Spring 1-1-2011

Analog Physical Experiments to Investigate Mechanisms Controlling Enhanced Ice Flow by Basal Sliding

Michael Kirk Records
University of Colorado at Boulder, records@colorado.edu

ANALOG PHYSICAL EXPERIMENTS TO INVESTIGATE MECHANISMS CONTROLLING ENHANCED ICE FLOW BY BASAL SLIDING

by

MICHAEL KIRK RECORDS

B.S., University of Colorado Boulder, 2009

A thesis submitted to the

Faculty of the Graduate School of the

University of Colorado in partial fulfillment

of the requirement for the degree of

Master of Science

Department of Civil Engineering

2011

This thesis entitled:
Analog Physical Experiments to Investigate Mechanisms
Controlling Enhanced Ice Flow by Basal Sliding
written by Michael Kirk Records
has been approved for the Department of Civil, Architectural, and Environmental Engineering

_____
Harihar Rajaram

_____
Robert S. Anderson

Date_____

The final copy of this thesis has been examined by the signatories, and we find that both the content and the form meet acceptable presentation standards of scholarly work in the above mentioned discipline.

Abstract

Records, Michael Kirk (M.S., Department of Civil, Environmental, and Architectural Engineering)

Analog Physical Experiments to Investigate Mechanisms Controlling Enhanced Ice Flow by Basal Sliding

Thesis directed by Dr. Harihar Rajaram

Numerous field observations document enhanced ice flow of terrestrial glaciers during brief periods of increased delivery of water to the bed. Contemporary understanding of the effect of basal water fails to consistently predict these speedup events. We carried out laboratory experiments using a transparent silicone (PDMS) as an analog for glacier ice, to investigate how basal water enhances ice velocities. The PDMS was allowed to flow over a rough checkerboard bed topography inside a tilted rectangular channel. Water was injected into a basal water system beneath the PDMS. We tested various configurations of the basal water system, including linked cavity and conduit systems, and measured basal water pressure, storage, and discharge. Velocity fields at the PDMS surface and bed were calculated by tracking bead markers. Transient and steady-state experiments were conducted to define the water pressure conditions associated with enhanced sliding. We evaluate the implications of the experimental results on mechanisms controlling enhanced ice flow. Our results suggest that: (i) bed separation enhances ice flow by reducing basal drag; (ii) changes in water pressures affect ice flow rates by changing cavity geometry, and (iii) the development of an efficient conduit system reduces the extent of water at the bed, thus slowing enhanced ice flow.

Acknowledgements

I would like to sincerely thank my primary advisor Harihar Rajaram for his guidance, advice, amazing patience, positive attitude, and perseverance throughout this entire project. Robert Anderson's insightful comments, enthusiasm, and creativity in solving several challenging lab puzzles were crucial to the success of this project. I would also like to thank my other committee member, Tad Pfeffer for his helpful comments about my thesis.

In the Water Resources graduate student office I would like to particularly thank Mike Soltys for his willing and endless help on everything from coding to laboratory challenges. Jose Solis gave me invaluable assistance on a seconds notice in the Environmental Fluid Mechanics Laboratory, I will always appreciate this. The constantly positive and inspiring attitudes and useful suggestions of Masoud Arshadi, James Cullis, and Scott Griebling were an invaluable asset to me every day of my master's degree.

Funding for this project was generously supplied by a seed grant to Harihar Rajaram from the Department of Civil Engineering at the University of Colorado at Boulder and Robert Anderson's NSF EAR/GLD grant 1123855. I would like to thank John Crimaldi for providing one of the DSLR cameras I used for my experiments.

Finally, I would like to thank my family, friends, and girlfriend Rachel for their endless patience and support over the last two years.

Table of Contents

Tables

Figures

1 Introduction

Almost half of the recently observed doubling in the rate of sea level rise is due to increased mass loss from glaciers and ice sheets (IPCC, 2007). Field observations suggest that the Greenland Ice Sheet's (GIS) mass loss rate has doubled over this same time period, and that about half of this loss is due to accelerated ice flow (Zwally, 2002; Rignot, 2006; van den Broeke et al., 2009). At the same time, many smaller terrestrial glaciers have been observed to accelerate during brief periods of intense water delivery to the bed (Iken and Bindshadler, 1986; Anderson et al., 2004; Harper et al., 2007). Enhanced basal motion due to increased surface water delivery to the bed is believed to be one of the important mechanisms responsible for enhanced ice flow (Zwally et al., 2002). In ice flow models, basal sliding velocity is generally expressed as a function of the difference between basal water pressure and the overburden ice pressure (e.g., Paterson, 1994; Van Der Veen, 1999). However, even on smaller glaciers, these formulations fail to consistently predict the magnitude and duration of these basal speedup events (Paterson, 2004; Van der Veen, 1999).

Predicting the relationship between water input to the bed and basal sliding is challenging because basal sliding is a complicated process influenced by several other coupled processes. In addition, water at the bed exists within a highly complex system of links and cavities whose geometry is constantly evolving in space and time due to melt, changing basal water pressures, and sliding. These complex spatial and temporal variations make predicting basal sliding velocity as a function of modeled or observed basal water pressures very difficult. As the climate continues to warm, understanding the fundamental mechanisms linking basal water delivery to the bed, enhanced basal motion, and enhanced ice flow is becoming increasingly important for predicting future glacial contributions to sea level rise.

Field observations show that higher ice velocities preferentially occur during early summer water input events than late summer events; and that thicker glaciers, where cavities collapse more quickly, show more speed up events (Anderson et al., 2004; Bartholomaus et al., 2007; 2011; Harper et al., 2007). These observations point to a relationship between basal water input and enhanced sliding that dependent on both the rate of input of water to the subglacial system, and the state of the basal cavity system. The usual interpretation of these observations is that in early summer the basal water system is dominated by a hydraulically inefficient (referred to hereafter simply as inefficient) linked cavity system,

while as the summer progresses, the system becomes dominated by a hydraulically efficient (referred to hereafter as efficient) tunnel system. Thus, during early summer events, even small water inputs overwhelm the capacity of the basal water system to transmit water, thus pressurizing it; but as the system evolves, efficient passage of large water inputs is possible at lower water pressures.

We used a three-dimensional laboratory analog model to reproduce short duration water inputs to the glacier bed, the evolution of the basal cavity system, and accelerated ice flow. A transparent high viscosity silicone, Polydimethylsiloxane (PDMS), was used as an ice analog. The unique transparent quality of the PDMS allowed investigation within and under the model glacier. Because PDMS is much less viscous than ice, glacial space and time could be modeled on a much shorter laboratory space and time scales. To represent surface melt delivery to the bed, we injected pressurized pulses of dyed water at the model bed to identify how basal water enhances ice flow. We performed a sequence of experiments to explore the relationship between the evolution of the basal water system and enhanced ice flow. These experiments were designed to address the effects of steady state and transient pressurized water pulses that encounter an inefficient cavity network and an efficient basal conduit system. Time series for basal water pressure, basal water storage, rate of change of basal water storage, basal sliding velocity, and surface velocity were collected. The controlled conditions of our experiments, which can never be achieved on real glaciers, allow clear interpretation of the coupling between basal water and PDMS flow.

2 Experimental Model

2.1 Analog Modeling

Analog models have been widely used to simulate geological and geomorphological processes. PDMS has a long history of use in modeling geologic and geomorphological processes (Weijermars, 1986), and more recently has successfully been used to model ice flow dynamics (Corti, 2008) and basal water-induced enhanced ice flow (Catania et al., 2009). Through correct scaling, the laboratory model can be used to predict the behavior of the real world process of interest (referred to as the prototype). Experiments were carried out at the Civil Engineering Department of the University of Colorado Boulder.

PDMS is a clear, non-toxic fluid with no yield stress and a Newtonian rheology under laboratory strain rates (Boutelier, 2007). Because it is clear, internal deformation and basal water extent can be clearly visualized through it. The Newtonian rheology of PDMS differs from the power law stress-strain relationship of ice (Paterson, 1994), but it is still useful for investigating the primary effects of the response of ice to basal water perturbations. The relatively low viscosity of PDMS (24400 Pa s) measured by a falling sphere viscometer) allowed experimental durations on the order of minutes and hours to simulate glacier processes that take days or months to observe in the field, at a significantly reduced cost. Analog modeling allowed levels of control and monitoring that could never be achieved in fieldwork, including control of bed geometry and monitoring of both water inputs and outputs.

Table 1: Model and prototype parameters

Parameter	Model	Nature
Density (kg/m^3)	965	920
Gravity (m/s^2)	9.81	9.81
Thickness (m)	.051	500
Viscosity (Pa sec)	24400	5E15
High Diurnal Velocity (m/sec)	6.3E-5	8.3E-6
Low Diurnal Velocity (m/sec)	5.1E.5	5.0E-6
$V_{enhanced}/V_{background}$	1.24	1.64

2.2 Experimental Setup

All experiments were run inside a 60 x 30 x 15 cm acrylic flow channel. Within the channel, flow dynamics were monitored inside of a study area centered over the channel center; at 23 cm wide and 30 cm long it was sized to be free of significant edge and end effects. The channel width to PDMS depth ratio was set at 1:10, so that the middle 50% of the channel becomes essentially free of sidewall effects, according to analytical solutions to velocity profiles in rectangular cross-sections (Langlois, 1964). As end effects were assumed to act over three times the PDMS depth (Kamb, 1986), the upstream and downstream 15 cm of the channel were excluded from the study area.

Basal topography was constructed on the channel bed to provide resistance against basal sliding and control the extent and pattern of basal water. The simple, but realistic topography consisted of dome-shaped plasticine bumps, which were arranged in a checkerboard pattern (see Figure 1). With widespread basal sliding, distinct low pressure water-filled cavities developed on the downstream side of each bed bump. The water-filled cavities are visible as the dark surfaces on the downstream side of the bed bumps in Figure 1.

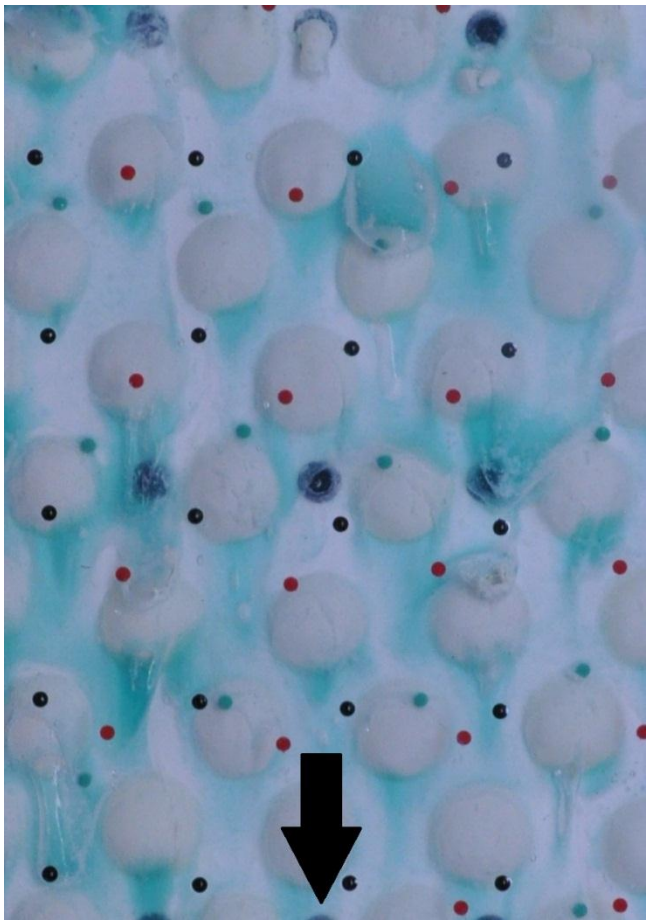

Figure 1: Experiment sample. Flow direction is shown by the black arrow, the large white disks are the bed bumps, the light blue areas are dyed basal water. The small black, green and red dots are marker beads. The larger black dots are basal water ports.

Two topographies were used, each with the same bump height and designed to generate the same sliding resistance and basal water head loss. The difference between the two topographies was the percent of the bed that they covered. The first topography (CB1) covered 37% of the bed, while the second topography (CB2) covered 60% of the bed.

Because the PDMS is very sticky, it was necessary to apply a lubricant to the entire bed. The lubricant performed three roles: i) It prevented PDMS from sticking to the bed when the water cavities closed; ii) It allowed basal water to collect between the bed and the PDMS, and iii) It supported basal sliding by breaking the no slip condition between the bed and the PDMS layer. The lubricant, a two part system composed of synthetic grease covered by a wetting glycerin, was designed to provide lubrication for the duration of each experiment. The lubricant was significantly more viscous than water, but it

provided a rudimentary analog for a prototype water film that is believed to exist under temperate bed regions in glaciers (Paterson, 1994).

A water supply system was constructed to provide water to the channel bed at known flow rates and pressures. Water from four constant head reservoirs (CHRs) entered the channel bed at 12 distributed basal water ports arranged in a grid of four rows of three ports (see Figure 2). Each CHR provided water to one row of water ports. This distributed water supply system allowed a relatively constant hydraulic gradient to be maintained down the length of the channel bed. Water pressures at the bed were monitored continuously through eight pressure taps. Pressures were logged at each 5 second time step by a camera pointed at a manometer board connected to the eight pressure taps. The manometer camera was synced to fire at the same time as the overhead camera. Recognizing that the entire bed was not hydraulically connected to the basal water system, pressure taps that showed no fluctuation in pressure with time were removed from the analysis. A representative basal water pressure, referred to hereafter as water pressure, was calculated as the average water pressure in the active pressure taps. Water discharge from the model terminus was directed through an outflow channel to a balance were the outflow rate was measured gravimetrically. The flow channel, outflow channel, and basal water ports are shown in Figure 2 below.

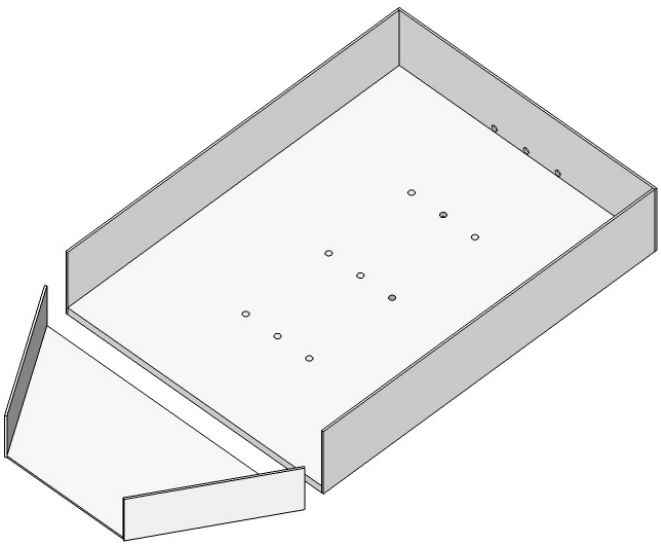

Figure 2: Experimental channel, outflow channel & basal water ports.

Marker beads were used as tracking particles beneath, within, and at the surface of the PDMS. Their positions were tracked by the overhead camera, which fired every five seconds. Marker beads are visible as the small dark dots in Figure 1**Error! Reference source not found.**. Each tracking particle was followed through time using modified particle image velocimetry (PIV) algorithms developed for tracking bacteria colonies over time (Crocker, 1999; Dufresne, 2005). From the particle tracks, displacements and velocities were calculated for each particle. Representative surface and basal sliding velocities were calculated as a linearly interpolated average velocity of the eight beads closest to the center of the study area. By using representative velocities as defined above, local effects associated with variations in the bed topography were removed, and errors associated with image resolution were reduced. Representative surface and basal sliding velocities are referred to respectively as surface velocity and sliding velocity hereafter.

Basal water extent and volume were quantified using the overhead camera. To improve imaging, the water was dyed blue with FD&C Blue dye. The extent (area covered by basal water) was quantified by filtering the overhead imagery for water color then finding the percent of bed where water existed. The thickness of the water filled basal link-cavity system was calculated following the work of Detwiler et al. [1999], which is based on the Beer-Lambert law, which relates the absorption of light to the properties of the material through which it is passes (Rossiter and Baetzold, 1993). A relationship (Figure 3) between basal water thickness and light intensity observed in a pixel within the camera image was developed by photographing a water filled wedge of known geometry.

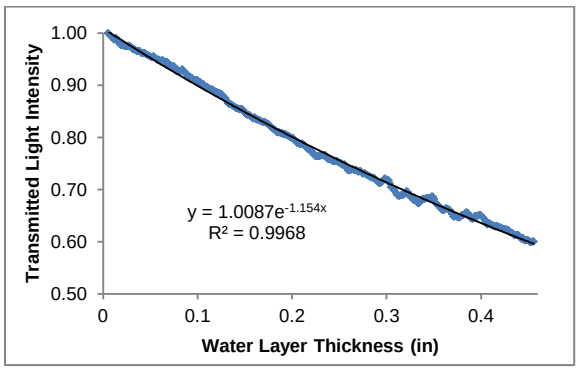

Figure 3: Transmitted light intensity vs. dyed water layer thickness; transmitted light intensity is the unitless ratio of incident light to transmitted light.

The thickness of the water layer at the channel bed was quantified at each pixel in the overhead camera images. By integrating the individual pixel level water thickness over the entire study area, the total volume of water stored at the bed, referred to hereafter as storage, was calculated. The rate of change of storage (dS/dt) was calculated between each time step. dS/dt physically represents how quickly the basal water system is filling with or draining water which is accomplished by changing basal cavity geometry.

2.3 Scaling

Scaling focused on the physical processes that are known to control the ice dynamics of interest. PDMS has roughly the same density as ice and satisfies the important constraint that it is less dense than water (965 kg/m^3), so that water tended to flow under the PDMS. In terms of viscosity, the PDMS was significantly more viscous than water (9 orders of magnitude), so it maintained the necessary large viscosity contrast. Cavity growth and closure, which control the volume of the water at the bed, were controlled by the ratio of water pressure to overburden pressure (floatation fraction). Model floatation fractions were set to closely mimic observed prototype floatation fractions and were used as a metric of experimental validity. Recognizing that dS/dt is clearly related to surface velocity enhancement (Barthalomaus et al., 2007; 2011), the duration of the high pressure water events was set to be long enough to allow change in cavity geometry, as can be seen in Figure 8.

Complete geometric scaling of PDMS depth to bump height was not feasible because the PDMS depth was dictated by the width to depth aspect ratio; at a PDMS depth of 5 cm, geometrically scaled bed topography bumps would have been too small to control the water extent. Concerned that out-of-scale topography would mask the physical mechanisms of widespread uniform basal sliding and cavity formation, we checked that no significant velocity variations were induced by the out of scale bed topography. Local effects (i.e. PDMS flow over crests vs. troughs) were observed not to overwhelm widespread uniform sliding or uniform surface velocities. Also, as cavity growth and closure were controlled by floatation fraction rather than by bump size, an out-of-scale bump size to analog ice depth ratio did not affect this process.

3 Experimental Suite

We performed four experiments with varied bed topographies, water systems, and water pressures. All experiments had in common a PDMS depth of 5 cm, an 8° bed slope, checkerboard bed topography, and an initial 15-minute period of steady state (SS) water pressure. The SS period was designed to be long enough to ensure that the basal water system came to steady state both in terms of water pressure and cavity evolution before the SS velocities were calculated. In one of the experiments, the steady state period was followed by a hydrologic perturbation. The four experiments are described below and shown visually in Table 2.

Table 2: Experimental Suite Summary

Experiment	Lubricant Extent	Water Extent	Mechanism Investigated
Background One-Layer PDMS Flow	None	None	Background representative velocity obtained
CB1 Sliding	Complete lubrication	Water throughout linked cavity system	1) Effect of bed separation on enhanced ice flow. 2) Effect of equal high pressure water pulses on ice flow.
CB2 Sliding	Complete lubrication	Water throughout linked cavity system	Different basal topography for comparison w/ CB1
CB2 Patchy Sliding	Only bed depressions lubricated	Water throughout linked cavity system	1) Can widespread water filled linked cavities exist with basal sliding only above bed depressions? 2) Can patchy basal sliding produce enhanced ice flow?
CB1 Developed Conduit	Complete lubrication	Water only in conduits	Effect of small, high flow rate conduits on ice flow.

3.1 Background 1 Layer Flow Experiment

This experiment provided a reference velocity (no sliding, purely internal deformation) for comparison with all other experiments. It used the CB1 bed topography described in the Experimental Setup. No lubricant was placed at the bed, so that the PDMS stuck to the bed, producing a no-slip

condition at the bed-ice analog interface. No water was present at the bed and no transient water events were imposed.

3.2 CB1 Developed Conduit Experiment

In this experiment, we investigated the effect of small, high flow rate basal water conduits on ice flow. We employed the CB1 bed topography described in the Experimental Setup section above, but incised conduits into it. Basal water existed only within several small, efficient conduits that covered a low percentage of the bed. Analogous to a water film, the entire bed was lubricated so that widespread sliding could occur. No water was present at the bed outside of the conduits and no transient water events were imposed.

3.3 CB1 Sliding Experiment

In this experiment, we investigated: i) whether steady state bed separation enhanced ice flow, and ii) the effect of multiple transient high water pressure events on ice flow. Basal water was present throughout the linked cavity system, and was contained by the bed topography. Again, the entire bed was lubricated so that widespread sliding could occur.

3.4 CB2 Sliding Experiment

Similar to the CB1 Sliding experiment, in the CB2 Sliding experiment we investigated the effect of steady state bed separation on enhanced ice flow, but on a different bed topography (CB2).

3.5 CB2 Patchy Sliding Experiment

With the CB2 Patchy Sliding experiment, we investigated: i) whether widespread water filled linked cavities can exist with basal sliding only above bed depressions, and ii) whether patchy basal sliding can produce enhanced steady state ice flow.

3.6 Experimental Procedure

All experiments followed the same setup and run methodology. First, the channel was leveled and the lubricant system was brushed onto the bed in a uniform thin layer. Then, PDMS was laid in two 2.5 cm layers inside the channel and allowed to settle for 24 hours; this allowed any trapped large air bubbles to rise to the surface. Grids of tracking beads were placed at the bed, mid-layer within the PDMS, and at

the surface. Once the PDMS was settled and level, the channel was tipped, PDMS flow began, and water was immediately introduced at the bed. The tipping of the channel marked the beginning of the 15-minute SS period.

As can be seen in Figure 4, the 15-minute SS period allowed the basal water cavity system to reach to an equilibrium pressure regime. Figure 4 shows three distinct phases. First, an initial jump in water pressure occurs as water was introduced into the under-capacity cavity system. This spike in water pressures was followed by a period of dropping water pressures. During this phase, the cavity system evolved toward a steady state regime. Finally, water pressures reached a period of prolonged stability in which the cavity system and its ability to transmit water was at equilibrium with the water supply. This trend towards steady state water pressures and an adjusted cavity system was characteristic of all four SS experiments.

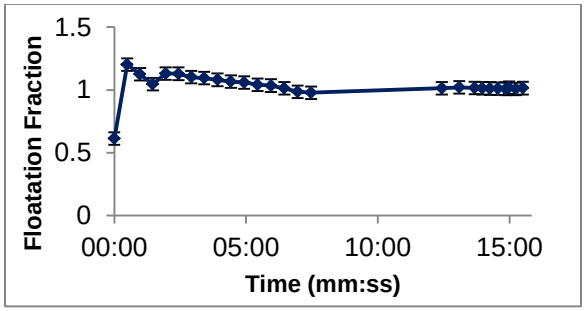

Figure 4: CB1 Sliding Experiment: Basal Water Pressure vs. Time, Transient Water Pressures Period

For CB1 Sliding, a transient water pressure period was run after the SS period. Transient water pressures were imposed by closing the valves between CHRs and the basal water supply, then reopening them. Thus, when the supply valves were closed low water pressures existed at the bed, and when the valves were reopened the basal water system was pressurized.

4 Steady State Experimental Results

SS results and a discussion are presented in chapter 4, followed by the results and a discussion for the transient experiment in chapter 5. Results for the four SS experiments are presented in Table 3.

