionprops(im5,'Area','Centroid');  %finds  properties of an image frame.
   [A max_i] = sort([props.Area]);              %sorts from smallest to blob to largest
blob
```

```
%finds the centroids of the buoys
cent1 = props(max_i(end)).Centroid;
cent2 = props(max_i(end-1)).Centroid;

x1(i) = cent1(1);                                   %x and y for big buoy
y1(i) = cent1(2);
x2(i) = cent2(2);                                   %x and y for small buoy
y2(i) = cent2(2);

%show animation of moving centriods for verification
if animate == 1
   if i == 1;
      disp('Opening animation window...');
   end
   figure(1)
   if i == 1;
      disp('Analyzing video, plotting real-time data.');
   end
   subplot(2,1,1)
   imshow(im1)
   hold on
   title('Original Image')

   hold off
   subplot(2,1,2)
   imshow(im5)
   hold on
   title('Filtered Image')
   xlabel(['Frame ',num2str(i),' of ',num2str(nframes)])

   plot(x1,y1,'*')                                  %plots the centroid points
   plot(x2,y2,'*')

else
```

```
        skip animation
        clc
        disp(['Analyzing frame ',num2str(i),' of ',num2str(nframes),'.'])
    end
end


% convert x and y pixel values to  mm
x1_act = x1*scale_factor;
y1_act = y1*scale_factor;
x2_act = x2*scale_factor;
y2_act = y2*scale_factor;

%velocity calculation
Q = [(x2_act-x1_act);(y2_act-y1_act)];          %crossbar normalizing vector

for i = 1:length(Q)
    Q(:,i) = Q(:,i)/norm(Q(:,i));               %normalizing Q
end

% velocity of big buoy
for k=1:(nframes-1)
    velocity_x(k)=(x1_act(k+1)-x1_act(k))/dt;   %x component of velocity
    velocity_y(k)=(y1_act(k+1)-y1_act(k))/dt;   %y component of velocity
    time_vel(k)=time(k+1);
end

%velocity in the Q direction
Vq = dot([velocity_x,0; velocity_y,0],Q);
Vq_smooth = smooth(smooth(smooth(Vq)))';        %smooths out the result

%average velocity in mm/s
avg_velocity_x = mean(velocity_x);              %in x direction
avg_velocity_y = mean(velocity_y);              %in y direction
```

```
avg_velocity = mean(Vq');                    %in Q direction
avg_Vel = avg_Velocity/1000;                 %in m/s

l=0;
for k=1:(nframes-2)
   if Vq_smooth(k)<10 && Vq_smooth(k+1)>10 && Vq_smooth(k+2)>10
      l=l+1;
   end
end

frequency=(l/(info.Duration))*60;            %frequency in rpm

figure(1)
plot(time,Vq_smooth)

save MVI_0001.mat
```