Comparing velocities between all four experiments, CB1 Sliding and CB2 Sliding experienced the greatest surface enhancement of flow over the internal deformation case (by 45%); CB2 patchy sliding saw intermediate enhancement (35%), while CB1 Developed Conduit (25%) displayed the least enhancement. The surface speed increase seen in the CB1 Sliding and CB2 Sliding is significant because it was achieved without transient water pressures. To achieve this high level of surface speed enhancement, widespread basal sliding over the entire bed was necessary: bed velocities were consistent between areas with and without water. Without sliding over the entire bed, CB2 Patchy Sliding experienced a smaller surface speed increase, even at an unrealistically high water pressure. The non-local control on basal sliding seen in the uniform sliding velocity of the CB1 Sliding and CB2 Sliding experiments points to a net effect in which the individual cavities act together to produce a widespread uniform basal velocity perturbation.

Table 3: SS Experimental Results

Parameters		CB1 Sliding	CB2 Sliding	CB1 Conduit	CB2 Patchy Sliding
Representative Velocities (m/sec)	Surface	6.2E-05	6.2E-05	5.4E-05	5.8E-05
	Surface BG (No-slip equivalent)	4.3E-05	---	4.3E-05	---
	Bed (Water Filled Region)	1.0E-05	1.1E-05	7.E-06	1.1E-05
	Bed Water Free Region)	1.1E-05	1.2E-05	3.E-06	3.E-06
Basal Water Properties	Discharge (ml/sec)	5.9	8.8	14.1	0.9
	Floatation Fraction (P_w/P_e)	1.01	0.5	1.19	1.4
	Percentage of bed available to be covered by water	63	41	17	41
	Actual percent of bed covered by water	46	35	17	40

This uniform sliding was in general agreement with the conclusions of Balise and Raymond (1985), who theoretically predicted how velocity perturbations at the bed, induced over a distance L, should result in surface velocity variations for the case of a linearly viscous glacier of depth H. They found that velocity perturbations at the bed of length scale L<H were completely attenuated within the ice, while perturbations of length scale L > 10H, the full perturbation was seen at the surface. Our interpretation that cavities act together over a larger area to produce a widespread basal perturbation is consistent with the conclusions of Balise and Raymond (1985). In disagreement with traditional sliding laws (Hooke, 2005; Paterson, 1994), we found that the actual observed surface velocity was greater than the observed sliding velocity plus the calculated deformational velocity ($u_{actual} > u_{deformation} + u_{sliding}$). The surface velocity enhancement during sliding events was greater than just the contribution of the sliding velocity. We suggest that this is due to the fact that simply stacking the deformational velocity profile on top of the sliding velocity at the bed does not correctly represent the change in the boundary condition at the bed for a coupled two-layer flow.

Comparing the basal regimes of CB2 Patchy Sliding to the CB2 Sliding experiment, the key feature that stood out was that extensive basal sliding was necessary for widespread water coverage at realistic water pressures. Even with complete basal lubrication, widespread water coverage was necessary for enhanced ice flow as was seen by comparing CB1 Developed Conduit to CB1 Sliding. Comparing CB1 Sliding and CB2 Sliding, the same surface and sliding velocities were produced under significantly different water pressure conditions. This suggests that water pressure is not the only control over enhanced ice velocities; it appears that volume of water storage and the unique resistance to basal sliding by each different bed also have an influence.

Observed discharge of basal water from the analog terminus was studied as a metric of experimental validity. The developed conduit system supported the highest water flow rate, CB1 and CB2 Sliding supported intermediate flow rates, and CB2 Patchy Sliding supported the lowest discharge. The relatively high water discharge of CB1 Developed Conduit showed the success of the analog developed conduit system in representing a hydraulically efficient system. The fact that CB1 Developed Conduit showed the smallest increase in velocity, despite the highest water velocities, confirmed that the existence of high

velocity water over a small portion of the bed cannot enhance surface velocity significantly; and that high water pressures observed over a small fraction of the bed cannot produce enhanced ice flow.

Water pressure was used as a metric of experimental validity: experiments with water pressures near or below overburden were deemed to be more realistic (as measured water pressures in glaciers rarely exceed the ice overburden pressure). CB1 Developed Conduit was considered realistic despite water pressures above overburden because: i) melt was not present to help keep the conduits open, and ii) in the limited space of the model there were few flow paths to follow. This meant that if one of the conduits failed to stay open, the system was immediately stressed. The extremely high water pressures of the CB2 Patchy Sliding experiment were seen as a clear indication that patchy sliding was not a realistic mechanism to maintain a water-filled linked cavity system at the bed a glacier for extended periods.

In general, we consider the water pressures in the experimental suite to be higher than they would be in the analogous prototype because no melt was present to support cavity existence. When widespread basal sliding was present, basal water could exist at pressures below overburden pressure. The observation that lower water pressures over a large portion of the bed (linked-cavity) enhanced ice flow more than high pressure water confined to a few small conduits confirmed the work of Kamb (1987), who suggested that water pressures should be averaged over a substantial part of the bed when computing sliding velocities.

The results of the SS experimental suite identified a mechanism for enhanced sliding that did not require high water pressures or changes in water pressure. Rather, widespread low pressure water produced significant enhancement of surface velocity. The increase in surface velocity was produced by a smaller increase in a widespread basal sliding velocity. These widespread non-zero bed velocities are referred to simply as “sliding velocity” below. We attribute the enhanced sliding velocity to a decrease in the basal drag due to increased separation between the bed and the basal PDMS. However, because a true water film was not present at the bed, we cannot differentiate whether the reduced drag force was due to a reduction in the coulomb friction at the bed, or due to a reduction in form drag due to bed topography.

5 CB1 Sliding Transient Experimental Results

We present results and discussion of the transient water pressure experiment below. A series of pressurized water pulses were delivered to the model bed in an attempt to mimic a series of diurnal melt events. The pulses were interrupted by periods of no basal water input. To investigate the effect of the cavity closure timescale, the first three high water pressure events each had a duration of one minute and were spaced by one minute; the following three events each had a duration of two minutes and were spaced by two minutes. Figure 5(a) shows the series of pressurized basal water events (labeled 1-6) and the corresponding peaks in surface velocity. These occur shortly after each peak in water pressure.

5.1 Results

As seen in Figure 5(a), the pressurized water pulses produced a series of fluctuations of surface velocity, in which each surface velocity peak occurred shortly after each water pressure peak, followed by surface velocity and water pressure lows. A ratio of enhanced to background velocities of nearly 1.25 was observed in the model. While this is smaller than the ratio of 1.6 observed on the Kennicott glacier (Bartholomaus, 2007; 2011), it is still significant. We attribute the difference in ratios of enhanced flow between the model and the prototype to two components of the model's lubricant film. One, in the model, the film was more viscous than in the prototype, so unlike the water film it supported basal shear. Two, because the analog film cannot transmit imposed pressures, an increase in the pressure of the water supply is only felt where water exists at the bed, not through the entire film. So, an increase in water supply pressure does not drive a widespread decrease in effective pressure and basal friction as it would in the prototype.

Recognizing that the enhancements in surface velocity were directly caused by enhanced basal sliding, sliding velocity was used to investigate the mechanism that controlled the enhanced flow observed in Figure 5(a). Figure 5(b) confirmed the relationship between enhanced basal sliding and enhanced surface velocities, showing that each peak in surface velocity corresponded with a peak in sliding velocity. With the exception of event #3, the peaks in sliding velocity were observed to be translated from the bed to the surface over a time interval that was less than the resolution of the experimental monitoring system. The lag of peak sliding velocity behind peak surface velocity in event #3

was attributed to experimental error for event #3. By dimensional analysis, $H^2/\square$, where $\square$ is kinematic viscosity, was used to estimate the time a sliding velocity perturbation takes to reach the surface. As this gave a propagation timescale of 10^{-4} seconds, we conclude that transient velocity changes at the bed were transferred almost instantaneously to the surface. For the case of a prototype glacier of any realistic thickness this timescale will be roughly 10^{-10} seconds.

Observed characteristics of the basal water system were reviewed to determine which features controlled transient enhanced sliding. Figure 6(b) shows histories of sliding velocity and storage. For each event, storage peaked after sliding velocity peaked, especially for the last four events which had longer closure periods. The fact that peak storage volume lags peak sliding velocity suggests that periods of transient enhanced ice flow were not simply caused by an increase in bed separation due to increased storage.

Unlike storage, both water pressure and dS/dt peaked immediately before each peak in sliding velocity. The time series in Figure 5(c) suggest that no clear relationship exists between the magnitudes of peaks in water pressure and associated peaks in sliding velocity. With the exception of event #4, all sliding velocity peaks had very similar magnitudes. Sliding event #4 had the largest velocity despite having very a peak in water pressure that was similar to the following two peaks. Unlike water pressure, dS/dt showed peaks of similar magnitude for all events, shown in Figure 6(b). With the exception of event #4, the variations in the magnitudes of the peaks in dS/dt more clearly correspond to the sliding velocity peaks. The large magnitude of the sliding velocity in event #4 was attributed to the change in the basal water system due to the switch from one minute water pulse durations to two minute pulse durations. This change in event duration may have allowed links and cavities to close more than they had in the previous set of perturbations. When repressurization occurred, water was initially confined to the existing cavities before the links could enlarge.

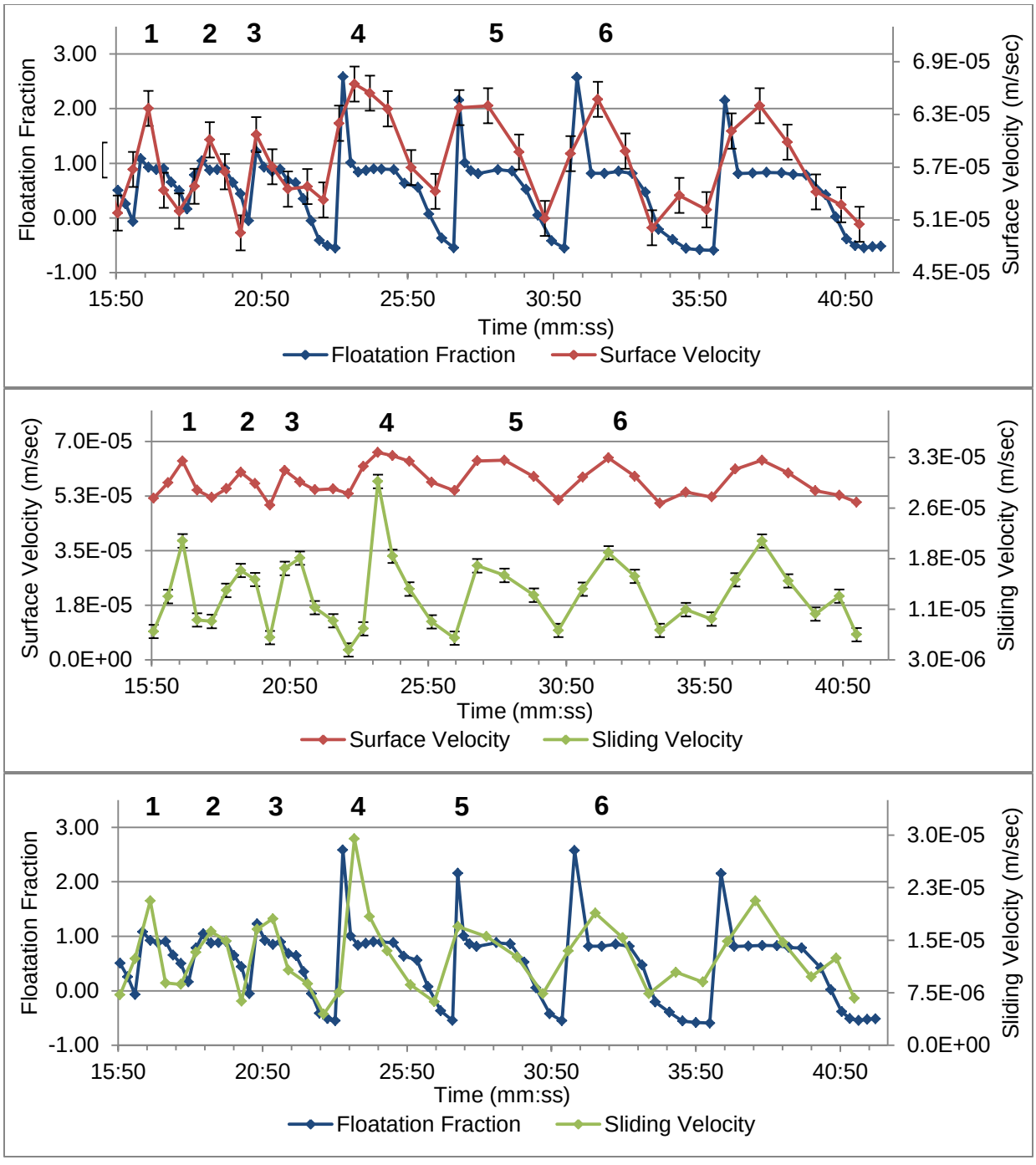

Figure 5: (a) Basal Water Pressure & Surface Velocity vs. Time, Transient Basal Water Pressures Period. (b) Velocity vs. Time, Transient Basal Water Pressures Period. (c) Basal Water Pressure & Basal Sliding Velocity vs. Time, Transient Basal Water Pressures Period. Pulsed pressurized water events are labeled 1 – 6 at the top of each plot.

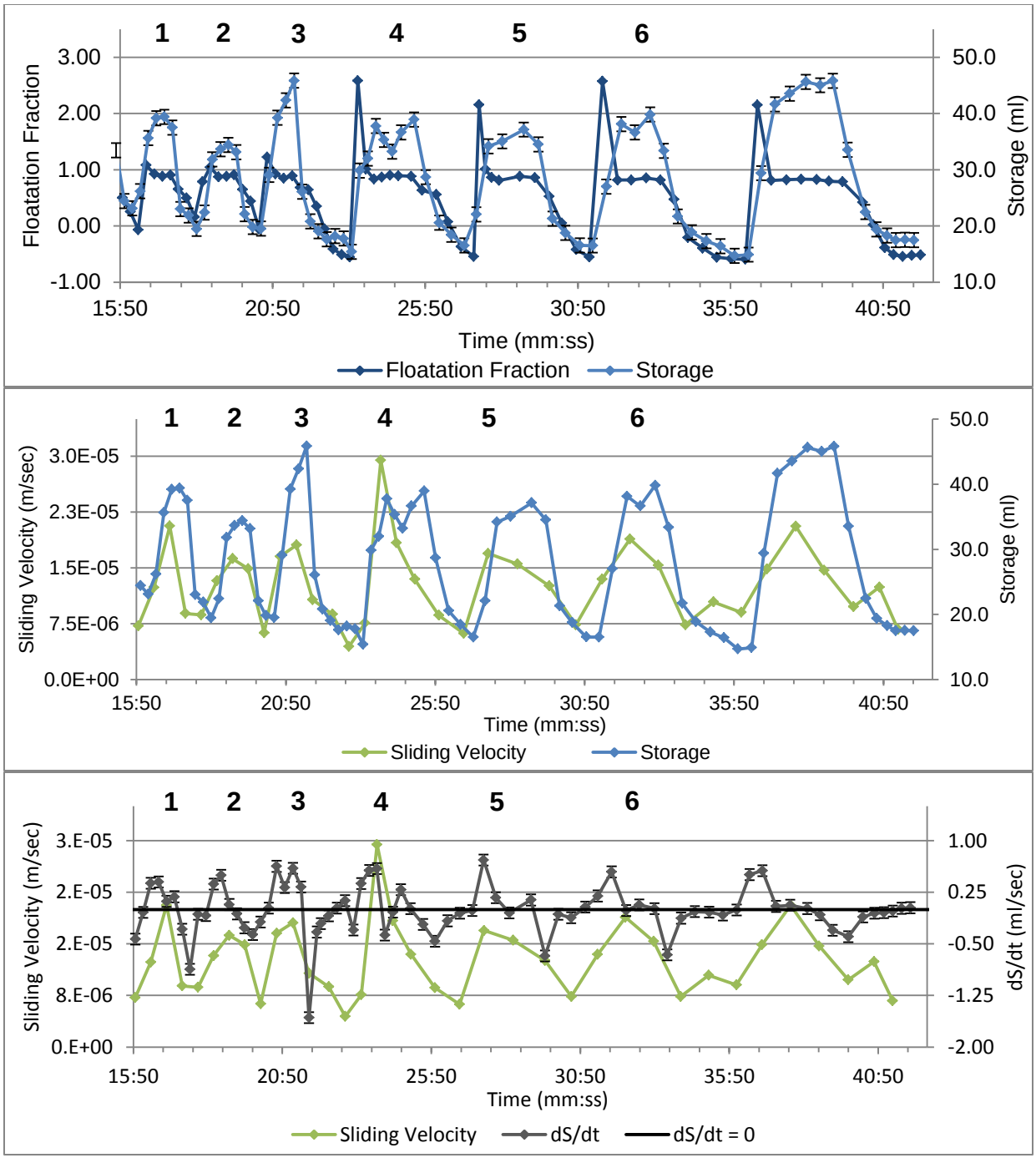

Figure 6: (a) Basal Water Pressure & Storage vs. Time, Transient Basal Water Pressures Period. (b) Basal Sliding Velocity & Basal Water Storage vs. Time, Transient Basal Water Pressures Period. (c) Basal Sliding Velocity & dS/dt vs. Time, Transient Basal Water Pressures Period. Pulsed pressurized water events are labeled 1 – 6 at the top of each plot.

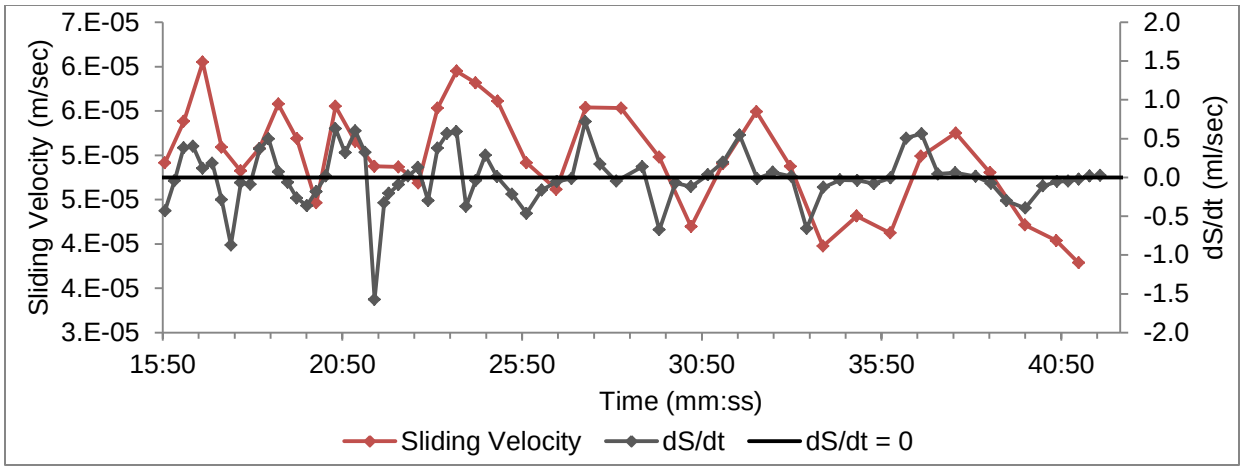

Figure 7: Surface Velocity and dS/dt vs. Time

5.2 Discussion

The clear peak in basal water pressure for each “diurnal event” points an under capacity linked cavity system. However, the unclear relationship between peak water pressure and peak sliding velocity does not support the water pressure based hydraulic jacking type mechanism described by (Iken, 1981). For this hydraulic jacking mechanism, Iken [1981] asserts that high water pressures enhance basal sliding by exerting a downstream force on the downstream walls of basal water cavities, accelerating ice. A decrease in coulomb-type friction was also ruled out as a mechanism for enhanced ice flow. Unlike the prototype water film, the model’s lubricant film cannot transmit pressures, so there was no decrease in friction associated with a reduced effective pressure. The much clearer correspondence between variations in dS/dt and those in sliding velocity confirm field observations (Bartholomaus et al., 2007; 2011) that dS/dt is directly linked to transient enhancements in ice flow during spring events.

To explain the relationship between dS/dt and sliding velocity, we propose dS/dt based hydraulic jacking rather than the water pressure based hydraulic jacking of Iken (1981) and Iken and Bindshadler (1986). To predict the metamorphism that the analog cavity system undergoes over the series of “spring events”, cavity height is plotted as a function of time and water pressure in Figure 8. These plots follow the work of (Nye, 1953) to calculate cavity height, using times and water pressures taken directly from the experimental data. We note that for our use the relation is approximate because it was derived for a circular cylinder without sliding; however, it provides an approximation of cavity geometry evolution due to observed basal water pressure. These plots confirm that significant cavity closure should occur during

each low pressure period; followed by a reopening during the next high water pressure period. This is confirmed by the similarity of the magnitudes of peaks in storage shown in Figure 6(b). Based on the above information, we propose that non-zero dS/dt affected basal sliding velocity by changing cavity geometry. Cavity growth in the downstream direction produced a downglacier normal force on the PDMS cavity roof which accelerated the sliding velocity. This mechanism also explains the delay, observed both in our work and the field (Barthalomaus et al., 2007; 2011), between the peak in dS/dt and the peak in sliding velocity, as the downglacier force accelerates the basal ice over a period of time. We suggest that the spike in water pressure at the beginning of each pressurized event is water trying to force its way through the under-capacity linked-cavity system after the period of cavity closure predicted in Figure 8.

The observed transient enhancements in analog ice flow show a combination of both rigid body and fluid flow dynamics. At the bed, enhanced sliding was controlled by cavity growth that acted as a downstream rigid body force that accelerated basal material. At the surface, a baseline surface velocity existed due to internal deformation and some background separation-based basal sliding. This baseline surface velocity was accelerated by enhanced sliding at the bed, whose peaks were smoothed during the transfer of force through the viscous substance from the bed to the surface of the model glacier.

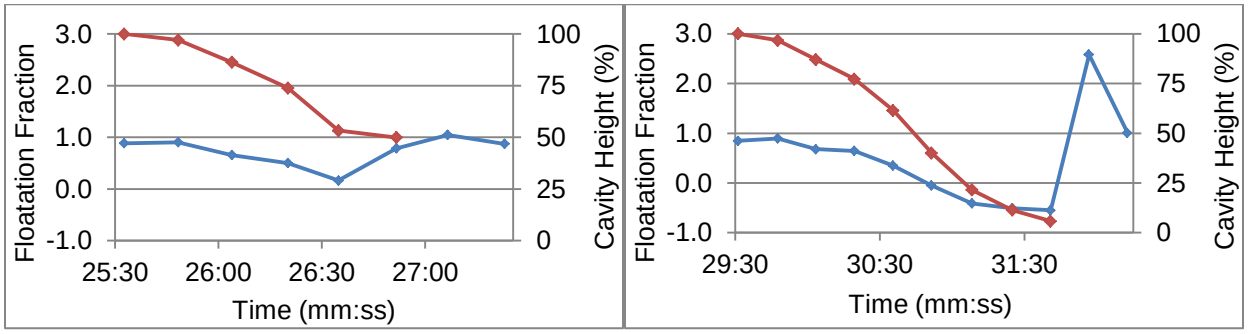

Figure 8: Cavity height and water pressure vs. time: (a) Short low water pressure period. (b) Long low water pressure period. Cavity height is represented as % of initial height and is plotted in red; floatation fraction is shown in blue.

6 Error Quantification

Error quantification for each parameter of interest followed the same methodology. Experimental time periods were selected where the parameter of interest was known to be constant. A sample of time steps was then run through the corresponding processing algorithm and the standard deviation of the parameter was determined.

PDMS flow displacement error was determined by tracking a zero displacement bead grid over a sample time period. Displacements were then calculated using the particle-tracking algorithm, and the standard deviation from zero displacement was determined. The reported error represents an upper bound as the error was intentionally maximized by jarring the channel and changing the lighting during the error quantification process. The reported error in displacement was equal to an image distance of only 0.17 pixels. Image timestamp error was determined by placing a stopwatch with a known precision in the camera's field of view to confirm the timestamp. Velocity measurement error accounted for all related sources of error: bead displacement, time, and surface velocity slowdown associated with a non-zero mass balance. The non-zero mass balance existed because no PDMS was added upstream over the entire experiment duration, because upstream PDMS addition can produce significant spikes in surface velocity that would have overshadowed the effects of the basal water. Sliding velocities have a smaller error than surface velocities because they experience the same relative slowdown from the non-zero mass balance flow slowdown.

Error in water pressure was quantified by setting the water pressures in all eight manometer tubes at many known levels. Representative water pressures were then calculated using the pressure calculation algorithm. The deviation of reported pressures from known pressure was then determined and the standard deviation was assessed.

Water discharge measurement error was found by setting the water supply pump at a constant and known flow rate. An accurate flow rate was determined volumetrically via a large volume long duration sample. Flow rates were then calculated multiple times using the experimental gravimetric method and the standard deviation from the true flow rate was determined.

Error in the measurement of basal water extent and volume was calculated by running the basal water extent algorithms over many time steps during which the basal water system remained at steady

state. The standard deviation from the mean reported volume was then determined. Error in the reported bed topography extent, a proxy for where basal water cannot exist, was quantified by determining the actual basal coverage for each bump and the corresponding standard deviation for the entire channel. Experimental error for all parameters is reported in Table 4.

Table 4: Experimental Error

Parameter	σ
Bead displacement (m)	4.2E-5
Time (sec)	0.1
Surface Velocity (m/sec)	2E-6
Sliding Velocity (m/sec)	1E-6
Floatation Fraction (P_w/P_o)	0.05
Discharge (ml/sec)	0.08
Basal Water Coverage (%)	0.5
Basal Water Volume (ml)	1.3
dS/dt (ml/sec)	.08
Bed topography extent (%)	4

7 Discussion and Conclusion

We developed an experimental setup and method that employed an ice analog substance that was subjected to basal water pulses. Our goal was to reproduce flow enhancements observed on temperate glaciers and the Greenland Ice Sheet. We developed methods for quantifying basal water extent and volume, representative basal water pressure, representative sliding velocity, and representative surface velocities. Our analog physical model complements other methods used in the study of ice dynamics because it allows: i) much better control and observation of significant variables than field work, at a fraction of the cost, and ii) the simulation of more complex bed topographies, water distributions, and their evolution than numerical modeling can achieve.

We found that widespread basal sliding was necessary for a sustainable water-filled linked cavity system to exist. At the same time, enhanced widespread basal sliding could not exist without this linked cavity system. Enhanced sliding and linked cavity systems are mutualistic, coupled processes that cannot exist without the other. The water filled linked cavity system separated the PDMS from the bed, enhancing ice flow by reducing basal drag. With a constant water pressure less than floatation, ice flow was enhanced by nearly 50% by bed separation. The development of an efficient conduit system reduced the extent of water at the bed, thus slowing enhanced ice flow. The modeled surface velocity enhancements were less than observed in the field; we attributed this to the viscous lubricant film which could not transmit basal water pressure and provided some resistance to basal sliding.

We observed that transient water pressures affected ice flow by changing water input to the bed. Increase in the rate of water storage at the bed (positive dS/dt) was observed to be highly correlated with enhanced ice speeds when water pressures are increasing. Abrupt decreases in water pressure and in dS/dt produced slowdowns corresponding to the speedups associated with water input to the bed. Similar to our suggestion that cavity growth enhances sliding by exerting a downstream normal force, we propose that negative dS/dt arrests sliding by allowing ice to flow back into collapsing cavities.

Based on the results of our experimental suite, we propose the following conceptual model of the observed evolution of ice flow over the course of the melt season. Several mechanisms combine to produce a net effect on ice flow. Over the course of a series of spring events, bed separation increases as water input to the bed increases. This produces a baseline enhanced ice flow by reducing basal drag.

On top of this, the diurnal melt cycle produces periods of strongly oscillating dS/dt, which produce corresponding accelerations and decelerations in sliding velocity and ice flow on a diurnal time scale. Over the course of the melt season, the subglacial hydrologic system evolves from one dominated by an inefficient linked cavity system to one dominated by an efficient conduit system. Drainage of the cavity system by these efficient conduits is followed by cavity closure and increased ice contact with the bed, ending the period of enhanced flow.

Although ice melt cannot be modeled with PDMS, PDMS analog experiments hold promise as a method to study and model ice flow phenomena, including: i) tracking the evolution of the basal water system through an entire annual cycle; ii) investigating the basal conditions that control surging glacier dynamics; iii) reproducing the quick retreat of ungrounded marine terminating glaciers; iv) and controlled experiments against which computational flow models can be checked.

References

Anderson, R.S., S. Anderson, K. MacGregor, E. Waddington, S. O'Neel, C. Riihimaki, M. Loso. "Strong feedbacks between hydrology and sliding of a small alpine glacier." *Journal of Geophysical Research*, 2004: 109, F03005.

Balise, Michael J., Charles F. Raymond. "Transfer of basal sliding variations to the surface of a linearly viscous glacier." *Journal of Glaciology*, 1985: 308-318.

Bartholomaus, Timothy C., Robert S. Anderson, Suzanne P. Anderson. "Response of glacier basal motion to transient water storage." *Nature*, 2007: 33-37.

Boutelier, D., C. Schrank, A. Cruden. "Power-law viscous materials for analogue experiments: New data on the rheology of highly-filled silicone polymers." *Journal of Structural Geology*, 2007: 341-353.

Catania, G., J. Buttles, D. Mohrig. "Analogue modeling of water flow under ice; what can we learn?" *16th Joint WAIS/FRISP Workshop.* Seattle WA, 2009.

Corti, Giacomo, Antonio Zeoli, Pietro Belmaggio, and Luigi Folco. "Physical modeling of the influence of bedrock topography and ablation on ice flow and meteorite concentration in Antarctica." *Journal of Geophysical Research*, 2008: 113, F01018.

Crocker, J. C. *track.pro.* University of Pennsylvania, 1999.

Crocker, J.C., D.G. Grier. *bpass.pro.* University of Chicago, 1997.

Detwiler, Russell L., Scott E. Pringle, Robert J. Glass. "Measurement of fracture aperture fields using transmitted light: An evaluation of measurement errors and their influence on simulations of flow and transport through a single fracture." *Water Resources Research*, 1999: 2605-2617.

Dufresne, Eric R. *cntrd.m.* Yale University, February 4, 2005.

Dufresne, Eric R. *pkfnd.m.* Yale University, February 4, 2005.

Harper, J.T., N.F. Humphrey, W.T. Pfeffer, B. Lazzar. "Two modes of accelerated glacier sliding related to water." *Geophysical Research Letters*, 2007: 34, L12503.

Hooke, Roger LeB. *Principles of Glacier Mechanics 2nd Edition.* Cambridge: Cambridge University Press, 2005.

Iken, A., R. A. Bindschadler. "Combined measurements of subglacial water pressure and surface velocity of Findelengletscher, Switzerland: Conclusions about drainage system andsliding mechanism." *Journal of Glaciology*, 1986: 101–119.

Iken, Almut. "The effect of the subglacial water pressure on the sliding velocity of a glacier in an idealized numerical model." *Journal of Glaciology*, 1981: 407-421.

Kamb, B. "Glacial surge mechanism based on the linked-cavity configuration of the basal water conduit system." *Journal of Geophysical Research*, 1987: 9083-9100.

Kamb, B., K. Echelmeyer. "Stress-gradient coupling in glacier flow: I. Longitudinal averaging of the influence of ice thickness and surface slope." *Journal of Glaciology*, 1986: 267-284.

Langlois, W. E. *Slow Viscous Flow.* NYC: Macmillan, 1964.

Nye, J.F. "The flow law of ice from measurements in glacier tunnels, laboratory experiments and the Jungfraufirn borehole experiment." *Proceedings of the Royal Society of London*, 1953: 477-489.

Paterson, W.S.B. *The Physics of Glaciers 3rd Edition.* Oxford: Elsevier Science, 1994.

Rignot, E., P. Kanagaratnam. "Changes in the velocity structure of the Greenland ice sheet." *Science*, 2006: 986-990.

Rossiter, B.W., and R.C. Baetzold. *Physical Methods of Chemistry: Determination of Electronical and Optical Properties.* New York: John Wiley, 1993.

Solomon, S., D. Qin, M. Manning, Z. Chen, M. Marquis, K.B. Averyt, M.Tignor and H.L. Mills. *IPCC, 2007: Summary for Policymakers. In: Climate Change 2007: The Physical Science Basis. Contribution of Working Group I to the Fourth Assessment Report of the Intergovernmental Panel on Climate Change.* Cambridge: Cambridge University Press, 2007.

Van Den Broeke, M., J. Bamber, J. Ettema, E. Rignot, E. Schrama. "Partitioning Recent Greenland Mass Loss." *Science*, 2009: 326, 984–986.

Van Der Veen, C.J. *Fundamentals of Glacier Dynamics.* Rotterdam: A.A.Balkema, 1999.

Weijermars, Ruud. "Flow behaviour and physical chemistry of bouncing putties and related polymers in view of tectonic laboratory applications." *Tectonophysics*, 1986: 325-358.

Zwally, J., W. Abdalati, T. Herring, K. Larson, J. Saba, K. Steffen. "Surface melt induced acceleration of Greenland ice-sheet flow." *Science*, 2002: 218-222.

Appendix 1 Experimental Setup

A.1.1 Ice Analog

The dynamic viscosity of the PDMS was calculated using the falling ball viscometer test. In this test a steel ball of known geometry and weight was allowed to fall through an acrylic cylinder filled with PDMS. The fall distance and time were recorded and used to calculate the settling velocity and then the dynamic viscosity of the PDMS. The viscometer, PDMS, and falling ball are shown below in Figure 9.

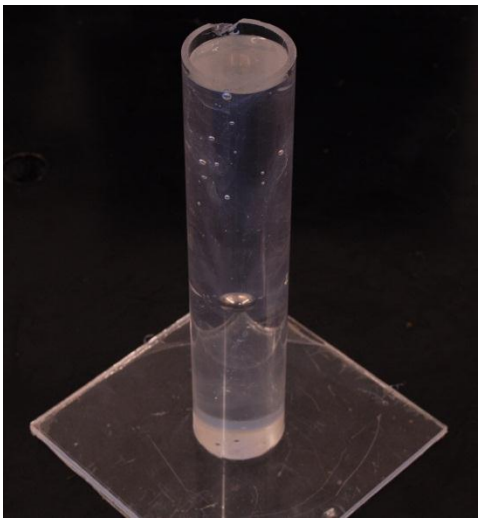

Figure 9: Falling ball viscometer test.

The viscometer cylinder was 22 cm tall with an internal diameter of 4.4 cm. By measuring the diameter (1.27 cm) of the ball and recording its mass, the density of the steel ball was calculated as 7794 kg/m^3, which was confirmed as falling within the widely reported range of steel densities. To avoid end effects and to allow the steel ball to reach terminal velocity, the middle 7 cm of the cylinder was used for the viscosity calculation. The fall distance and travel time of the ball was recorded via time-lapse photography by a camera pointed perpendicular to side of the cylinder. The ball fell 7.22 cm in 6596 seconds, equal to a terminal velocity of 1.1E-5 meters per second. PDMS dynamic viscosity was calculated using equation (1) below.

(1) $$\mu = \frac{2r^2 g(\rho_s - \rho_f)}{9V_s}$$

Where μ is dynamic viscosity, g is acceleration of gravity, r is sphere radius, ρ_s is density of the sphere, ρ_f is PDMS density, and V_s is settling velocity. This formula, derived by Stokes, represents the balance of the force of gravity and the drag force exerted on spherical objects within very low Reynolds number viscous fluids. Faxen's correction, used to account for the sidewall drag effects, was multiplied by the calculated viscosity to find the true viscosity. Faxen's correction is given by equation (2).

(2) $$\boldsymbol{F = 1 - 2.104\ \frac{d_s}{d_v} + 2.09\left(\frac{d_s}{d_v}\right)^3 - .95\left(\frac{d_s}{d_v}\right)^5}$$

Where d_s was the diameter of the sphere and d_v was the diameter of the viscometer cylinder. The final viscosity, corrected for sidewall effects, was calculated to be 24400 Pa sec, the same value as that reported by Boutelier, 2007. Because previous works have reported that PDMS has a Newtonian rheology under the range of strain rates seen in our work (Weijermars, 1986; Boutelier, 2007), we did not investigate its rheology further.

PDMS density as reported by Boutelier, 2007 and Weijermars, 1986 was confirmed by measuring the volume and mass of a PDMS sample. The volume was calculated by displaced water volume. I tested three samples finding an mean density of 975 kg/m^3. However, I assumed a value of 965 kg /m^3 based the slightly lower values reported by Boutelier and Weijermars.

A.1.1 Channel and Channel Frame

As reported in Chapter 2, the flow channel was built out of acrylic sheet with dimensions of 24 inches long, 18 inches wide, and 6 inches deep. The flow channel and its supporting frame are shown in Figure 10 below.

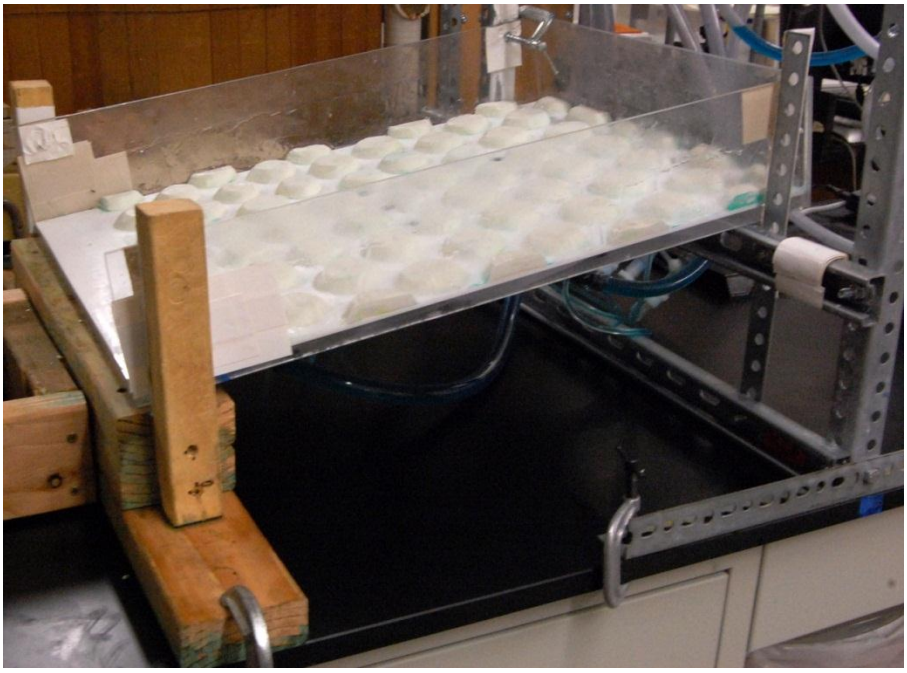

Figure 10: Flow channel & channel frame.

The flow channel walls were made of 1/8 inch thick acrylic sheet, while the channel bottom was made out of ¾ inch thick white acrylic sheet. Clear acrylic sheet was used for the flow channel walls to allow along channel visualization of PDMS flow and basal cavity evolution. The flow channel bed was built out of thick acrylic sheet to prevent channel bed distortion from the weight of the overlying PDMS. A thin layer of white acrylic sheet was placed on top of the thick acrylic bed to allow clear imaging of the basal water and tracking particles. Because of the high price of colored acrylic sheet it was significantly cheaper to build the bed as a composite of a thin acrylic sheet placed on top of a thick clear acrylic sheet. Data collection was performed within a study area located over the channel center. As discussed in Chapter 2, this area was sized based on the work of Kamb (1986) and Langlois (1964) to be essentially free of sidewall and end effects. Figure 11, a plot of the theoretical surface velocity profile for a fluid flowing down a rectangular duct, shows that there should be a region in the center of the flow channel that is essentially free of sidewall effects.

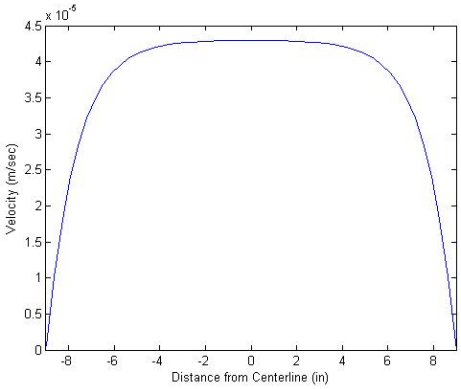

Figure 11: Calculated theoretical surface velocity for PDMS flowing down the experimental channel with a no slip condition at the bed.

The channel frame, designed to support and stabilize the flow channel and the outflow channel, was built out of steel bar and wood. The front and back of the flow channel rested on the support bars of the channel frame. The flow channel was prevented from moving horizontally by vertical bars extending upward from the channel frame at all four corners. By adjusting the height of the upstream support bar the channel bed slope could be adjusted. At each corner the channel was clamped securely to the frame; which was then clamped at four points to the lab bench. The outflow channel was then secured to the main channel and rested on the frame. Vibrations transmitted through the lab bench did produce small movements in the flow channel and channel frame. These vibrations could be further minimized by using a stiffer lab bench, and performing experiments during periods of low activity in the engineering center.

As discussed in Chapter 2, a bed topography made of plasticine bumps was stuck to the acrylic bed. Plasticine was used for the bed topography because it can be easily molded and smoothed. Its smoothabilty was a great asset as it allowed easy removal of the small scale cracks that tended to form in the bed mold materials I tested. These cracks were a significant hindrance to our experiments as the PDMS tended to stick in them. The two checkerboard topographies were designed to have similar head loss at equivalent flow rates, and similar sliding resistance, but to cover differ portions of the bed. Head loss was modeled assuming that the vast majority of head loss in the basal water system occurred when water exited each link into the immediately downstream cavity. So, this head loss was modeled like an abrupt expansion in pipe flow, assuming that all of the velocity head was lost as the water flowed out of each contraction. The total head loss for the channel was then calculated as the sum of the individual

head losses through all the contractions between bed topography bumps. Basal sliding resistance was described as the total upstream vertical cross sectional area of the bed topography.

A.1.2 Water Supply System

The water supply system was designed to supply pressurized dyed water through a distributed system to the channel bed. Two dyes were used for the basal water in the quantitative experimental suite. CB1 Developed Conduit and CB1 Sliding used a blue and green water coloring, while CB2 Sliding and CB2 Patchy sliding used FD&C Blue. I switched to FD&C Blue because it is a stronger dye with better visualization characteristics than food coloring. Water was pumped by a 500 ml/sec capacity aquarium pump from a large supply reservoir into the water distribution system. The supply reservoir was an 18 inch acrylic cube with a capacity of 60000 ml; this volume was large enough to continuously supply water to the bed of the experimental channel for 100 minutes without refilling the supply reservoir. Immediately after leaving the pump, a pipe splitter distributed water to four pipes which each delivered water to a different constant head reservoir (CHR). A constant head reservoir and its components are diagrammed in Figure 12.

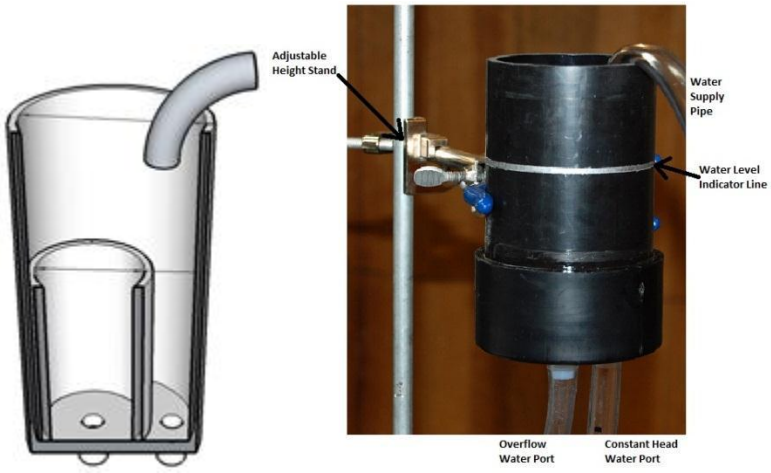

Figure 12: Constant head reservoir section and side view.

Each CHR was built as one large pipe with a smaller pipe embedded within it, both of which were capped at the bottom. These two pipes formed two small water reservoirs. The inner pipe was shorter than the outer pipe, so that when the outer pipe volume filled with water it could overflow into the inner pipe. The

outer pipe acted as the constant head reservoir, while the inner pipe acted as the overflow reservoir. At the bottom of each pipe was a water outlet port. Each outlet port was connected to the basal water supply system. Excess water that overflowed from the outer pipe to the inner pipe flowed out the drain port back to the supply reservoir. By supplying water to each CHR at a high flow rate, the outer pipe was always spilling into the inner pipe, maintaining a constant elevation head for the channel bed. Each CHR was placed on its own individual stand, allowing their height to be adjusted separately. Each stand was then clamped securely to the lab bench. Flow out of each constant head reservoir to the channel bed was controlled individually by adjusting the CHR elevation and by an outflow valve.

From each CHR water was directed to a rotameter. The rotameters were installed with the hope of measuring the inflow rate of water to the basal water system. The four rotameters were mounted together on a small panel which was mounted to the upstream edge of the channel frame. The rotameters were monitored by the side view camera. Unfortunately, flow rates reported by the rotameters were not accurate enough to be used. The low accuracy was due to the variability in camera angle relative to the rotameters as the flow rate through them changed and the difficulty resolving the flow rate marker in the experimental imagery. After passing through the rotameters, the water was directed to the basal water ports which were embedded in the channel bed. As discussed in Chapter 2, each CHR provided water to three locations on one row of basal water ports through a basal water supply splitter (shown in Figure 13a).

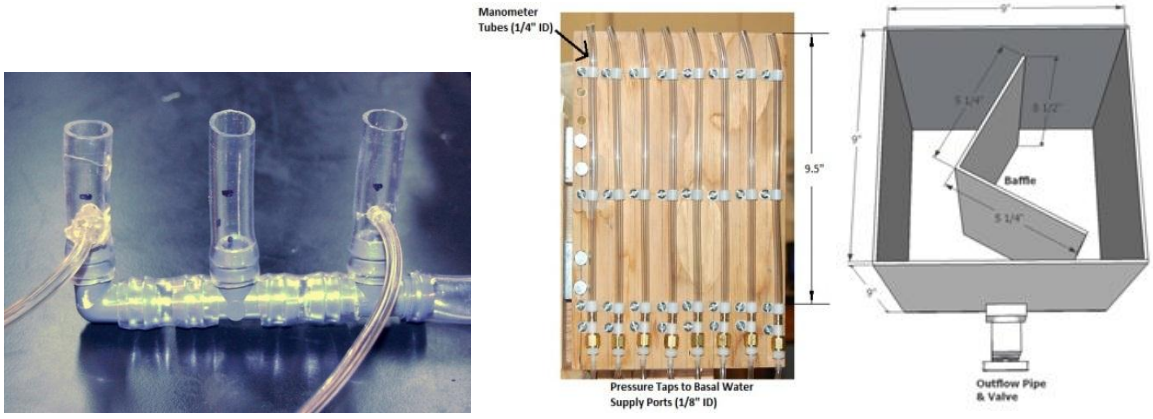

Figure 13: (a) Basal water supply splitter and pressure taps. (b) Manometer board (c) Outflow tank.

Pressure taps connected to the basal water supply splitters were used to monitor the basal water pressure. As can be seen in Figure 14, two pressure taps were connected to each row of basal water ports. The pressure taps are the two small tubes connecting to the sides of the larger pipes. Rather than locating the pressure taps in the channel bed, the pressures taps were located just below it because of the inherent stickiness of the PDMS. If the pressure taps had been located in the channel bed, then the PDMS would have quickly flowed into the taps, clogged them, and prevented any pressure measurement. The location of the pressure taps represented the best compromise I could find of some level of pressure measurement without clogging of the taps. The pressure taps were connected the manometer board, shown in Figure 13b, via low diameter connector pipes which were pressed onto the fittings at the bottom of each manometer. Both the pressure taps and the manometer pipes themselves were designed to be thin enough that an insignificant volume of water was needed to fill them.

Once water entered the basal water ports it flowed through the basal water system and exited under the model terminus. From the terminus, water was routed by the outflow channel into an acrylic outflow tank, shown in Figure 13c. The outflow tank was placed on a balance, and with a 10000 ml capacity could store at least 16 minutes of model discharge. A baffle was built inside the outflow tank to reduce movement and sloshing of water and hence improve balance accuracy. The outflow tank could be drained via an overflow valve, located at its base, which directed water into an overflow tank. The overflow tank had a capacity of 2000 ml and could be dumped back into the storage reservoir when it was full.

A.1.3 Lubricant System

A lubricant investigation was performed to find a lubricant which would separate the PDMS from the bed, and allow the PDMS to slide over the bed. A lubricant was necessary to accomplish these requirements for two reasons. First, the PDMS was very sticky. Without a lubricant the PDMS stuck to the bed so strongly that pressurized basal water could not pry the PDMS off the bed, instead the water pushed bubbles up into the PDMS. Even when a basal water layer was established at the bed without a lubricant, as soon as the water layer drained the PDMS re-stuck to the bed. Second, a lubricating layer

was necessary as a water film analog to break the no slip condition at the bed and allow ice analog basal sliding.

Finding a suitable lubricant proved very difficult. First, a thin, non-water based lubricant was tested. Not only did the PDMS displace the lubricant, it absorbed it. Second, I tried a viscous non-water based lubricant. The lubricant was not displaced by the PDMS, but it was absorbed by the PDMS. Although the absorption of the lubricant did not occur over the timescale of an experimental run, it occurred over the time scale necessary to allow the PDMS to settle in the channel prior to each experiment. Next, glycerin was tried as the basal lubricant. The PDMS did not absorb the glycerin lubricant, but when basal water was introduced, the water quickly washed the water soluble glycerin away from the bed and the PDMS adhered to the bed.

Finally, I tested a combined lubricant system using both glycerin and high viscosity grease. The grease was first brushed on the channel bed and then the glycerin was brushed on top of it. Both lubricant layers were made as thin as possible without leaving any areas of the bed without lubricant. The glycerin prevented the PDMS from absorbing the grease over the long pre-experiment PDMS settling period. When water was introduced at the bed the glycerin was eventually washed out, but the grease continued to provide lubrication over the duration of the experiment run. Recognizing that glycerin acted as a non-wetting fluid on the acrylic channel bed, hand soap was added to the glycerin. The hand soap acted as a surfactant, making the glycerin wet more on the acrylic. The lubricant system was not perfect: the PDMS still occasionally stuck to the bed and the grease was highly viscous, providing more resistance to sliding than the prototype water film. But it worked: it separated the PDMS from the bed, and allowed basal sliding. Further improvement of the lubricant system would improve experimental modeling and results. The ideal lubricant would be low viscosity and would adhere to the bed; providing little resistance to basal sliding and not washing away from flowing basal water.

A.1.4 Data Collection System

As discussed briefly in Chapter 2, the data collection system was made up of two cameras and a digital balance. One camera was mounted above the channel, one to the side of it, and the balance was

located just below the outflow channel of the model glacier. The top view camera was mounted on a ceiling frame, designed and built by the author, which is shown in Figure 14.

Figure 14: Top view camera, ceiling frame & auxiliary lighting.

The ceiling frame was mounted to the ceiling for increased stability, to remove it from the work area, and to allow it to also support a supplementary lighting system. The frame was built from slotted steel tubing from which the top view camera and the fluorescent light strips hung. The camera mount was adjustable, allowing it to be rotated and translated. The top view camera was a Nikon D5000 DSLR with a Nikkor DX 18-55mm f/3.5-5.6G VR lens. This camera was selected because of its high imaging quality, remote fire capabilities, and relatively low cost. To allow longer run times the camera battery was replaced with the Nikon EP-5 Power Supply Connector, allowing the camera to be plugged into a wall outlet. The camera was remotely fired by the side view camera through radio remotes mounted on the flash shoes of both cameras. The remote fire mechanisms, Digital Radio Slaves by SM-Development Co., were essentially affordable versions of Pocket Wizards. When the side view camera fired it triggered the radio remote mounted on top of it. The radio remote on the top view camera then picked up this signal and triggered the top view camera via a USB cable attached to the D5000's remote shutter release port.

The side view camera was a Nikon D40 DSLR with a NIKKOR AF-S DX Zoom 18-55mm f/3.5-5.6G ED II lens. The D40 was used as the side view camera because it was already available in the Environmental Fluid Mechanics Lab. It was fired remotely by the lab computer via a USB cable using a very nice freeware script called DIYPhotoBits.com Camera Control 5.2. DIYPhotoBits can be downloaded from http://www.diyphotobits.com/. DIYPhotobits allows the user to operate the camera remotely from a computer; it allows control of all the functions that are normally adjusted on the camera, has a time-lapse function, and immediately uploads imagery to the computer. I used DIYPhotobits to run time lapses on the cameras and to immediately upload new imagery to the lab computer. The ability to instantly upload and view experimental imagery as it was taken was a great assent as it allowed quick experimental troubleshooting and analysis. It should be noted that the time-lapse method used produced a significant amount of variability in the actual time between shutter releases, and as each experiment progressed the period between shutter releases increased. This was inconvenient, but was accounted for by logging the timestamp for every experimental image of interest. This was extremely inconvenient, and must be dealt with if any more experiments in this vein are performed.

As mentioned in Chapter 2, continuous outflow was measured gravimetrically by an Acculab balance placed under the outflow tank. The balance sent mass measurements to the lab computer five times every second. The timestamped mass data from the balance was received on the lab computer by WinCT, a hyperterminal developed by A&D Company, Limited. WinCT can be downloaded from http://www.aandd.jp/products/software/winctdown.html.

A.1.5 Image Processing

MATLAB codes were used to process the top and side view experimental imagery. I used a combination of freeware scripts, and codes that I developed. Images were processed for tracking particles, basal water, and basal water pressure. Basal water was quantified using wedge.m and watercalc.m, both developed by the author. All codes used can be seen in Appendix 2.

To be able to quantify basal water extent, volume, and thickness I developed a relationship between the light emitted from the water underneath the PDMS layer and the thickness of the water at that point. To do this I developed Wedge.m, which used a dyed water filled wedge built onto the channel

bed to find a relationship between water thickness and emitted light intensity. To ensure similarity between the experiment and the wedge test, I placed a two inch thick layer of PDMS on top of the water filled wedge, used the same lighting scheme, the same dyed water, and the same channel bed slope. The emitted light intensity was reported by MATLAB as a grayscale value. The wedge was built out of acrylic sheet, allowing high quality imaging. To eliminate diffraction of light passing through the fluid layers, the water surface and the PDMS surface were oriented normal to the supplementary lighting system and the top view camera. The wedge thickness went from zero inches at the upstream end to .5 inches at the downstream end; this ensured that all experimentally possible water thicknesses were quantified. The developed relationship followed the concept of the Beer-Lampert Law: that there is a logarithmic relationship between light emitted from a fluid, and the distance the light first travels through the fluid. This relationship is described by the absorption coefficient of the substance. Clearly, our situation is different in that the light source is not below the water, but rather above it, so light must enter the water and then reflect back towards the direction it came from. However, the plot of water layer thickness vs. emitted light intensity shows that there is a strong and clear relationship between thickness and intensity.

Wedge.m used the following methodology: First, under the exact same lighting and camera settings as the experimental suite, a top view photo was taken of the water wedge. The image was then loaded into MATLAB and a grayscale version of the image was produced. Then, at each pixel distance downstream the wedge was sampled at 20 unique points for emitted light intensity. The mean of these 20 samples was calculated to find the representative light intensity for each thickness. Once light intensity was calculated at each thickness, an exponential trend line was fit to light intensity as a function of water layer thickness. The resulting relationships between transmitted light intensity and dyed water layer thickness for FD&C Blue and food coloring had r^2 values of .998 and .997 respectively, plots for these relationships are shown in Figure 15 below. Clearly FD&C Blue provides more accuracy in the quantification of basal water than food coloring because it exhibits a wider range of light transmission values than food coloring.

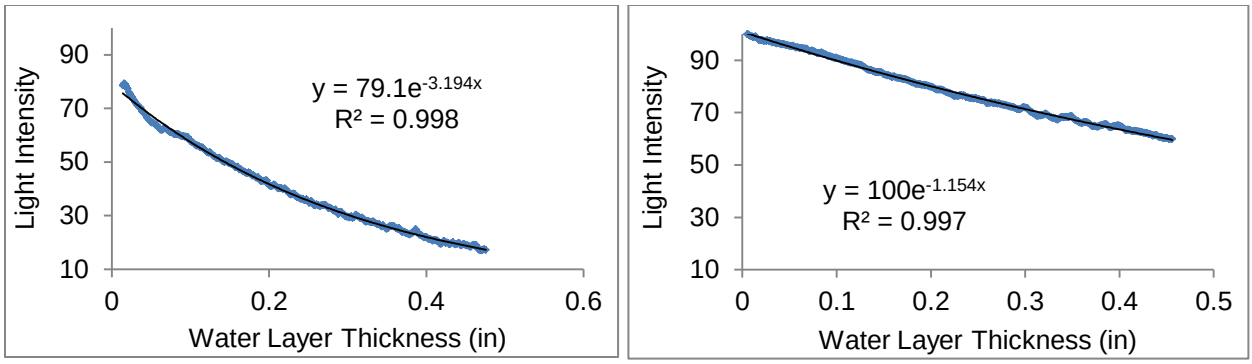

Figure 15: Transmitted light intensity vs. dyed water layer thickness: (a) FD&C Blue. (b) Food coloring. Transmitted light intensity is the unitless ratio of incident light to transmitted light.

Once the relationship between emitted light intensity and water thickness was developed, I was able to calculate the extent, thickness, and volume of water at the bed at all times for all experiments. My script, watercalc.m, quantified these basal water properties. Watercalc.m first loaded all the experiment images into MATLAB, and then processed each image individually using the following methodology: First, a grayscale version of each image was produced. Second, all tracking particles were removed from the image by filtering for their color. Tracking particle colors were identified by their RGB image values. Third, all locations of basal water were identified, unrealistically bright spots were removed, and the light intensity for each location of basal water was logged. The existence of dyed basal water was identified based on a range of color values and light intensities that I specified. Unrealistically bright spots, identified by grayscale light intensity values higher than a zero thickness water layer, were removed because they were diagnosed as reflections of the supplementary lighting system off of the PDMS surface. Basal water extent was then calculated as the portion of the total bed area where basal water was identified as existing. Water layer thickness was calculated at each pixel based on the derived Beer-Lampert relationship. A sample of how original top view imagery was processed for basal water is show in Figure 16. Black denotes no basal water; lighter shades denote water where the thickest water layer is represented by bright white.

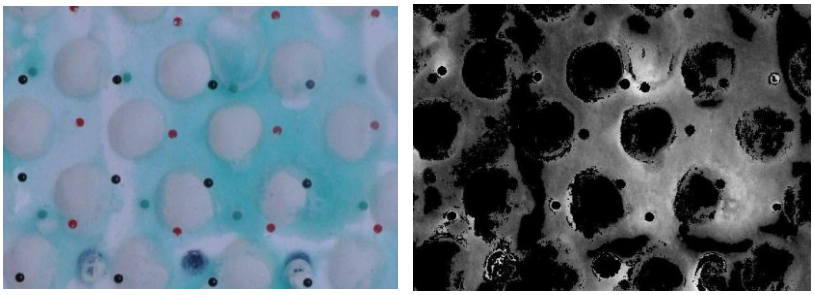

Figure 16: Basal water quantification process: (a) Original image. (b) Quantification of basal water extent and thickness.

Total basal water volume was found by integrating the water layer thickness at each pixel over the entire study area. dS/dt was calculated as the change in stored basal water volume over the duration of each time step. Although images were taken roughly every 5 seconds, I found that averaging dS/dt over every two time steps significantly reduced noise associated with imperfect processing algorithms and changing experimental lighting conditions.

Seven millimeter diameter colored polystyrene beads were used as tracking particles. Following the tracking particles over time and calculating velocity profiles was accomplished using a combination of several freeware scripts and a script that I developed. The freeware scripts I used were developed by Eric Dufresne and John Crocker to track round bacteria colonies over time; they identified the tracking particles and followed them through time. My script, velpfrl.m, was used to call the individual particle tracking scripts, filter for possible beads based on their color, and then calculate velocity profiles. The steps taken by Velprfl.m are listed next. First, each top view image was loaded into MATLAB, and it was color filtered to identify possible bead locations. The color filter range was set to select one level of particles at a time: either surface or bed particles. Unique colors of beads were used at each level to facilitate this process. Second, bpass.m was called to remove pixel level noise and possible particles that were not the characteristic length of the tracking particles from the color filtered image. Bpass.m is a two-step bandpass process that removes non important data by first convolving the original image with a Gaussian, then convolving the original image with a boxcar. By subtracting the boxcar version from the Gaussian version, the bandpassed image is produced.

Third, pkfnd.m was called to identify each individual particle and provide a rough guess of the location of each particle center. Particles were identified by finding local grayscale maxima in each image,

based on user defined particle diameter and minimum brightness. Fourth, cntrd.m was called to calculate the location of the centroid of each particle. The progression from a raw image to one where each particle and its center were identified can be seen in Figure 17.

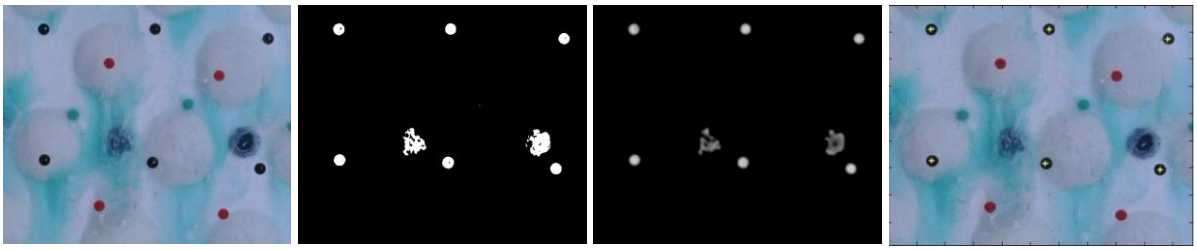

Figure 17: Image processing progression: a) Original image. b) Image color filtered for tracking particles. c) Non-important image data surpressed. d) Bead centers located.

Fifth, track.m was called to determine trajectories from a scrambled list of particle coordinates at each time step. Track.m followed particles over each time step by selecting particle tracks which produced the minimum total squared displacement for all particles. With the particle tracks and their associated displacements over time calculated, velocities were then calculated for each particle at each time step. Representative bed and surface velocities were calculated using the eight particles closest to the study area center. Of these selected eight particles, four were immediately downstream of the study area center, while the other four were immediately upstream of the center. Particles were paired in groups of two; each group consisted of one particle downstream of and one particle upstream of the channel center. A linearly interpolated velocity was calculated for each particle pair as a function of the distance along channel from the channel midpoint. The mean of these four interpolated velocities was then computed to find a representative channel velocity.

Basal water pressures at each time step were calculated using presspick.m, a user assisted MATLAB script that I developed. Using the side view camera imagery of the manometer board the script followed the steps listed below to identify representative pressures at each time step. For the first image of each experiment presspick.m first required a series of user designated image reference points: i) The four corners of the manometer board were selected to determine optical distortion; ii) Then the center of each manometer pipe was then selected; iii) Finally, a known length was selected to determine the image pixel to true length discretization. Next, each image was loaded into a viewer window and the water

height was manually selected for all eight manometers. This was done using a command which allowed rapid identification of the y coordinate of each pipe via a mouse click, and automated entering of this data into an array. After all the images had been displayed, the water pressure was calculated at each tap accounting for image distortion and varying channel elevation change along the length of the channel. The data was then loaded into excel. Pressure taps that were clearly not connected to the basal water supply system were completely removed from the analysis and a mean water pressure was calculated for the entire channel.

Ideally a color filtering script, similar in some ways to the color filtering algorithm for the tracking beads, would have been developed which would have automated the process of identifying the height of water in each manometer. In concept this would not be very hard, but given the time restraints of my project, manually identifying the water heights with the click of a mouse was the simpler solution. I would use the following methodology to develop a fully automated manometer water height identification algorithm: First, acceptable ranges for the horizontal location of each manometer pipe would be identified and input. Second, the locations of water in each pipe would be identified based on a range of RGB values corresponding to the color of dyed water in the clear manometer tubes. Third, the water level surface in each manometer tube would be identified as the maximum vertical position where a set number of pixels of water are identified.

Appendix 2 Image Processing Codes

A.2.1 Light Transmission Quantification

```
% Name: wedge.m
% Author: Mike Records
% Purpose: Following the principle of the Beer-Lampert law, a water filled wedge is used to develop a
%       relationship between grayscale light intensity emitted from the dyed water and the thickness of
%       the water.
% Inputs: Geometry of water filled wedge, incipient light intensity, wedge image
% Outputs: Emitted light intensity as a function of water thickness.
% Procedure: The Beer-Lampert law describes the relationship between light entering a fluid, the light
%       absorbed, and the light emitted by the fluid. An image of a water filled wedge is loaded into
%       MATLAB. the dye concentration and the lighting of the wedge are the same as those used for the
%       experimental suite. At each pixel level wedge thickness 20 samples of emitted light intensity are
%       taken and an average light intensity is reported. Emitted light intensity vs. water layer thickness
%       over the entire wedge is then reported which can be fit by an exponential function where one
%       constant defines the properties of the dye.
clear; clc
%Read image
im = imread('RDnC.jpg');
imtool(im);
%Convert to grayscale pixel intensity
tstgry = rgb2gray(im);
% imtool(tstgry);
%Set ranges of study area
xrange = [60, 80];
yrange = [50, 350];
%Preallocate
sum1 = zeros((yrange(2)-yrange(1)+1),(xrange(2)-xrange(1)+1));
```

```
avg = zeros((yrange(2)-yrange(1)+1),1);
%Loop to cover entire range of water thicknesses
for y = yrange(1):yrange(2)
    y1 = y - yrange(1) + 1;
    %Loop to sample multiple points at each thickness
    for x = xrange(1):xrange(2)
        x1 = x - xrange(1) + 1;
        sum1(y1, x1) = tstgry(y,x);
    end
    %calculate average intensity at each thickness
    avg(y1) = sum(sum1(y1,:))/(xrange(2)-xrange(1)+1);
    y
end
%Plot Results
yplot = yrange(1):yrange(2);
plot(yplot, avg)
xlabel('Thickness (sort of)'); ylabel('Light Intensity');
```

A.2.2 Basal Water Quantification

```
% Name: watercalc.m
% Author: Mike Records
% Purpose: This code takes a series of top view channel images where basal water may exist and
%       determines whether or not water exists at each pixel, what percentage of the bed is covered by
%       water, how thick the water layer is at each pixel, and the total volume of water under the analog
%       glacier. Inputs: Image range, image stepsize, bead color/intensity range, basal water
%       color/intensity range.
% Outputs: Fractional coverage of basal water, total basal water volume
% Procedure: Given a series of images, the images are loaded into MATLAB and a grayscale version of
%       each image is created. Color and grayscale images are then filtered in tandem to determine
%       where particles exist. Particles are removed as possible locations of water. The filtered image is
%       then processed again for the existance or non existance of water based on a user input for the
%       range of color and light intensities that represent water. From this the percent coverage of water
%       is determined. The determined locations of water existence are then filtered for thickness by light
%       intensity from the predetermined relationship between light intensity and thickness. Pixel level
%       water thicknesses are then integrated to determine the total volume of basal water.
%       1) Grayscale intensity calculated for each image
%       2) Color filter run to remove basal particles
%       3) Color filter run to only look at areas with basal water
%       4) Unrealistically bright/intense data removed
%       5) Background intensity removed
%       6) Summation of grayscale intensities for entire image where basal water was identified
pic = strcat('DSC_0', int2str(l), '.jpg');
eval('im=imread(pic);')
imtool(im)
clear; clc;
li = 832; lo = 842; l = li;
```

```
pic = strcat('DSC_0', int2str(l), '.jpg');
eval('im=imread(pic);')
sz=size(im);
% imtool(im)
%Preallocate
t = []; tot = []; extent = zeros(2,1);
% plothold = zeros(sz(1), sz(2), lo);
for l = li:1:lo
    t = [t;l];
    index = (l - li + 1)
    pic = strcat('DSC_0', int2str(l), '.jpg');
    eval('im=imread(pic);')
    tstgry = rgb2gray(im);
%     imtool(im)
%     imtool(tstgry)
    %Color filter particles
    sz=size(im);
    red = zeros(sz(1), sz(2));
    for i=1:sz(1)
        for j=1:sz(2)
            if ((im(i,j,1)<40 & im(i,j,2)<40 & im(i,j,3)<50) || (im(i,j,1)<75 && im(i,j,2)<75 && im(i,j,3)<110 &&
tstgry(i,j)<95)) %black (5.9.2011)
                tstgry(i,j)=152.34;
                else if ((im(i,j,1)>80 && im(i,j,2)<50 && im(i,j,3)<80) || (tstgry(i,j)<85 && im(i,j,1)>85 &&
im(i,j,2)<70 && im(i,j,3)<95)) %red (5.9.2011)
                    tstgry(i,j)=152.34;
                end
            end
```

```
        end
    end
%     imtool(tstgry)
    %Filter for colored basal water and remove unrealistically bright spots
    %(i.e. reflections or particles the color filter missed)
    extent = zeros(lo);
    total = zeros(sz(1), sz(2));
    for i=1:sz(1)
        for j=1:sz(2)
%             if ((im(i,j,1)<im(i,j,2) && im(i,j,2)<im(i,j,3)+25 && im(i,j,1)<130 && im(i,j,2)>65 && tstgry(i,j)<155)
|| (im(i,j,1)<im(i,j,2) && im(i,j,2)<im(i,j,3) && im(i,j,1)<90 && tstgry(i,j)<125))
            if ((im(i,j,1)<im(i,j,2) && im(i,j,2)<im(i,j,3) && im(i,j,1)<135 && im(i,j,2)>145 && tstgry(i,j)<152.34)
|| (im(i,j,1)<im(i,j,2) && im(i,j,2)<im(i,j,3) && im(i,j,1)<90 && tstgry(i,j)<152.34))
                extent(l) = extent(l) + 1;
                total(i,j) = tstgry(i,j);
            else
                total(i,j) = 152.34;
            end
        end
    end
%     imtool(total)
%     imtool(total/255)
    %Save each processed image for later viewing
%     plothold(:,:,index) = total(:,:);
    clear vctrz;
    vctrz = -log(total/152.35)/1.154;
    tot(index) = sum(sum(vctrz));
end
```

```
% Plotting commands just to check that everything is working
% imtool(plothold(:,:,1))
plot(tot)
%Calculate Fractional Percentage
exttot = size(im,1)*size(im,2);
extper = extent/exttot*100;
%Transpose qualitative total water storage vectoy vector
tot = tot';
%Change water storage vector to correct units
delx = 108.6;
totml = tot/delx/delx*16.39%(ml)
imnumplt = (li:1:lo)';
plot(imnumplt, totml)
%Plot qualitative total water vs time
figure(1)
plot( t, tot, '.')
title('Total Basal Water Index vs. Time')
xlabel('Time (Image #)')
ylabel('Total Basal Water Index')
%Plot qualitative total water vs time
figure(2)
plot( t, extper, '.')
title('Percent Basal Water Extent vs. Time')
xlabel('Time (Image #)')
ylabel('Percent Basal Water Extent')
figure(1)
imshow(plot1/max(max(total)))
title('Sample Image #1')
```

```
figure(2)
imshow(plot2/max(max(total)))
title('Sample Image #2')
figure(3)
imshow(plot3/max(max(total)))
title('Sample Image #3')
pic = strcat(int2str(l), 'edit.jpg');
eval('im=imread(pic);')
imtool(im)
im = imread('discretization1.jpg')
imtool(im)
```

A.2.3 Tracking Particle Noise Suppression

```
function res = bpass(image_array,lnoise,lobject,threshold)
% NAME: bpass.m
% PURPOSE: Implements a real-space bandpass filter that suppresses pixel noise and long-wavelength
% image variations while retaining information of a characteristic size.
% CATEGORY: Image Processing
% CALLING SEQUENCE: res = bpass( image_array, lnoise, lobject )
% INPUTS:
%       image:  The two-dimensional array to be filtered.
%       lnoise: Characteristic lengthscale of noise in pixels. Additive noise averaged over this length
%           should vanish. May assume any positive floating value. May be set to 0 or false, in which
%           case only the highpass "background subtraction" operation is performed.
%       lobject: (optional) Integer length in pixels somewhat larger than a typical object. Can also be
%           set to 0 or false, in which case only the lowpass "blurring" operation defined by lnoise is
%           done, without the background subtraction defined by lobject.  Defaults to false.
%       threshold: (optional) By default, after the convolution, any negative pixels are reset to 0.
%           Threshold changes the threshhold for setting pixels to 0.  Positive values may be useful
%           for removing stray noise or small particles.  Alternatively, can be set to -Inf so that no
%           threshholding is performed at all.
% OUTPUTS:
%       res:    filtered image.
% PROCEDURE:
%       simple convolution yields spatial bandpass filtering.
% NOTES:
% Performs a bandpass by convolving with an appropriate kernel.  You can think of this as a two part
% process.  First, a lowpassed image is produced by convolving the original with a gaussian.  Next, a
% second lowpassed image is produced by convolving the original with a boxcar function. By subtracting
% the boxcar version from the gaussian version, we are using the boxcar version to perform a highpass.
```

```
% original - lowpassed version of original => highpassed version of the original. Performing a lowpass
% and a highpass results in a bandpassed image. Converts input to double.  Be advised that commands
% like 'image' display double precision arrays differently from UINT8 arrays.
% MODIFICATION HISTORY:
% Written by David G. Grier, The University of Chicago, 2/93.
% Greatly revised version DGG 5/95.
% Added /field keyword JCC 12/95.
% Memory optimizations and fixed normalization, DGG 8/99.
% Converted to Matlab by D.Blair 4/2004-ish
% Fixed some bugs with conv2 to make sure the edges are removed D.B. 6/05
% Removed inadvertent image shift ERD 6/05
% Added threshold to output.  Now sets all pixels with negative values equal to zero. Gets rid of
%   ringing which was destroying sub-pixel accuracy, unless window size in cntrd was picked perfectly.
%   Now centrd gets sub-pixel accuracy much more robustly ERD 8/24/05
% Refactored for clarity and converted all convolutions to use column vector kernels for speed. Running
%   on my macbook, the old version took ~1.3 seconds to do bpass(image_array,1,19) on a 1024 x 1024
%   image; this version takes roughly half that. JWM 6/07
% This code 'bpass.pro' is copyright 1997, John C. Crocker and David G. Grier.  It should be
%   considered 'freeware'- and may be distributed freely in its original form when properly attributed.
if nargin < 3, lobject = false; end
if nargin < 4, threshold = 0; end
normalize = @(x) x/sum(x);
image_array = double(image_array);
if lnoise == 0
  gaussian_kernel = 1;
else
  gaussian_kernel = normalize(...
    exp(-((-ceil(5*lnoise):ceil(5*lnoise))/(2*lnoise)).^2));
```

```
end
if lobject
  boxcar_kernel = normalize(...
     ones(1,length(-round(lobject):round(lobject))));
end
% JWM: Do a 2D convolution with the kernels in two steps each.  It is
% possible to do the convolution in only one step per kernel with
  % gconv = conv2(gaussian_kernel',gaussian_kernel,image_array,'same');
  % bconv = conv2(boxcar_kernel', boxcar_kernel,image_array,'same');
% but for some reason, this is slow.  The whole operation could be reduced
% to a single step using the associative and distributive properties of convolution:
  % filtered = conv2(image_array,...
  %   gaussian_kernel'*gaussian_kernel - boxcar_kernel'*boxcar_kernel,...
  %   'same');
% But this is also comparatively slow (though inexplicably faster than the above).  It turns out that
% convolving with a column vector is faster than convolving with a row vector, so instead of transposing
% the kernel, the image is transposed twice.
gconv = conv2(image_array',gaussian_kernel','same');
gconv = conv2(gconv',gaussian_kernel','same');
if lobject
  bconv = conv2(image_array',boxcar_kernel','same');
  bconv = conv2(bconv',boxcar_kernel','same');
  filtered = gconv - bconv;
else
  filtered = gconv;
end
% Zero out the values on the edges to signal that they're not useful.
lzero = max(lobject,ceil(5*lnoise));
```

```
filtered(1:(round(lzero)),:) = 0;

filtered((end - lzero + 1):end,:) = 0;

filtered(:,1:(round(lzero))) = 0;

filtered(:,(end - lzero + 1):end) = 0;

% JWM: I question the value of zeroing out negative pixels.  It's a nonlinear operation which could

% potentially mess up our expectations about statistics.  Is there data on 'Now centroid gets subpixel

% accuracy much more robustly'?  To choose which approach to take, uncomment one of

% the following two lines.

% ERD: The negative values shift the peak if the center of the cntrd mask

% is not centered on the particle.

% res = filtered;

filtered(filtered < threshold) = 0;

res = filtered;
```

A.2.4 Rough Tracking Particle Locator

```
function out=pkfnd(im,th,sz)
%  Name: pkfnd.m
%  Finds local maxima in an image to pixel level accuracy. This provides a rough guess of particle
%  centers to be used by cntrd.m.  Inspired by the lmx subroutine of Grier and Crocker's feature.pro
% INPUTS:
%         im: image to process, particle should be bright spots on dark background with little noise. Often
%           an bandpass filtered brightfield image (fbps.m, fflt.m or bpass.m) or a nice fluorescent image
%         th: the minimum brightness of a pixel that might be local maxima.
%           (NOTE: Make it big and the code runs faster but you might miss some particles.  Make it small
%           and you'll get everything and it'll be slow.)
%         sz:  if your data's noisy, (e.g. a single particle has multiple local maxima), then set this optional
%           keyword to a value slightly larger than the diameter of your blob. If multiple peaks are found
%           within a radius of sz/2 then the code will keep only the brightest.  Also gets rid of all peaks within
%           sz of boundary.
% OUTPUT:  a N x 2 array containing, [row,column] coordinates of local maxima
%          out(:,1) are the x-coordinates of the maxima
%          out(:,2) are the y-coordinates of the maxima
% CREATED: Eric R. Dufresne, Yale University, Feb 4 2005
% MODIFIED: ERD, 5/2005, got rid of ind2rc.m to reduce overhead on tip by
% Dan Blair;  added sz keyword
% ERD, 6/2005: modified to work with one and zero peaks, removed automatic normalization of image
% ERD, 6/2005: due to popular demand, altered output to give x and y
%  instead of row and column
% ERD, 8/24/2005: pkfnd now exits politely if there's nothing above threshold instead of crashing rudely
% ERD, 6/14/2006: now exits politely if no maxima found
% ERD, 10/5/2006:  fixed bug that threw away particles with maxima consisting of more than two
%  adjacent points
```

```
%find all the pixels above threshold
%im=im./max(max(im));
ind=find(im > th);
[nr,nc]=size(im);
tst=zeros(nr,nc);
n=length(ind);
if n==0
   out=[];
   display('nothing above threshold');
   return;
end
mx=[];
%convert index from find to row and column
rc=[mod(ind,nr),floor(ind/nr)+1];
for i=1:n
   r=rc(i,1);c=rc(i,2);
   %check each pixel above threshold to see if it's brighter than it's neighbors
   %  THERE'S GOT TO BE A FASTER WAY OF DOING THIS.  I'M CHECKING SOME MULTIPLE
TIMES,
   %  BUT THIS DOESN'T SEEM THAT SLOW COMPARED TO THE OTHER ROUTINES, ANYWAY.
   if r>1 & r<nr & c>1 & c<nc
      if im(r,c)>=im(r-1,c-1) & im(r,c)>=im(r,c-1) & im(r,c)>=im(r+1,c-1) & ...
       im(r,c)>=im(r-1,c)  & im(r,c)>=im(r+1,c) &   ...
       im(r,c)>=im(r-1,c+1) & im(r,c)>=im(r,c+1) & im(r,c)>=im(r+1,c+1)
      mx=[mx,[r,c]'];
      %tst(ind(i))=im(ind(i));
      end
   end
```

```
end
%out=tst;
mx=mx';
 [npks,crap]=size(mx);
%if size is specified, then get ride of pks within size of boundary
if nargin==3 & npks>0
   %throw out all pks within sz of boundary;
    ind=find(mx(:,1)>sz & mx(:,1)<(nr-sz) & mx(:,2)>sz & mx(:,2)<(nc-sz));
    mx=mx(ind,:);
end
 %prevent from finding peaks within size of each other
[npks,crap]=size(mx);
if npks > 1
    %CREATE AN IMAGE WITH ONLY PEAKS
    nmx=npks;
    tmp=0.*im;
    for i=1:nmx
        tmp(mx(i,1),mx(i,2))=im(mx(i,1),mx(i,2));
    end
    %LOOK IN NEIGHBORHOOD AROUND EACH PEAK, PICK THE BRIGHTEST
    for i=1:nmx
        roi=tmp( (mx(i,1)-floor(sz/2)):(mx(i,1)+(floor(sz/2)+1)),(mx(i,2)-floor(sz/2)):(mx(i,2)+(floor(sz/2)+1)))
        [mv,indi]=max(roi);
        [mv,indj]=max(mv);
        tmp( (mx(i,1)-floor(sz/2)):(mx(i,1)+(floor(sz/2)+1)),(mx(i,2)-floor(sz/2)):(mx(i,2)+(floor(sz/2)+1)))=0;
        tmp(mx(i,1)-floor(sz/2)+indi(indj)-1,mx(i,2)-floor(sz/2)+indj-1)=mv;
    end
    ind=find(tmp>0);
```

```
    mx=[mod(ind,nr),floor(ind/nr)+1];
end
 if size(mx)==[0,0]
    out=[];
else
    out(:,2)=mx(:,1);
    out(:,1)=mx(:,2);
end
```

A.2.5 Tracking Particle Centroid Determination

```
function out=cntrd(im,mx,sz,interactive)
% out=cntrd(im,mx,sz,interactive)
% NAME: cntrd.m
% PURPOSE:  calculates the centroid of bright spots to sub-pixel accuracy.
%         Inspired by Grier & Crocker's feature for IDL, but greatly simplified and optimized for matlab
% INPUT:
%         im: image to process, particle should be bright spots on dark background with little noise
%           often an bandpass filtered brightfield image or a nice fluorescent image
%         mx: locations of local maxima to pixel-level accuracy from pkfnd.m
%         sz: diamter of the window over which to average to calculate the centroid. Should be big enough
%           to capture the whole particle but not so big that it captures others. If initial guess of center (from
%           pkfnd) is far from the centroid, the window will need to be larger than the particle size.
%           RECCOMMENDED size % is the long lengthscale used in bpass plus 2. interactive:
% OPTIONAL INPUT: set this variable to one and it will show you the image used to calculate each
%           centroid, the pixel-level peak and the centroid
% NOTE:
%         - if pkfnd, and cntrd return more then one location per particle then you should try to filter your
%           input more carefully.  If you still get more than one peak for particle, use the optional sz
%           parameter in pkfnd
%         - If you want sub-pixel accuracy, you need to have a lot of pixels in your window (sz>>1). To
%           check for pixel bias, plot a histogram of the fractional parts of the resulting locations - It is
%           HIGHLY recommended to run in interactive mode to adjust the parameters before you analyze a
%           bunch of images.
% OUTPUT:  a N x 4 array containing, x, y and brightness for each feature
%          out(:,1) is the x-coordinates
%          out(:,2) is the y-coordinates
%          out(:,3) is the brightnesses
```

```
%           out(:,4) is the sqare of the radius of gyration
% CREATED: Eric R. Dufresne, Yale University, Feb 4 2005
% Modifications:
% 5/2005 inputs diameter instead of radius
% D.B. (6/05) Added code from imdist/dist to make this stand alone.
% ERD (6/05) Increased frame of reject locations around edge to 1.5*sz
% ERD 6/2005  By popular demand, 1. altered input to be formatted in x,y
%  space instead of row, column space  2. added forth column of output, rg^2
% ERD 8/05  Outputs had been shifted by [0.5,0.5] pixels.  No more!
% ERD 8/24/05  Woops!  That last one was a red herring.  The real problem is the "ringing" from the
%  output of bpass.  I fixed bpass (see note), and no longer need this kludge.  Made it quite nice if mx=[];
% ERD 6/06  Added size and brightness output ot interactive mode.  Also
%   fixed bug in calculation of rg^2 JWM 6/07  Small corrections to documentation
if nargin==3
  interactive=0;
end
if sz/2 == floor(sz/2)
warning('sz must be odd, like bpass');
end
if isempty(mx)
    warning('there were no positions inputted into cntrd. check your pkfnd theshold')
    out=[];
    return;
end
r=(sz+1)/2;
%create mask - window around trial location over which to calculate the centroid
m = 2*r;
x = 0:(m-1) ;
```

```
cent = (m-1)/2;
x2 = (x-cent).^2;
dst=zeros(m,m);
for i=1:m
   dst(i,:)=sqrt((i-1-cent)^2+x2);
end
ind=find(dst < r);
msk=zeros([2*r,2*r]);
msk(ind)=1.0;
%msk=circshift(msk,[-r,-r]);
dst2=msk.*(dst.^2);
ndst2=sum(sum(dst2));
[nr,nc]=size(im);
%remove all potential locations within distance sz from edges of image
ind=find(mx(:,2) > 1.5*sz & mx(:,2) < nr-1.5*sz);
mx=mx(ind,:);
ind=find(mx(:,1) > 1.5*sz & mx(:,1) < nc-1.5*sz);
mx=mx(ind,:);
[nmx,crap] = size(mx);
%inside of the window, assign an x and y coordinate for each pixel
xl=zeros(2*r,2*r);
for i=1:2*r
   xl(i,:)=(1:2*r);
end
yl=xl';
pts=[];
%loop through all of the candidate positions
for i=1:nmx
```

```
    %create a small working array around each candidate location, and apply the window function
    tmp=msk.*im((mx(i,2)-r+1:mx(i,2)+r),(mx(i,1)-r+1:mx(i,1)+r));
    %calculate the total brightness
    norm=sum(sum(tmp));
    %calculate the weigthed average x location
    xavg=sum(sum(tmp.*xl))./norm;
    %calculate the weighted average y location
    yavg=sum(sum(tmp.*yl))./norm;
    %calculate the radius of gyration^2
    %rg=(sum(sum(tmp.*dst2))/ndst2);
    rg=(sum(sum(tmp.*dst2))/norm);
    %concatenate it up
    pts=[pts,[mx(i,1)+xavg-r,mx(i,2)+yavg-r,norm,rg]'];
    %OPTIONAL plot things up if you're in interactive mode
    if interactive==1
     imagesc(tmp)
     axis image
     hold on;
     plot(xavg,yavg,'x')
     plot(xavg,yavg,'o')
     plot(r,r,'.')
     hold off
     title(['brightness ',num2str(norm),' size ',num2str(sqrt(rg))])
     pause
    end
end
out=pts'
```

A.2.6 Particle Tracker

```
function tracks = track(xyzs,maxdisp,param)
% NAME: track.m
% PURPOSE:
%       Constructs n-dimensional trajectories from a scrambled list of particle coordinates determined at
%       discrete times (e.g. in consecutive video frames).
%       see http://glinda.lrsm.upenn.edu/~weeks/idl for more information
% CATEGORY: Image Processing
% CALLING SEQUENCE:
%       result = track( positionlist, maxdisp, param ) set all keywords in the space below
% INPUTS:
%       positionlist: an array listing the scrambled coordinates and data of the different particles at
%         different times, such that positionlist(0:d-1,*): contains the d coordinates and data for all the
%         particles, at the different times. Must be positive positionlist(d,*): contains the time that the
%         position was determined, must be integers (e.g. frame number. These values must be
%         monotonically increasing and uniformly gridded in time.
%       maxdisp: an estimate of the maximum distance that a particle would move in a single time
%         interval.(see % Restrictions)
%  OPTIONAL INPUT:
%       param: a structure containing a few tracking parameters that are needed for many applications.  If
%         param is not included in the function call, and then default values are used.  If you set one value
%         make sure you set them all:
%       param.mem: this is the number of time steps that a particle can be 'lost' and then recovered
%         again. If the particle reappears after this number of frames has elapsed, it will be tracked as a
%         new particle. The default setting is zero. This is useful if particles occasionally 'drop out' of
%         the data.
%       param.dim: if the user would like to unscramble non-coordinate data
%         for the particles (e.g. apparent radius of gyration for the particle images), then positionlist should
```

```
%          contain the position data in positionlist(0:param.dim-1,*) and the extra data in
%          positionlist(param.dim:d-1,*). It is then necessary to set dim equal to the dimensionality of the
%          coordinate data to so that the track knows to ignore the non-coordinate data in the construction
%          of the trajectories. The default value is two.
%        param.good: set this keyword to eliminate all trajectories with
%          fewer than param.good valid positions.  This is useful for eliminating very short, mostly 'lost'
%          trajectories due to blinking 'noise' particles in the data stream.
%        param.quiet: set this keyword to 1 if you don't want any text
% OUTPUTS:
%         result:  a list containing the original data rows sorted into a series of trajectories.  To the original
%          input data structure there is appended an additional column  containing a unique 'id number' for
%          each identified particle trajectory.  The result array is sorted so rows with corresponding id
%          numbers are in contiguous blocks, with the time variable a monotonically increasing function
%          inside each block.  For example:
%     For the input data structure (positionlist):
%        (x)      (y)      (t)
%     pos = 3.60000      5.00000      0.00000
%          15.1000      22.6000      0.00000
%          4.10000      5.50000      1.00000
%          15.9000      20.7000      2.00000
%          6.20000      4.30000      2.00000
%     IDL> res = track(pos,5,mem=2)
%     track will return the result 'res'
%        (x)      (y)      (t)          (id)
%     res = 3.60000      5.00000      0.00000      0.00000
%          4.10000      5.50000      1.00000      0.00000
%          6.20000      4.30000      2.00000      0.00000
%          15.1000      22.6000      0.00000      1.00000
```

```
%         15.9000      20.7000      2.00000      1.00000
%    NB: for t=1 in the example above, one particle temporarily vanished.  As a result, the trajectory id=1
%    has one time missing, i.e. particle loss can cause time gaps to occur
%    in the corresponding trajectory list. In contrast: IDL> res = track(pos,5) track will return the result 'res'
%       (x)      (y)      (t)         (id)
%    res = 15.1000      22.6000      0.00000      0.00000
%                3.60000      5.00000      0.00000      1.00000
%              4.10000      5.50000      1.00000      1.00000
%              6.20000      4.30000      2.00000      1.00000
%              15.9000      20.7000      2.00000      2.00000
%    where the reappeared 'particle' will be labelled as new rather than as a continuation of an old particle
%    since mem=0. It is up to the user to decide what setting of 'mem' will yield the highest fidelity.
%       SIDE EFFECTS:
%         Produces informational messages. Can be memory intensive for extremely large data sets.
%       RESTRICTIONS:
%         maxdisp should be set to a value somewhat less than the mean spacing between the particles.
%         As maxdisp approaches the mean spacing the runtime will increase significantly. The function
%         will produce an error message: "Excessive Combinatorics!" if the run time would be too long,
%         and the user should respond by re-executing the function with a smaller value of maxdisp.
%         Obviously, if the particles being tracked are frequently moving as much as their mean separation
%         in a single time step, this function will not return acceptable trajectories.
%       PROCEDURE:
%         Given the positions for n particles at time t(i), and m possible new positions at time t(i+1), this
%         function considers all possible identifications of the n old positions with the m new positions,
%         and chooses that identification which results in the minimal total squared displacement. Those
%         identifications which don't associate a new position within maxdisp of an old position (particle
%         loss) penalize the total squared displacement by maxdisp^2. For non-interacting Brownian
%         particles with the same diffusivity, this algorithm will produce the most probable set of
```

```
%       identifications (provided maxdisp >> RMS displacement between frames ). In practice it works
%       reasonably well for systems with oscillatory, ballistic, correlated and random hopping motion, so
%       long as single time step displacements are reasonably small. NB: multidimensional functionality
%       is intended to facilitate tracking when additional information regarding target identity is available
%       (e.g. size or color). At present, this information should be rescaled by the user to have a
%       comparable or smaller (measurement) variance than the spatial displacements.
% MODIFICATION HISTORY:
%       2/93 Written by John C. Crocker, University of Chicago (JFI).
%       7/93 JCC fixed bug causing particle loss and improved performance for large numbers of (>100)
%       particles.
%       11/93 JCC improved speed and memory performance for large numbers of (>1000) particles
%       (added subnetwork code).
%       3/94 JCC optimized run time for trivial bonds and d<7. (Added d-dimensional raster metric code.)
%       8/94 JCC added functionality to unscramble non-position data along with position data.
%       9/94 JCC rewrote subnetwork code and wrote new, more efficient permutation code.
%       5/95 JCC debugged subnetwork and excessive combinatorics code.
%       12/95 JCC added memory keyword, and enabled the tracking of newly appeared particles.
%       3/96 JCC made inipos a keyword, and disabled the adding of 'new' particles when inipos was set.
%       3/97 JCC added 'add' keyword, since Chicago users didn't like having particle addition be the
%       default.
%       9/97 JCC added 'goodenough' keyword to improve memory efficiency when using the 'add'
%       keyword and to filter out bad tracks.
%       10/97 JCC streamlined data structure to speed runtime for >200 timesteps.  Changed 'quiet'
%       keyword to 'verbose'. Made time labelling more flexible (uniform and sorted is ok).
%       9/98 JCC switched trajectory data structure to a 'list' form, resolving memory issue for large, noisy
%       datasets.
%       2/99 JCC added Eric Weeks's 'uberize' code to post-facto rationalize the particle id numbers,
%       removed 'add' keyword.
```

```
%       1/05 Transmuted to MATLAB by D. Blair
%       5/05 ERD Added the param structure to simplify calling.
%       6/05 ERD Added quiet to param structure
%       7/05 DLB Fixed slight bug in trivial bond code
%       3/07 DLB Fixed bug with max disp pointed out by Helene Delanoe-Ayari
% This code 'track.pro' is copyright 1999, by John C. Crocker.
% It should be considered 'freeware'- and may be distributed freely (outside of the military-industrial
%   complex) in its original form when properly attributed.
dd = length(xyzs(1,:));
%use default parameters if none given
if nargin==2
   %default values
   memory_b=0; % if mem is not needed set to zero
   goodenough = 0;  % if goodenough is not wanted set to zero
   dim = dd - 1;
   quiet=0;
else
   memory_b    =   param.mem;
   goodenough  =   param.good;
   dim         =   param.dim;
   quiet       =   param.quiet;
end
% checking the input time vector
t = xyzs(:,dd);
st = circshift(t,1);
st = t(2:end) - st(2:end);
if  sum(st(find(st < 0))) ~= 0
   disp('The time vectors is not in order')
```

```
    return
end
info = 1;
 w = find(st > 0);
z = length(w);
z = z +1;
if isempty(w)
    disp('All positions are at the same time... go back!')
    return
end
 % partitioning the data with unique times
 %res = unq(t);
% implanting unq directly
    indices = find(t ~= circshift(t,-1));
        count = length(indices);
        if count > 0
            res = indices;
        else
            res = length(t) -1;
        end
 %%%%%%%%%%%%%%%%%%%%%%%%
res = [1,res',length(t)];
ngood = res(2) - res(1) + 1;
eyes = 1:ngood;
pos = xyzs(eyes,1:dim);
istart = 2;
n = ngood;
zspan = 50;
```

```
if n > 200
    zspan = 20;
end
if n > 500
    zspan = 10;
end
resx = zeros(zspan,n) - 1;
bigresx = zeros(z,n) - 1;
mem = zeros(n,1);
%  whos resx
%  whos bigresx
uniqid = 1:n;
maxid = n;
olist = [0.,0.];
if goodenough > 0
    dumphash = zeros(n,1);
    nvalid = ones(n,1);
end
%  whos eyes;
resx(1,:) = eyes;
% setting up constants
maxdisq = maxdisp^2;

% John calls this the setup for "fancy code" ???
notnsqrd = (sqrt(n*ngood) > 200) & (dim < 7);
notnsqrd = notnsqrd(1);
 if notnsqrd
    %;   construct the vertices of a 3x3x3... d-dimensional hypercube
```

```
    cube = zeros(3^dim,dim);
    for d=0:dim-1,
        numb = 0;
        for j=0:(3^d):(3^dim)-1,
            cube(j+1:j+(3^(d)),d+1) = numb;
            numb = mod(numb+1,3);
        end
    end
    %   calculate a blocksize which may be greater than maxdisp, but which
    %   keeps nblocks reasonably small.
    volume = 1;
    for d = 0:dim-1
        minn = min(xyzs(w,d+1));
        maxx = max(xyzs(w,d+1));
        volume = volume * (maxx-minn);
    end
    volume;
    blocksize = max( [maxdisp,((volume)/(20*ngood))^(1.0/dim)] );
end
%   Start the main loop over the frames.
for i=istart:z
    ispan = mod(i-1,zspan)+1;
    %disp(ispan)
    % get new particle positions
    m = res(i+1) - res(i);
    res(i);
    eyes = 1:m;
    eyes = eyes + res(i);
```

```
if m > 0
    xyi = xyzs(eyes,1:dim);
    found = zeros(m,1);
    % THE TRIVIAL BOND CODE BEGINS
    if notnsqrd
        %Use the raster metric code to do trivial bonds
        % construct "s", a one dimensional parameterization of the space
        % which consists of the d-dimensional raster scan of the volume.)
        abi = fix(xyi./blocksize);
        abpos = fix(pos./blocksize);
        si = zeros(m,1);
        spos = zeros(n,1);
        dimm = zeros(dim,1);
        coff = 1.;
        for j=1:dim
            minn = min([abi(:,j);abpos(:,j)]);
            maxx = max([abi(:,j);abpos(:,j)]);
            abi(:,j) = abi(:,j) - minn;
            abpos(:,j) = abpos(:,j) - minn;
            dimm(j) = maxx-minn + 1;
            si = si + abi(:,j).*coff;
            spos = spos + abpos(:,j).*coff;
            coff = dimm(j).*coff;
        end
        nblocks = coff;
        % trim down (intersect) the hypercube if its too big to fit in the
        % particle volume. (i.e. if dimm(j) lt 3)
        cub = cube;
```

```
deg = find( dimm < 3);
if ~isempty(deg)
   for j = 0:length(deg)-1
      cub = cub(find(cub(:,deg(j+1)) < dimm(deg(j+1))),:);
   end
end
% calculate the "s" coordinates of hypercube (with a corner @ the origin)
scube = zeros(length(cub(:,1)),1);
coff = 1;
for j=1:dim
   scube = scube + cub(:,j).*coff;
   coff = coff*dimm(j);
end
% shift the hypercube "s" coordinates to be centered around the origin
coff = 1;
for j=1:dim
   if dimm(j) > 3
      scube = scube - coff;
   end
   coff = dimm(j).* coff;
end
scube = mod((scube + nblocks),nblocks);
% get the sorting for the particles by their "s" positions.
[ed,isort] = sort(si);
% make a hash table which will allow us to know which new particles
% are at a given si.
strt = zeros(nblocks,1) -1;
fnsh = zeros(nblocks,1);
```

```
h = find(si == 0);
lh = length(h);
if lh > 0
si(h) = 1;
end
for j=1:m
    if strt(si(isort(j))) == -1
        strt(si(isort(j))) = j;
        fnsh(si(isort(j))) = j;
    else
        fnsh(si(isort(j))) = j;
    end
end
if lh > 0
si(h) = 0;
end
coltot = zeros(m,1);
rowtot = zeros(n,1);
which1 = zeros(n,1);
for j=1:n
    map = fix(-1);
    scub_spos = scube + spos(j);
    s = mod(scub_spos,nblocks);
    whzero = find(s == 0 );
    if ~isempty(whzero)
        nfk = find(s ~=0);
        s = s(nfk);
    end
```

```
        w = find(strt(s) ~= -1);
        ngood = length(w);
        ltmax=0;
        if ngood ~= 0
           s = s(w);
           for k=1:ngood
              map = [map;isort( strt(s(k)):fnsh(s(k)))];
           end
           map = map(2:end);
%             if length(map) == 2
%                if (map(1) - map(2)) == 0
%                   map = unique(map);
%                end
%             end
           %   map = map(umap);
           %end
           % find those trival bonds
           distq = zeros(length(map),1);
           for d=1:dim
              distq = distq + (xyi(map,d) - pos(j,d)).^2;
           end
           ltmax = distq < maxdisq;
           rowtot(j) = sum(ltmax);
           if rowtot(j) >= 1
              w = find(ltmax == 1);
              coltot( map(w) ) = coltot( map(w)) +1;
              which1(j) = map( w(1) );
           end
```

```
        end
    end
    ntrk = fix(n - sum(rowtot == 0));
    w = find( rowtot == 1);
    ngood = length(w);
    if ngood ~= 0
        ww = find(coltot( which1(w) ) == 1);
        ngood = length(ww);
        if ngood ~= 0
            %disp(size(w(ww)))
            resx(ispan,w(ww)) = eyes( which1(w(ww)));
            found( which1( w(ww))) = 1;
            rowtot( w(ww)) = 0;
            coltot( which1(w(ww))) = 0;
        end
    end
    labely = find( rowtot > 0);
    ngood = length(labely);
    if ngood ~= 0
        labelx = find( coltot > 0);
        nontrivial = 1;
    else
        nontrivial = 0;
    end
else
    %   or: Use simple N^2 time routine to calculate trivial bonds
    % let's try a nice, loopless way!
    % don't bother tracking perm. lost guys.
```

```
wh = find( pos(:,1) >= 0);
ntrack = length(wh);
if ntrack == 0
   'There are no valid particles to track idiot!'
   break
end
xmat = zeros(ntrack,m);
count = 0;
for kk=1:ntrack
   for ll=1:m
      xmat(kk,ll) = count;
      count = count+1;
   end
end
count = 0;
for kk=1:m
   for ll=1:ntrack
      ymat(kk,ll) = count;
      count = count+1;
   end
end
xmat = (mod(xmat,m) + 1);
ymat = (mod(ymat,ntrack) +1)';
[lenxn,lenxm] = size(xmat);
%     whos ymat
%     whos xmat
%     disp(m)
for d=1:dim
```

```
    x = xyi(:,d);
    y = pos(wh,d);
    xm = x(xmat);
    ym = y(ymat(1:lenxn,1:lenxm));
    if size(xm) ~= size(ym)
        xm = xm';
    end
    if d == 1
        dq = (xm -ym).^2;
        %dq = (x(xmat)-y(ymat(1:lenxn,1:lenxm))).^2;
    else
        dq = dq + (xm-ym).^2;
        %dq = dq + (x(xmat)-y(ymat(1:lenxn,1:lenxm)) ).^2;
    end
end
ltmax = dq < maxdisq;
% figure out which trivial bonds go with which
rowtot = zeros(n,1);
rowtot(wh) = sum(ltmax,2);
if ntrack > 1
    coltot = sum(ltmax,1);
else
    coltot = ltmax;
end
which1 = zeros(n,1);
for j=1:ntrack
    [mx, w] = max(ltmax(j,:));
    which1(wh(j)) = w;
```

```
    end
    ntrk = fix( n - sum(rowtot == 0));
    w= find( rowtot == 1) ;
    ngood = length(w);
    if ngood ~= 0
       ww = find(coltot(which1(w)) == 1);
       ngood = length(ww);
       if ngood ~= 0
          resx( ispan, w(ww) ) = eyes( which1( w(ww)));
          found(which1( w(ww))) = 1;
          rowtot(w(ww)) = 0;
          coltot(which1(w(ww))) = 0;
       end
    end
    labely = find( rowtot > 0);
    ngood = length(labely);
    if ngood ~= 0
       labelx = find( coltot > 0);
       nontrivial = 1;
    else
       nontrivial = 0;
    end
end
%THE TRIVIAL BOND CODE ENDS
if nontrivial
    xdim = length(labelx);
    ydim = length(labely);
    %  make a list of the non-trivial bonds
```

```
bonds = zeros(1,2);
bondlen = 0;
for j=1:ydim
   distq = zeros(xdim,1);
   for d=1:dim
      %distq
      distq = distq + (xyi(labelx,d) - pos(labely(j),d)).^2;
      %distq
   end
   w= find(distq <  maxdisq)' - 1;
   ngood = length(w);
   newb = [w;(zeros(1,ngood)+j)];
   bonds = [bonds;newb'];
   bondlen = [ bondlen;distq( w + 1) ];
end
bonds = bonds(2:end,:);
bondlen = bondlen(2:end);
numbonds = length(bonds(:,1));
mbonds = bonds;
max([xdim,ydim]);
if max([xdim,ydim]) < 4
   nclust = 1;
   maxsz = 0;
   mxsz = xdim;
   mysz = ydim;
   bmap = zeros(length(bonds(:,1)+1),1) - 1;
else
   %   THE SUBNETWORK CODE BEGINS
```

```
lista = zeros(numbonds,1);
listb = zeros(numbonds,1);
nclust = 0;
maxsz = 0;
thru = xdim;
while thru ~= 0
   %  the following code extracts connected
   %   sub-networks of the non-trivial
   %   bonds.  NB: lista/b can have redundant entries due to
   %   multiple-connected subnetworks
   w = find(bonds(:,2) >= 0);
%              size(w)
   lista(1) = bonds(w(1),2);
   listb(1) = bonds(w(1),1);
   bonds(w(1),:) = -(nclust+1);
   bonds;
   adda = 1;
   addb = 1;
   donea = 0;
   doneb = 0;
   if (donea ~= adda) | (doneb ~= addb)
      true = 0;
   else
   true = 1;
   end
   while ~true
      if (donea ~= adda)
         w = find(bonds(:,2) == lista(donea+1));
```

```
        ngood = length(w);
        if ngood ~= 0
            listb(addb+1:addb+ngood,1) = bonds(w,1);
            bonds(w,:) = -(nclust+1);
            addb = addb+ngood;
        end
        donea = donea+1;
    end
    if (doneb ~= addb)
        w = find(bonds(:,1) == listb(doneb+1));
        ngood = length(w);
        if ngood ~= 0
            lista(adda+1:adda+ngood,1) = bonds(w,2);
            bonds(w,:) = -(nclust+1);
            adda = adda+ngood;
        end
        doneb = doneb+1;
    end
  if (donea ~= adda) | (doneb ~= addb)
      true = 0;
  else
      true = 1;
  end
end
[pp,pqx] = sort(listb(1:doneb));
%unx =  unq(listb(1:doneb),pqx);
%implanting unq directly
    arr = listb(1:doneb);
```

```
    q = arr(pqx);
    indices = find(q ~= circshift(q,-1));
    count = length(indices);
    if count > 0
        unx = pqx(indices);
    else
        unx = length(q) -1;
    end
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
xsz = length(unx);
[pp,pqy] = sort(lista(1:donea));
%uny =  unq(lista(1:donea),pqy);
%implanting unq directly
    arr = lista(1:donea);
    q = arr(pqy);
    indices = find(q ~= circshift(q,-1));
    count = length(indices);
    if count > 0
        uny = pqy(indices);
    else
        uny = length(q) -1;
    end
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
ysz = length(uny);
if xsz*ysz > maxsz
    maxsz = xsz*ysz;
    mxsz = xsz;
    mysz = ysz;
```

```
        end
        thru = thru -xsz;
        nclust = nclust + 1;
    end
    bmap = bonds(:,2);
end
% THE SUBNETWORK CODE ENDS
% put verbose in for Jaci
%   THE PERMUTATION CODE BEGINS
for nc =1:nclust
    w = find( bmap == -1*(nc));
    nbonds = length(w);
    bonds = mbonds(w,:);
    lensq = bondlen(w);
    [pq,st] = sort( bonds(:,1));
    %un = unq(bonds(:,1),st);
      %implanting unq directly
        arr = bonds(:,1);
        q = arr(st);
        indices = find(q ~= circshift(q,-1));
        count = length(indices);
        if count > 0
            un = st(indices);
        else
            un = length(q) -1;
        end
      %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
    uold = bonds(un,1);
```

```
nold = length(uold);
%un = unq(bonds(:,2));
%implanting unq directly
indices = find(bonds(:,2) ~= circshift(bonds(:,2),-1));
count = length(indices);
  if count > 0
    un = indices;
  else
    un = length(bonds(:,2)) -1;
  end
 %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
unew = bonds(un,2);
nnew = length(unew);
if nnew > 5
  rnsteps = 1;
  for ii =1:nnew
    rnsteps = rnsteps * length( find(bonds(:,2) == ...
      unew(ii)));
    if rnsteps > 5.e+4
      disp('Warning: difficult combinatorics encountered.')
    end
    if rnsteps > 2.e+5
      disp(['Excessive Combinitorics you FOOL LOOK WHAT YOU HAVE' ...
          ' DONE TO ME!!!'])
      return
    end
  end
end
```

```
st = zeros(nnew,1);
fi = zeros(nnew,1);
h = zeros(nbonds,1);
ok = ones(nold,1);
nlost = (nnew - nold) > 0;
for ii=1:nold
   h(find(bonds(:,1) == uold(ii))) = ii;
end
st(1) = 1 ;
fi(nnew) = nbonds; % check this later
if nnew > 1
   sb = bonds(:,2);
   sbr = circshift(sb,1);
   sbl = circshift(sb,-1);
   st(2:end) = find( sb(2:end) ~= sbr(2:end)) + 1;
   fi(1:nnew-1) = find( sb(1:nbonds-1) ~= sbl(1:nbonds-1));
end
%          if i-1 == 13
%             hi
%          end
checkflag = 0;
while checkflag ~= 2
   pt = st -1;
   lost = zeros(nnew,1);
   who = 0;
   losttot = 0;
   mndisq = nnew*maxdisq;
   while who ~= -1
```

```
if pt(who+1) ~= fi(who+1)
    w = find( ok( h( pt( who+1 )+1:fi( who+1 ) ) ) ); % check this -1
    ngood = length(w);
    if ngood > 0
        if pt(who+1) ~= st(who+1)-1
            ok(h(pt(who+1))) = 1;
        end
        pt(who+1) = pt(who+1) + w(1);
        ok(h(pt(who+1))) = 0;
        if who == nnew -1
            ww = find( lost == 0);
            dsq = sum(lensq(pt(ww))) + losttot*maxdisq;
            if dsq < mndisq
                minbonds = pt(ww);
                mndisq = dsq;
            end
        else
            who = who+1;
        end
    else
        if ~lost(who+1) & (losttot ~= nlost)
            lost(who+1) = 1;
            losttot = losttot + 1;
            if pt(who+1) ~= st(who+1) -1;
                ok(h(pt(who+1))) = 1;
            end
            if who == nnew-1
                ww = find( lost == 0);
```

```
            dsq = sum(lensq(pt(ww))) + losttot*maxdisq;
            if dsq < mndisq
               minbonds = pt(ww);
               mndisq = dsq;
            end
         else
           who = who + 1;
         end
      else
         if pt(who+1) ~= (st(who+1) -1)
            ok(h(pt(who+1))) = 1;
         end
         pt(who+1) = st(who+1) -1;
         if lost(who+1)
            lost(who+1) = 0;
            losttot = losttot -1;
         end
         who = who -1;
      end
   end
else
   if ~lost(who+1) & (losttot ~= nlost)
      lost(who+1) = 1;
      losttot = losttot + 1;
      if pt(who+1) ~= st(who+1)-1
         ok(h(pt(who+1))) = 1;
      end
      if who == nnew -1
```

```
            ww = find( lost == 0);
            dsq = sum(lensq(pt(ww))) + losttot*maxdisq;
            if dsq < mndisq
               minbonds = pt(ww);
               mndisq = dsq;
            end
         else
            who = who + 1;
         end
      else
         if pt(who+1) ~= st(who+1) -1
            ok(h(pt(who+1))) = 1;
         end
         pt(who+1) = st(who+1) -1;
         if lost(who+1)
            lost(who+1) = 0;
            losttot = losttot -1;
         end
         who = who -1;
      end
   end
end
checkflag = checkflag + 1;
if checkflag == 1
   plost = min([fix(mndisq/maxdisq) , (nnew -1)]);
   if plost > nlost
      nlost = plost;
   else
```

```
                checkflag = 2;
            end
        end
    end
    %   update resx using the minimum bond configuration
    resx(ispan,labely(bonds(minbonds,2))) = eyes(labelx(bonds(minbonds,1)+1));
    found(labelx(bonds(minbonds,1)+1)) = 1;
  end
  %   THE PERMUTATION CODE ENDS
end
w = find(resx(ispan,:) >= 0);
nww = length(w);

if nww > 0
  pos(w,:) = xyzs( resx(ispan,w) , 1:dim);
  if goodenough > 0
    nvalid(w) = nvalid(w) + 1;
  end
end  %go back and add goodenough keyword thing
newguys = find(found == 0);
nnew = length(newguys);
if (nnew > 0) % & another keyword to workout inipos
  newarr = zeros(zspan,nnew) -1;
  resx = [resx,newarr];
   resx(ispan,n+1:end) = eyes(newguys);
  pos = [[pos];[xyzs(eyes(newguys),1:dim)]];
  nmem = zeros(nnew,1);
  mem = [mem;nmem];
```

```
        nun = 1:nnew;
        uniqid = [uniqid,((nun) + maxid)];
        maxid = maxid + nnew;
        if goodenough > 0
            dumphash = [dumphash;zeros(1,nnew)'];
            nvalid = [nvalid;zeros(1,nnew)'+1];
        end
        % put in goodenough
        n = n + nnew;
    end
 else
    ' Warning- No positions found for t='
end
w = find( resx(ispan,:) ~= -1);
nok = length(w);
if nok ~= 0
    mem(w) =0;
end
mem = mem + (resx(ispan,:)' == -1);
wlost = find(mem == memory_b+1);
nlost =length(wlost);
if nlost > 0
    pos(wlost,:) = -maxdisp;
    if goodenough > 0
        wdump = find(nvalid(wlost) < goodenough);
        ndump = length(wdump);
        if ndump > 0
            dumphash(wlost(wdump)) = 1;
```

```
        end
    end
    % put in goodenough keyword stuff if
end
if (ispan == zspan) | (i == z)
    nold = length(bigresx(1,:));
    nnew = n-nold;
    if nnew > 0
        newarr = zeros(z,nnew) -1;
        bigresx = [bigresx,newarr];
    end
    if goodenough > 0
        if (sum(dumphash)) > 0
            wkeep = find(dumphash == 0);
            nkeep = length(wkeep);
            resx = resx(:,wkeep);
            bigresx = bigresx(:,wkeep);
            pos = pos(wkeep,:);
            mem = mem(wkeep);
            uniqid = uniqid(wkeep);
            nvalid = nvalid(wkeep);
            n = nkeep;
            dumphash = zeros(nkeep,1);
        end
    end
    % again goodenough keyword
    if quiet~=1
```

```
        disp(strcat(num2str(i), ' of ' ,num2str(z), ' done.  Tracking  ',num2str(ntrk),' particles  ', num2str(n),'
tracks total'));
    end
    bigresx(i-(ispan)+1:i,:) = resx(1:ispan,:);
    resx = zeros(zspan,n) - 1;
    wpull = find(pos(:,1) == -maxdisp);
    npull = length(wpull);
    if npull > 0
       lillist = zeros(1,2);
       for ipull=1:npull
          wpull2 = find(bigresx(:,wpull(ipull)) ~= -1);
          npull2 = length(wpull2);
          thing = [bigresx(wpull2,wpull(ipull)),zeros(npull2,1)+uniqid(wpull(ipull))];
          lillist = [lillist;thing];
       end
       olist = [[olist];[lillist(2:end,:)]];
    end
    wkeep = find(pos(:,1) >= 0);
    nkeep = length(wkeep);
    if nkeep == 0
          'Were going to crash now, no particles....'
    end
    resx = resx(:,wkeep);
    bigresx = bigresx(:,wkeep);
    pos = pos(wkeep,:);
    mem = mem(wkeep);
    uniqid = uniqid(wkeep);
    n = nkeep;
```

```
        dumphash = zeros(nkeep,1);
        if goodenough > 0
            nvalid = nvalid(wkeep);
        end
    end
end
if goodenough > 0
    nvalid = sum(bigresx >= 0 ,1);
    wkeep = find(nvalid >= goodenough);
    nkeep = length(wkeep);
    if nkeep == 0
        for i=1:10
        disp('You are not going any further, check your params and data')
        end
        disp('the code broke at line 1045')
        return
    end
    if nkeep < n
        bigresx = bigresx(:,wkeep);
        n = nkeep;
        uniqid = uniqid(wkeep);
        pos = pos(wkeep,:);
    end
end
wpull = find( pos(:,1) ~= -2*maxdisp);
npull = length(wpull);
if npull > 0
    lillist = zeros(1,2);
```

```
    for ipull=1:npull
        wpull2 = find(bigresx(:,wpull(ipull)) ~= -1);
        npull2 = length(wpull2);
        thing = [bigresx(wpull2,wpull(ipull)),zeros(npull2,1)+uniqid(wpull(ipull))];
        lillist = [lillist;thing];
    end
    olist = [olist;lillist(2:end,:)];
end
olist = olist(2:end,:);
%bigresx = 0;
%resx = 0;
nolist = length(olist(:,1));
res = zeros(nolist,dd+1);
for j=1:dd
    res(:,j) = xyzs(olist(:,1),j);
end
res(:,dd+1) = olist(:,2);
% this is uberize included for simplicity of a single monolithic code
ndat=length(res(1,:));
newtracks=res;
%u=unq(newtracks(:,ndat));
% inserting unq
indices = find(newtracks(:,ndat) ~= circshift(newtracks(:,ndat),-1));
        count = length(indices);
        if count > 0
            u = indices;
        else
            u = length(newtracks(:,ndat)) -1;
```

```
        end
ntracks=length(u);
u=[0;u];
for i=2:ntracks+1
    newtracks(u(i-1)+1:u(i),ndat) = i-1;
end
% end of uberize code
tracks = newtracks;
```

A.2.7 Velocity Profile Quantification

```
% Name: velpfrl.m
% Author: Mike Records
% Purpose: Code to calculate velocity profiles from trajectories determined in track.m
% Inputs: the list of data sorted into a series of trajectories sorted so rows with corresponding id numbers
%       are in contiguous blocks, with the time variable a monotonically increasing function inside each
%       block.
% Outputs: Outputs an N X 5 dimensional array containing, 2d velocity vectors, x and y coordinates and
%       timestamp, as well as a representative velocity for each time step
% Procedure: Given the particle trajectories, velocities are calculated for each particle over time. If a
%       particle cannot be identified in consecutive timesteps it is removed from the analysis for that
%       timestep. First images are filtered by grayscale intensity and color to identify each layer of
%       particles.Then the images are spacially Filtered using bpass.m to remove pixel level and non-
%       characteristic length scale noise. pkfnd.m is called to identify tracking particles. Cntrd.m is then
%       called to identify the center of each tracking particle. Then particles without a pair are removed.
%       Velocities and representative locations are calculated for each particle. Representative locations
%       are calculated as the average location of each particle between its initial and final location. A
%       representative velocity is calculated for the surface and the bed as an average of the velocities of
%       the eight closest particles to the center. This velocity is calculated by finding a linearly
%       interpolated velocity between pairs of two particles, then averaging the velocities of all the bead
%       pairs. Finally the data is placed in a format that allows it to be quickly imported from excel and
%       then easily returned to MATLAB for final processing.
 %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
clear; clc;
deltvec = load('delt.txt'); %Load text file containing vector of timesteps
deltp = 4277; %pixels/meter (101.233 pixels/in) (top view camera)
mo = 842;   %Second image in series
mi = 902;   %Final image in series
```

```
istep = 2;  %Image step size
 index = 1;  %How many images have been processed so far
dat(:,:,1) = zeros(25, 4);  %Initialize dat vector w/ a page of zeros to make sure initial page is as long or
longer than any following pages
 %Particle tracking and velocity profile code
for m = mo:istep:mi       %Loop through all timesteps
    %Reinitialize variables between each timestep
    clear pos_lst; pos_lst = []; clear tvec; clear tr;
    %Loop over 1st and 2nd image in each timstep
    for n = 1:2
        %Reinitialize variable between first and second images in timestep
        clear im; clear green; clear abp; clear pk0; clear cnt; clear cntr;
        %Load image
        pic = strcat('DSC_0', int2str(m-4+n*istep), '.jpg');
        eval('im=imread(pic);') %put image name here
        %Create grayscale version of each image used in filtering for particle identification
        tstgry = rgb2gray(im);
         %Filter image by color and intensity
        sz=size(im);
        for i=1:sz(1)
            for j=1:sz(2)
%             if (tstgry(i,j) > 75 && tstgry(i,j) < 140 && im(i,j,1)<80 & im(i,j,2)<130 & im(i,j,3)>90 &
im(i,j,3)<160) %green 5.9.2011
                if (tstgry(i,j) < 60 && im(i,j,1)<90) %black 5.9.2011
                    green(i,j)=1;
                else
                    green(i,j)=0;
                end
```

```
            %Remove basal water ports as potential tracking particles
            %and other artifacts
            %Basal Sliding Long Crop
%               if (((j>155) && (j<210) && (i>805) && (i<940)) || ((j>165) && (j<230) && (i>540) && (i<615)) ||
((j>400) && (j<460) && (i>540) && (i<605)) || ((j>630) && (j<700) && (i>545) && (i<615)) || ((j>165) &&
(i>1000)) || (j<60) || ((j>615) && (j<765))) %Basal Sliding Long Crop
            if (((j>180) && (j<245) && (i>530) && (i<585)) || ((j>415) && (j<478) && (i>530) && (i<580)) ||
((j>645) && (j<705) && (i>530) && (i<590))) %Surface Velocity Short Crop
                green(i,j)=0;
            end
          end
        end
        abp = bpass(green, 2, 20);          %Spatially Filter Image
        pk0 = pkfnd(abp, .7, 20);      %First value is threshold, second number is roughly feature diameter
        cnt = cntrd(abp, pk0, 20);      %Calc centroid of each blob
%       imagesc(im)
%       hold on
%       plot(cnt(:,1), cnt(:,2), '*')
        %%%%Linking particles to form trajectories
        [j,k] = size(cnt);
        tvec = ones(j,1)*(m-4+n*istep);          %Build time vector for each time
        cntr = [cnt(:,1), cnt(:,2), tvec];              %Build vector (x, y, t)
        pos_lst = [pos_lst; cntr];                    %Position List Vector for track.m
        %result = track(positionlist, maxdisp, param)
        %Once particle locations have been identified for both images in set
        %Run tracking algorithm and calculate velocities
        if n == 2
          m
```

```
%Run particle tracking algorithm
tr = track(pos_lst, 30);
%Remove tracking particles which were not resolved in both
%images of sequence
trc = 1;
trfc = 1;
clear trf;
while trc<length(tr)
    if tr(trc,4) == tr(trc+1,4)
        trf(trfc,:) = tr(trc,:);
        trf(trfc+1,:) = tr(trc+1,:);
        trfc = trfc+2;
    end
    trc = trc+1;
end
 %Calc and plot velocity at each point of interest
 J = length(trf)/2;
 clear vx; clear vy; clear v; clear plxx; clear plyy; clear plx; clear ply;
 delt = deltvec((m-mo)/istep+1);
 %replaced m with k as for loop index
 for k = 1:J
     vx(k,1) = -(trf(2*k,1) - trf(2*k-1,1))/delt/deltp;
     vy(k,1) = -(trf(2*k,2) - trf(2*k-1,2))/delt/deltp;
     v(k,1) = sqrt(vx(k)^2 + vy(k)^2);
     plxx(k,1) = (trf(2*k-1,1) + trf(2*k,1))/2;
     plyy(k,1) = (trf(2*k-1,2) + trf(2*k,2))/2;
 end
 %Build 3D vector of velocity profiles w/ one page per timestep
```

```
            dat(:,:,index+1) = [[plxx plyy vx -vy]; zeros(length(dat(:,1,1)) - length(plxx),4)];
            index = index +1;
        end
    end
end
%Decompose 3D dat array into a 2D array for import to EXCEL
datxls = [];
for m = mo:istep:mi
    index = (m-mo)/istep+1
    datxls = [datxls; [dat(:,:,index+1) ones(length(dat(:,:,index+1)),1)*m]];
end
%Loop to plot velocity profile vectors on img @ each timestep
%To check for outliers & incorrectly identified tracking particles
for m = mo:istep:mi
    m
    index = (m-mo)/istep+1;
    pic = strcat('DSC_0', int2str(m-4+n*istep), '.jpg');
    eval('im=imread(pic);') %put image name here
    imagesc(im)
    hold on
    plot(dat(:,1,index+1), dat(:,2,index+1), '*y')
    quiver(dat(:,1,index+1), dat(:,2,index+1), dat(:,3,index+1), dat(:,4,index+1))
    pause
    close
end
%Bring velocity data back in from excel, split it back up by page/tstep
datxlsed = [];
dated = zeros(25, 5, (mi-mo)/istep);     %Preallocate
```

```
page = 1;
count = 0;
for m = 1:1:length(datxlsed)-1
    if datxlsed(m,5) == datxlsed(m+1,5)
        count = count + 1;
        dated(count,:,page) = datxlsed(m,:);
    else
        dated(count,:,page) = datxlsed(m,:);
        count = 0;
        page = page + 1;
    end
end
%Calc representative velocity for each timestep
xcent = 447;
ycent = 556;
for m = 1:((mi-mo)/2+1)
    xshft = dated(:,1,m) - xcent;
    yshft = dated(:,2,m) - ycent;
    datshft = [xshft, yshft, dated(:,3,m), dated(:,4,m), dated(:,5,m)];
    sortdat = sortrows(datshft,2);
    %Break it into greater and less than zero then sort
    clear possort; clear negsort;
    posn = 1;
    negn = 1;
    for n = 1:length(sortdat)
        if sortdat(n,2) > 0
            possort(posn,:) = sortdat(n,:);
            posn = posn+1;
```

```
        else
            negsort(negn,:) = sortdat(n,:);
            negn = negn+1;
        end
    end
    negsort(:,2) = abs(negsort(:,2));
    %Select 4 closest upstream and downstream particles to centerline
    pos4 = possort(1:4,:);
    neg4 = negsort((length(negsort)-3):length(negsort),:);
    a1 = (neg4(1,4)-pos4(1,4))/(neg4(1,2)+pos4(1,2))*(pos4(1,2))+pos4(1,4);
    a2 = (neg4(2,4)-pos4(2,4))/(neg4(2,2)+pos4(2,2))*(pos4(2,2))+pos4(2,4);
    a3 = (neg4(3,4)-pos4(3,4))/(neg4(3,2)+pos4(3,2))*(pos4(3,2))+pos4(3,4);
    a4 = (neg4(4,4)-pos4(4,4))/(neg4(4,2)+pos4(4,2))*(pos4(4,2))+pos4(4,4);
    repv(m,1) = (a1+a2+a3+a4)/4;
    %%%%Should I add a timestep identifier?????
end
plot(repv)
```

A.2.8 Water Pressure Quantification

```
% Name: presspick.m
% Purpose: User facilitated code to calculate the basal water pressure at each pressure tap.
% Inputs: Experimental image range, image discretization, channel bed elevation
% Outputs: Water pressure at each pressure tap for each timestep
Clear; clc;
%coordinates to zero/center board
imgi = 832;
pic = strcat('', int2str(imgi), '.jpg'); eval('im=imread(pic);')
imshow(im);  [xmbrd, ymbrd] = ginput; close
tl = [xmbrd(1), ymbrd(1)]; bl = [xmbrd(2), ymbrd(2)]; tr = [xmbrd(3), ymbrd(3)]; br = [xmbrd(4), ymbrd(4)];
yshftttl = abs(((tl(2) - tr(2)) + (bl(2) - br(2)))/2);
%Mark x location of each pipe
imshow(im);  [mlocx, mlocy] = ginput; close
for k = 1:7
    yshft(k) = (mlocx(8) - mlocx(k))/(mlocx(8) - mlocx(1))*yshftttl;
end
yshft(8) = 0;
%Image Discretization
imshow(im);  [xsa, ysa] = ginput; close
discdel = abs(ysa(2) - ysa(1));
del = discdel/7.875;
clear pdatfl; clear pdatflsft; clear pdatin; clear ptbl; clear ptbltr;
pdat = [ , ]
imgi = 847;
for imgnum = imgi:2:847
    imgnum
    pic = strcat(int2str(imgnum), '.jpg');
```

```
    eval('im=imread(pic);') %put image name here
    imshow(im);      [xsa, ysa] = ginput;
    close
    pdat = [pdat, ysa];
end
%flip and shift to zero at bottom white tabs at lower end of board
pdatfl = br(2) - pdat; %distance above bottom right tab
for j = 1:8
        pdatflsft(j,:) = pdatfl(j,:) - yshft(j);
end
%Zero pressure measurements at table, convert water elevations to inches
pdatin = pdatflsft/del; ptbl = pdatin + 4; ptbltr = ptbl';
```

Appendix 3 Preliminary Experiments

A preliminary suite of experiments was performed before the quantitative experimental suite. These qualitative experiments were performed in a small test channel to confirm that the necessary processes could be produced before moving to the large channel.

A.3.1 Experimental Setup

The small test channel was an 18” X 6” X 6” box, which like the large channel, had walls on all sides but the downstream end and the top. It had upstream and channel bed basal water ports. Unlike the large channel, the channel bed basal water ports were only located along the channel centerline. There was no water pressure, inflow, or discharge measurement. Instead of the four constant head reservoirs used in the large channel, one large reservoir provided pressurized basal water to the entire channel bed. Plasticine bed topographies were pressed into the bed of the channel.

A.3.2 Bed Topography Investigation

An investigation was performed to see if a bed topography could be developed that would control the distribution of basal water and allow low pressure water cavity formation.

First, a topography free bed was tested. There was no control over the extent of basal water, and a lowing sheet of water covering the entire bed formed. In addition, when thicker water cavities formed along portions of the bed they advected a significant distance downstream over even the short timeframe of the trial. This basal water distribution was unacceptable because widespread distinct water filled cavities did not develop, and cavities that did form were quickly downstream by the flowing PDMS. Second, a series of two-dimensional cross-channel ridges was tested. Water was introduced between each of these ridges via a basal water port located between each set of ridges. Although the ridges did initially control the extent and location of basal water, once water overtopped the ridges it was impossible to control the extent of the water. This was an unacceptable basal water configuration because if the water did not overtop the bed ridges there was no hydraulic connection between the basal water cavities, and when the basal water system was hydraulically connected there was no control over the basal water extent. Third, a series of two-dimensional cross-channel troughs were tested, this bed geometry was very similar to the 2D ridge configuration. A basal water port located at the low point of each trough introduced

water into each trough. This bed topography had the same disadvantages as the ridge system: there was no control over water location once water overtopped the troughs, no hydraulic connection between individual troughs before overtopping, and water only existed near basal water ports. I even tried incising along channel grooves between the consecutive troughs, with the hope of creating a pressure relieve mechanism where water could flow between the basal grooves instead of overtopping them; still this was unable to control the spread of water. The investigatory 2D bed topography can be seen in Figure 18. The flow direction in the picture is from left to right, and red particle tracking beads are visible on the surface. The cross-channel troughs and the incised basal grooves can be seen as areas of the bed which are not made up of the white plasticine. The uncontrollable blue basal water is also visible extending out of the cross-channel troughs.

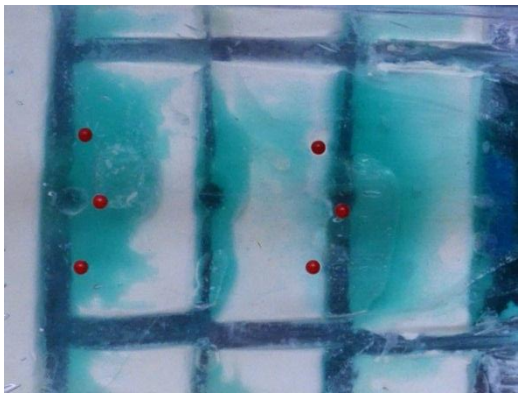

Figure 18: Investigatory 2D bed topography with uncontrollable basal water.

Finally, a three-dimensional checkerboard topography was tested. As described in Chapter 2, this topography was made of half-dome shaped bumps placed on an offset rectangular grid. It worked remarkably well, successfully containing basal water below the top of the bumps. Distinct water filled link and cavity areas developed, and with basal sliding low pressure cavities developed on the downstream side of the bumps. In addition, acceptably low levels of cavity advection were observed over experimental time frames. Just as we had hoped for, this bed geometry produced an ice analog-bed interface where the PDMS bottom slid over patches of the bed while other areas of the PDMS bottom were supported by water cavities.

With the goal of specifically and singly testing the proposed hydraulic jacking phenomena, I tried to develop a two-dimensional roche moutonnée type bed topography. This geometry featured a series of ridge and troughs, where I hoped low pressure water cavities would form behind the lee side of each bed bump. Unfortunately, I experienced the same problems as the other 2D topographies, and I was unable to control cavity geometry during formation and closure. We believe that the difficulty in producing realistic basal water without the checkerboard topography points to the validity of the 3D checkerboard bed geometry over the 2D bed topographies.

Appendix 4 Experimental Method

Preparation of each experiment and the experimental run methodology, covered briefly in Chapter 2, are described thoroughly below. The preparation and run methodology centered on producing accurate and reproducible experimental conditions for all experiments in the quantitative experimental suite.

A.4.1 Experiment Preparation

To prepare the channel it was first leveled in the channel frame. The downstream end of the channel was closed and the basal water ports were plugged. The closure of the downstream wall and basal water ports was done to prevent PDMS flow into these areas. During this time a 1" thick layer of PDMS was allowed to settle in the accessory channel. The accessory channel had the same dimensions as the main flow channel allowing a layer of PDMS that would fit exactly into the main channel to be prepared. The volume of PDMS to be prepared in the accessory channel for a one inch lift was first measured gravimetrically to ensure the correct PDMS depth. Then the lubricant system was applied to the bed. First, the grease was brushed onto the bed, and then glycerin was poured over the grease and smoothed uniformly with a brush. The lubricant layers were applied to be as thin as possible while still producing complete coverage. Next, the basal tracking particles were spread widely and evenly across the bed, ensuring that bed areas free of and covered by bed bumps would be well represented in the basal velocity profiles. Finally, the 1st layer of PDMS was laid directly on the bed.

Because of its fluid nature, the PDMS had to be moved in rectangular slabs from the accessory channel. Moving from one end of the channel to the other, PDMS slabs were laid until the entire flow channel was covered with the one inch layer of PDMS. The biggest difficulty in the laying process was preventing significant glycerin displacement, while introducing as few air bubbles as possible. Inevitably, some lubricant was displaced, creating bed areas where the PDMS stuck directly to the bed. After placing the first layer, the PDMS was given 24 hours to settle; allowing air bubbles to rise out of the PDMS and the PDMS surface to level. While the 1st layer settled, the 2nd later was prepared and settled in the accessory channel. After this settling period, a bead grid was placed on the surface of the PDMS in the flow channel. Then, like the 1st layer, the 2nd one inch thick PDMS layer was added in slabs on top of the

first layer. Again, the PDMS was given 24 hours to settle in the experimental channel before the surface bead grid was placed.

A.4.2 Final Experiment Setup

Immediately before starting each experiment the same experimental preparations were implemented. First, the water supply pump was started and the CHRs were filled. However, the CHR supply pipes were not yet connected to the basal water supply system. Next, the data collection systems (cameras & discharge balance) were started. Third, the water supply pipes were filled, but to ensure as little air as possible entered the basal water system they were not immediately attached to the basal water ports. Fourth, the basal water port plugs, which prevented PDMS from flowing down into and clogging the ports, were removed and the water supply pipes were immediately attached. This minimized PDMS flow into the water ports. Finally, the channel was tipped, the downstream end opened, and the water supply control valves opened. This marked the beginning of the 15 minute steady state basal water period.

A.4.3 Experiment Run

Once the experimental period began, the PDMS was allowed to flow down channel while the CHRs continued to provide a distributed, pressurized water supply to the bed. During this period the only adjustment made to the system was trimming excess PDMS from the terminus. This was done when the PDMS flowed past the bed topography to the point that basal water outflow was limited. Essentially, the bed bumps partially supported the PDMS, beyond the end of the bumps the PDMS easily stuck to the bed. No PDMS was added upstream over the entire experiment duration for several reasons. First, adding PDMS upstream affected imaging by making the PDMS surface uneven and introducing air bubbles, both of which prevented clear visualization of internal and bed processes. Second, adding PDMS upstream produced high displacement and velocity pulses. These enhanced displacements would have completely overwhelmed the measurement of the very small displacements associated with an analog basal water induced sliding event. It is recognized that this is not a completely representative of a prototype glacier at equilibrium, but given the experimental constraints and our interest in short term enhanced sliding events it is acceptable. Given more time a system could be developed that would allow steady addition of PDMS upstream through a large PDMS reservoir.

Appendix 5 Supplementary Experimental Results

The supplementary experimental results section contains figures, plots, images, and discussion that were either considered not important enough for the article's main body, or that did not accurately simulate prototype behavior. This section is meant as a supplement to the main article, not repeating previously discussed material, but rather further clarifying and explaining the experiments that were performed.

A.5.1 Experiment Summary

Figure 19 shows sample top view imagery for each experiment in the quantitative experimental suite. Figure 19a. shows the incised developed conduits used to model high efficiency conduits. Figure 19b, clearly shows the formation of low pressure water cavities on the lee side of many of the bed topography bumps. For CB2 Sliding, the bed topography clearly contains the basal water extent, but the lee cavities are indistinct. Without distinct lee cavities, bed separation as a flow enhancement mechanism could still be tested, but hydraulic jacking could not. The difference in bed bump radius is clearly visible between Figure 19b & c. Figure 19d) shows several large areas without basal water, and a complete lack of lee cavity formation. A clear difference in water dye color between the CB1 experiments and the CB2 experiments is due to an upgrade in water dye from food coloring to FD&C Blue.

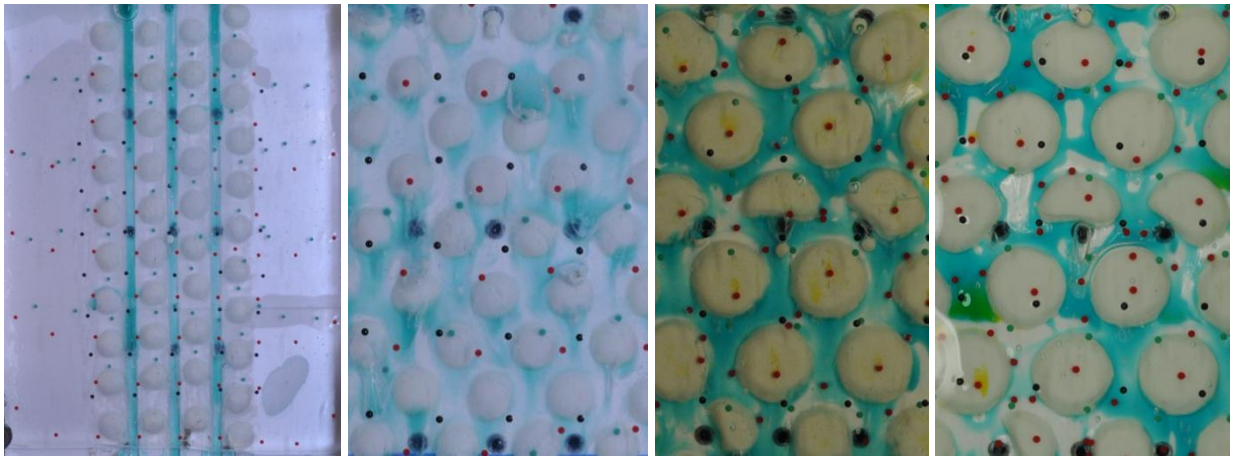

Figure 19: Above channel photographic snapshot overview of the experimental suite: (a) CB1 Developed Conduit Experiment. (b) CB1 Sliding Experiment. (c) CB2 Sliding Experiment. (d) CB2 Patchy Sliding Experiment.

A.5.2 CB1 Developed Conduit Experiment

As discussed in Chapter 4, maintaining the efficient conduit system proved to be extremely difficult. Unlike the linked cavity sliding experiments, CB1 Developed Conduit struggled to develop a basal water system that was at equilibrium with the water supply. Figure 20, a plot of water pressure vs. time, shows the significant variation in water pressure over time in the conduits. The extremely variable water pressures are diagnosed as a sign of the difficulty in maintaining analog conduits even with high water pressures. Some of the observed variability is attributed imperfect model lubricant and the limited number of conduits.

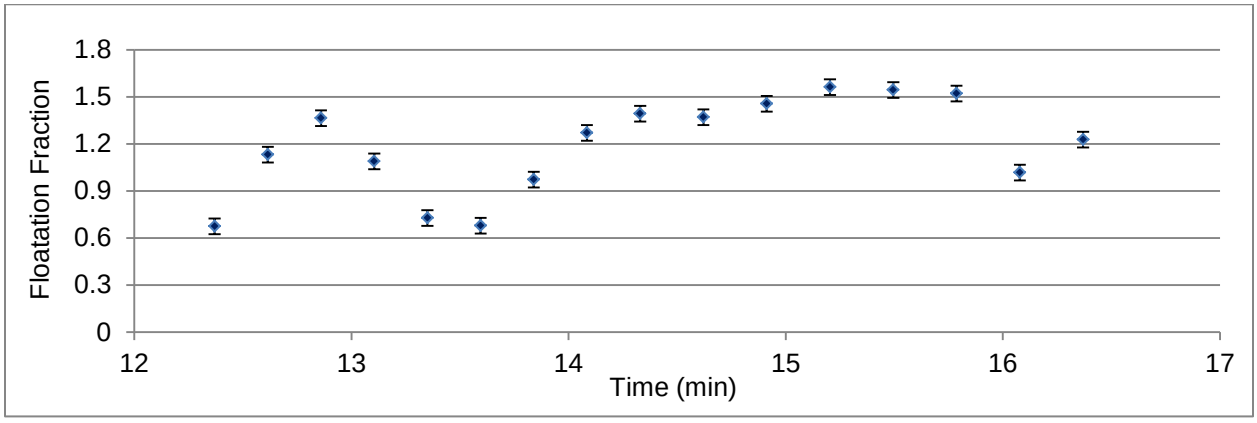

Figure 20: CB1 Sliding Experiment: water pressure vs. time.

A.5.3 CB2 Sliding Experiment

In addition to the steady state period reported on in the main article, CB2 Sliding featured two distinct transient periods. After the SS period the CHRs were raised three times, and then lowered once. Like the series of pressurized water pulses in CB1 Sliding, the increases in CHR elevation were supposed to mimic a series of diurnal melt events. Different from CB1 Sliding, the CHRs were not lowered between each high water pressure pulse, but were rather held constant at each elevation than raised further. Water pressures for the entire experimental duration of CB1 Sliding are shown in Figure 21.

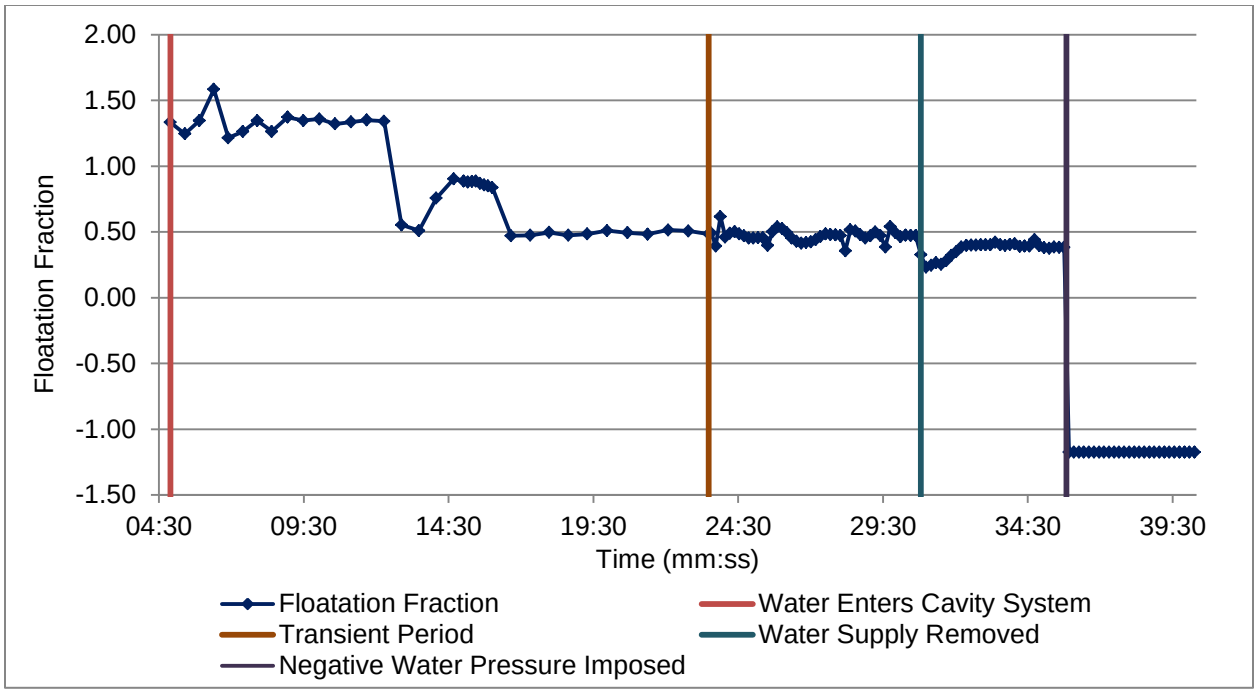

Figure 21: CB2 Sliding Experiment: water pressure vs. time (full experimental duration).

No clear relationship between each diurnal pulse and surface velocities was observed. The lack of correlation between increasing basal water pressure and surface velocities is attributed to several factors. First, since the lubricant film cannot support water pressures there was no decrease in coulomb friction associated with an increase in water pressure. Second, unlike CB1 Sliding, there were no low pressure periods to allow cavity collapse and a development of an inefficient hydraulic system. Third, I propose that the increases in the water pressure were so large that they overwhelmed the basal water system. Rather than a controlled closure and reopening of lee cavities as seen in CB1 Sliding, a complete restructuring of the link cavity system occurred with each water pressure increase, this was seen in the spikes and dips in water pressure as the system adapted. Cavities did not simply grow in the downstream direction, but instead grew in every direction. Fourth, and most important, the distinct wedge shaped lee cavities that developed in CB1 Sliding were indistinct in this experiment. To produce a hydraulic jacking effect these cavities must grow in the down-channel direction. With water simply confined below the bed topography an increase in storage correlated to vertical growth in the basal water system rather than downstream cavity growth.

The significant decrease in the CHR elevation that followed the high pressure transient period was meant to mimic the speedy drainage of a prototype link cavity system. This was supposed to be analogous to the end of the summer on the prototype where a developed drainage system can quickly and efficiently drain melt water input to the bed. To amplify the effect, and to quickly drain the basal water, a constant and strongly negative water pressure was imposed at the bed. Hypothesizing that cavity closure drives some sort of “hydraulic closing" where a negative dS/dt slows ice velocities, dS/dt and sliding velocity vs. time were plotted in Figure 22.

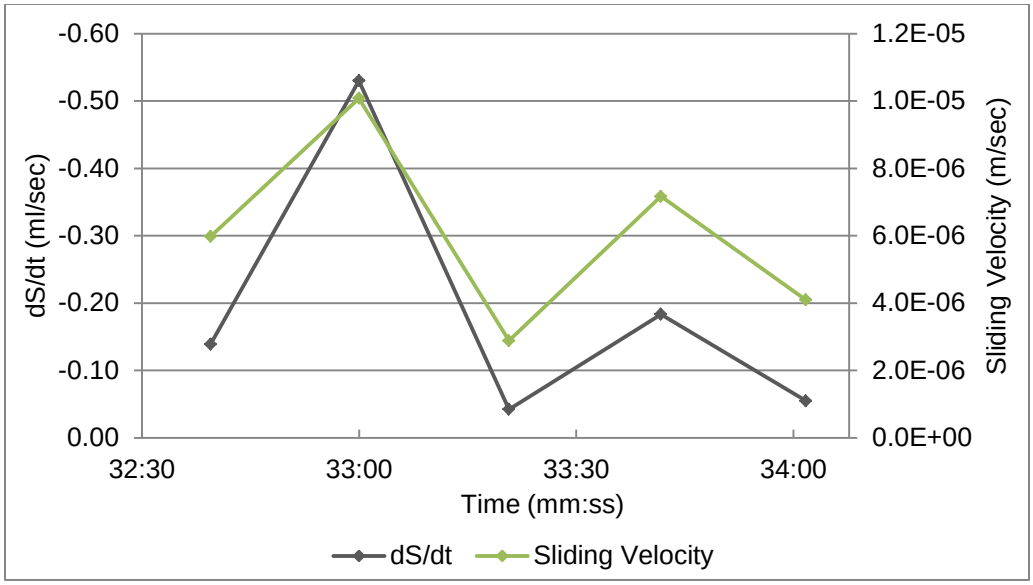

Figure 22: CB2 Sliding Experiment: dS/dt & sliding velocity vs. time (imposed negative water pressure period).

However, the opposite effect seemed to be shown: more negative dS/dt produced higher sliding velocities. On the surface this appeared to indicate that cavity closure actually enhanced ice velocities. However, I suspect that, as in CB1 sliding, there is a short lag between dS/dt and its effect on sliding velocities. If sliding velocity was lagged one time step behind dS/dt, Figure 22 would show a strong relationship between negative dS/dt and sliding velocity as was seen in CB1 Sliding. The hydraulic closing event of CB2 Sliding is not included in the main article because it is not seen as being a rigorous enough experiment. First, there were only five data points comparing dS/dt. Second, the distinct wedge shaped lee cavities are barely distinguishable.

A.5.4 CB2 Patchy Sliding Experiment

In addition to the steady state period of CB2 Patchy Sliding, reported in the main article, I attempted a series of stepped increases in CHR elevation, just as I did in CB2 Sliding. For each diurnal period, the CHRs were raised over the period of a minute, held steady for a minute, and then the process was repeated. These incremental increases were again meant to simulate a series of spring events. As Figure 24 and Figure 24 shows, the results are quite complex with no apparent relationship between each stepped increase in CHR elevation and basal water pressure or ice flow.

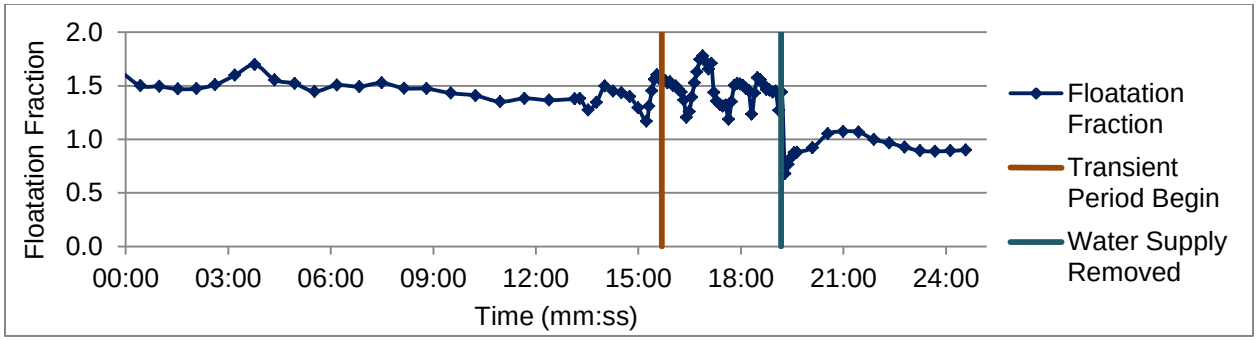

Figure 23: CB2 Patchy Sliding Experiment: water pressure vs. time (experiment duration).

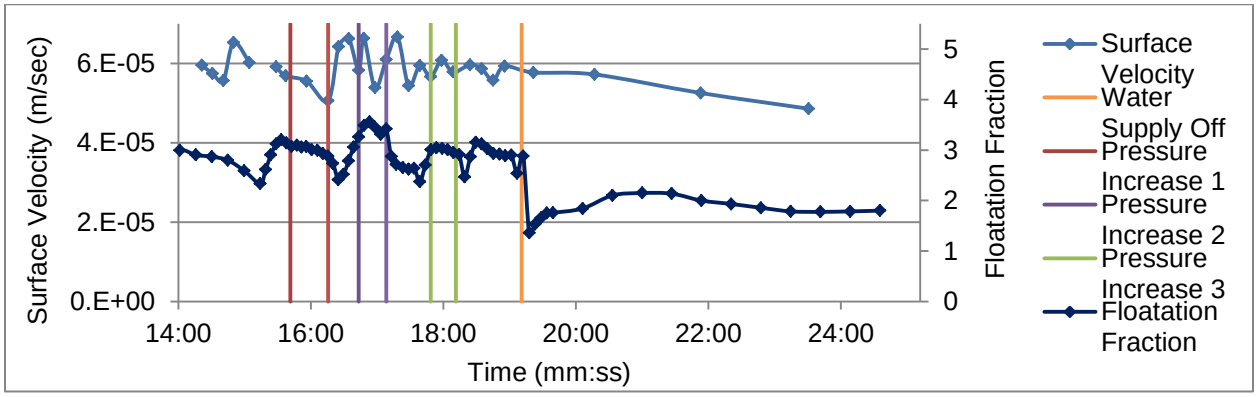

Figure 24: CB2 Patchy Sliding Experiment: surface velocity & water pressure vs. time (transient pressure period).

In fact there was not even a clear relationship between water pressure and ice flow. The lack of correlation was attributed to two factors. The first factor is attributed to the PDMS-bed boundary condition: either basal water cavities existed, or the PDMS was stuck directly to the bed. So, an increase in water pressure was not transmitted through the water film prototype anywhere. Second, because there was not

widespread basal sliding, no lee cavities were observed in the study area. With increased water pressure cavities grew up instead of down-glacier, as they would have with in the presence basal sliding, so they produced no net down-glacier sliding force.

Appendix 6 Original Data

A.6.1 Viscosity Determination Experiment

Table 5: Falling Ball Viscometer Experiment: a) parameters b) data

Ball Radius (m)	6.35E-03
Ball Volume (m^3)	1.07E-06
Ball Mass (kg)	8.36E-03
Ball Density (kg/m^3)	7.79E+03
Viscometer Diameter (m)	0.044
Correction Factor	0.446
Gravity (m/sec^2)	9.81
PDMS Density (kg/m^3)	965

Ball Position (m)	Time (sec)
0.000	0
0.013	1295
0.025	2604
0.072	6570
0.085	7900
0.098	9200
0.072	6596

Table 6: Falling Ball Viscometer Experiment: results

Fall Time (sec)	9200
Fall Distance (m)	.0722
Settling Velocity (m/sec)	1.10E-05
Dynamic Viscosity (Pa sec)	2.44E+04

A.6.2 Density Determination Experiment

Table 7: Density Determination Experiment: data and results

Trial Number	1	2	3
Mass PDMS (kg)	0.393	0.46	0.43
Mass Displaced Water (kg)	0.402	0.467	0.444
PDMS Volume (m^3)	4.03E-04	4.68E-04	4.45E-04
Calculated PDMS Density (kg/m^3)	976	983	967
Average PDMS Density (kg/m^3)			975

A.6.3 Absorption Coefficient Determination (Food Coloring, FD&C Blue)

Table 8: Food Coloring Light Transmission Data

ycord (pix)	Depth (in)	Light Intensity
2339	0.000	146.29
2338	0.001	149.03
2337	0.003	150.51
2336	0.004	151.03
2335	0.005	150.96
2334	0.006	150.80
2333	0.008	150.19
2332	0.009	149.91
2331	0.010	149.66
2330	0.011	149.13
2329	0.013	149.42
2328	0.014	149.60
2327	0.015	148.92
2326	0.017	148.24
2325	0.018	147.69
2324	0.019	147.56
2323	0.020	147.38
2322	0.022	147.48
2321	0.023	147.15
2320	0.024	147.16
2319	0.025	147.14
2318	0.027	147.35
2317	0.028	147.12
2316	0.029	147.08
2315	0.030	146.76
2314	0.032	146.23
2313	0.033	146.26
2312	0.034	146.09
2311	0.036	145.85
2310	0.037	145.64
2309	0.038	145.80
2308	0.039	145.65
2307	0.041	145.22
2306	0.042	145.11
2305	0.043	145.16
2304	0.044	145.10
2303	0.046	144.81
2302	0.047	144.62
2301	0.048	144.51
2300	0.050	144.29
2299	0.051	144.09
2298	0.052	144.10
2297	0.053	144.25
2294	0.057	143.27
2293	0.058	143.33
2292	0.060	143.21
2291	0.061	143.01
2290	0.062	143.03
2289	0.064	143.05
2288	0.065	142.71
2287	0.066	142.59
2286	0.067	142.66
2285	0.069	142.43
2284	0.070	141.80
2283	0.071	141.24
2282	0.072	141.26
2281	0.074	141.52
2280	0.075	141.38
2279	0.076	141.05
2278	0.077	140.64
2277	0.079	140.24
2276	0.080	139.99
2275	0.081	139.87
2274	0.083	140.07
2273	0.084	140.49
2272	0.085	140.14
2271	0.086	139.64
2270	0.088	139.31
2269	0.089	139.08
2268	0.090	138.54
2267	0.091	138.35
2266	0.093	138.20
2265	0.094	138.00
2264	0.095	137.62
2263	0.097	137.45
2262	0.098	137.34
2261	0.099	137.41
2260	0.100	137.15
2259	0.102	136.93
2258	0.103	136.80
2257	0.104	136.43
2256	0.105	136.09
2255	0.107	136.14
2254	0.108	135.68
2253	0.109	135.43
2252	0.111	135.38
2249	0.114	134.60
2248	0.116	134.64
2247	0.117	134.45
2246	0.118	134.16
2245	0.119	133.95
2244	0.121	133.87
2243	0.122	133.60
2242	0.123	133.49
2241	0.124	133.41
2240	0.126	133.24
2239	0.127	132.86
2238	0.128	132.38
2237	0.130	132.01
2236	0.131	131.46
2235	0.132	130.78
2234	0.133	130.51
2233	0.135	130.37
2232	0.136	130.27
2231	0.137	130.00
2230	0.138	129.74
2229	0.140	129.58
2228	0.141	129.24
2227	0.142	129.14
2226	0.144	129.03
2225	0.145	129.01
2224	0.146	129.02
2223	0.147	128.91
2222	0.149	128.64
2221	0.150	127.93
2220	0.151	127.60
2219	0.152	127.44
2218	0.154	127.21
2217	0.155	127.26
2216	0.156	126.96
2215	0.158	126.55
2214	0.159	126.23
2213	0.160	126.56
2212	0.161	126.48
2211	0.163	125.98
2210	0.164	125.91
2209	0.165	125.79
2208	0.166	125.66
2207	0.168	125.41

ycord (pix)	Depth (in)	Light Intensity
2204	0.171	125.16
2203	0.173	125.03
2202	0.174	124.78
2201	0.175	124.52
2200	0.177	124.16
2199	0.178	123.86
2198	0.179	123.41
2197	0.180	123.14
2196	0.182	123.09
2195	0.183	122.74
2194	0.184	122.26
2193	0.185	122.25
2192	0.187	122.24
2191	0.188	122.08
2190	0.189	122.01
2189	0.191	121.81
2188	0.192	121.78
2187	0.193	121.70
2186	0.194	121.38
2185	0.196	121.33
2184	0.197	121.22
2183	0.198	120.84
2182	0.199	120.64
2181	0.201	120.67
2180	0.202	120.27
2179	0.203	119.82
2178	0.205	119.52
2177	0.206	119.07
2176	0.207	119.09
2175	0.208	118.93
2174	0.210	118.92
2173	0.211	119.01
2172	0.212	118.75
2171	0.213	118.52
2170	0.215	118.04
2169	0.216	117.64
2168	0.217	117.33
2167	0.218	117.23
2166	0.220	117.08
2165	0.221	116.66
2164	0.222	116.16
2163	0.224	115.71
2162	0.225	115.38
2161	0.226	115.41

ycord (pix)	Depth (in)	Light Intensity
2158	0.230	115.49
2157	0.231	115.78
2156	0.232	115.68
2155	0.234	115.58
2154	0.235	115.73
2153	0.236	115.32
2152	0.238	114.63
2151	0.239	114.54
2150	0.240	114.42
2149	0.241	114.25
2148	0.243	114.13
2147	0.244	113.93
2146	0.245	113.89
2145	0.246	113.95
2144	0.248	113.95
2143	0.249	113.69
2142	0.250	113.43
2141	0.252	113.32
2140	0.253	112.95
2139	0.254	112.81
2138	0.255	112.96
2137	0.257	112.68
2136	0.258	112.14
2135	0.259	111.26
2134	0.260	111.20
2133	0.262	111.34
2132	0.263	111.48
2131	0.264	111.56
2130	0.265	111.42
2129	0.267	111.18
2128	0.268	111.44
2127	0.269	111.09
2126	0.271	110.86
2125	0.272	111.07
2124	0.273	110.92
2123	0.274	110.49
2122	0.276	110.33
2121	0.277	110.34
2120	0.278	110.19
2119	0.279	109.96
2118	0.281	109.73
2117	0.282	109.55
2116	0.283	109.25
2115	0.285	109.15

ycord (pix)	Depth (in)	Light Intensity
2112	0.288	108.62
2111	0.290	108.30
2110	0.291	107.85
2109	0.292	107.60
2108	0.293	107.64
2107	0.295	107.78
2106	0.296	108.12
2105	0.297	108.58
2104	0.299	108.89
2103	0.300	108.59
2102	0.301	108.52
2101	0.302	108.30
2100	0.304	107.54
2099	0.305	106.45
2098	0.306	106.08
2097	0.307	105.85
2096	0.309	105.37
2095	0.310	104.80
2094	0.311	104.16
2093	0.312	103.93
2092	0.314	103.90
2091	0.315	104.00
2090	0.316	104.09
2089	0.318	104.19
2088	0.319	104.36
2087	0.320	104.59
2086	0.321	105.16
2085	0.323	104.92
2084	0.324	104.18
2083	0.325	103.58
2082	0.326	103.34
2081	0.328	103.21
2080	0.329	102.63
2079	0.330	102.66
2078	0.332	102.42
2077	0.333	101.82
2076	0.334	101.67
2075	0.335	101.98
2074	0.337	102.58
2073	0.338	102.89
2072	0.339	103.04
2071	0.340	103.01
2070	0.342	103.15
2069	0.343	103.18

ycord (pix)	Depth (in)	Light Intensity
2067	0.346	103.20
2066	0.347	103.22
2065	0.348	103.67
2064	0.349	103.65
2063	0.351	102.86
2062	0.352	101.93
2061	0.353	101.45
2060	0.354	101.13
2059	0.356	100.90
2058	0.357	100.34
2057	0.358	99.99
2056	0.359	99.71
2055	0.361	99.75
2054	0.362	100.11
2053	0.363	100.29
2052	0.365	100.11
2051	0.366	99.90
2050	0.367	99.75
2049	0.368	99.29
2048	0.370	98.57
2047	0.371	97.90
2046	0.372	97.87
2045	0.373	98.05
2044	0.375	97.84
2043	0.376	97.58
2042	0.377	97.55
2041	0.379	97.87
2040	0.380	98.25
2039	0.381	98.25
2038	0.382	98.56
2037	0.384	98.91
2036	0.385	99.03
2035	0.386	98.88
2034	0.387	98.47
2033	0.389	97.59
2032	0.390	97.29
2031	0.391	97.26
2030	0.393	97.64
2029	0.394	98.03
2028	0.395	98.03
2027	0.396	97.95
2026	0.398	98.26
2025	0.399	98.34
2024	0.400	98.20
2023	0.401	97.82

ycord (pix)	Depth (in)	Light Intensity
2021	0.404	96.96
2020	0.405	96.23
2019	0.406	95.59
2018	0.408	95.62
2017	0.409	95.90
2016	0.410	95.73
2015	0.412	95.49
2014	0.413	95.24
2013	0.414	95.04
2012	0.415	95.04
2011	0.417	95.20
2010	0.418	95.00
2009	0.419	94.78
2008	0.420	94.73
2007	0.422	94.67
2006	0.423	94.35
2005	0.424	94.24
2004	0.426	94.14
2003	0.427	94.12
2002	0.428	93.95
2001	0.429	93.86
2000	0.431	93.42
1999	0.432	93.26
1998	0.433	93.13
1997	0.434	92.74
1996	0.436	92.44
1995	0.437	92.56
1994	0.438	92.53
1993	0.440	92.54
1992	0.441	92.01
1991	0.442	91.66
1990	0.443	91.43
1989	0.445	91.64
1988	0.446	91.15
1987	0.447	90.95
1986	0.448	91.00
1985	0.450	91.00
1984	0.451	90.85
1983	0.452	90.51
1982	0.453	90.48

Table 9: FD&C Blue Light Transmission Data

ycord (pix)	Depth (in)	Light Intensity
41	0.000	---
42	0.002	---
43	0.003	---
44	0.005	---
45	0.006	---
46	0.008	---
47	0.009	---
48	0.011	---
49	0.012	---
50	0.014	78.62
51	0.015	79.29
52	0.017	79.00
53	0.018	78.38
54	0.020	77.33
55	0.022	76.10
56	0.023	75.48
57	0.025	74.62
58	0.026	74.19
59	0.028	73.19
60	0.029	72.43
61	0.031	71.90
62	0.032	71.24
63	0.034	70.67
64	0.035	70.48
65	0.037	70.14
66	0.038	68.90
67	0.040	68.38
68	0.042	68.29
69	0.043	67.52
70	0.045	66.67
71	0.046	66.19
72	0.048	65.62
73	0.049	65.00
74	0.051	64.76
75	0.052	64.76
76	0.054	64.33
77	0.055	63.86
78	0.057	63.71
79	0.058	63.29
80	0.060	62.67
81	0.062	62.05
82	0.063	61.95
83	0.065	62.10

ycord (pix)	Depth (in)	Light Intensity
87	0.071	61.86
88	0.072	61.62
89	0.074	61.19
90	0.075	61.29
91	0.077	60.90
92	0.078	60.52
93	0.080	60.38
94	0.082	60.38
95	0.083	60.33
96	0.085	60.43
97	0.086	60.10
98	0.088	59.86
99	0.089	59.62
100	0.091	59.62
101	0.092	59.52
102	0.094	59.48
103	0.095	59.38
104	0.097	59.19
105	0.099	58.52
106	0.100	57.86
107	0.102	57.43
108	0.103	57.38
109	0.105	56.90
110	0.106	56.43
111	0.108	56.14
112	0.109	56.14
113	0.111	55.90
114	0.112	55.57
115	0.114	55.48
116	0.115	55.43
117	0.117	54.90
118	0.119	54.48
119	0.120	53.95
120	0.122	53.62
121	0.123	53.52
122	0.125	53.57
123	0.126	53.24
124	0.128	52.67
125	0.129	52.33
126	0.131	52.14
127	0.132	51.62
128	0.134	51.48
129	0.135	51.38

ycord (pix)	Depth (in)	Light Intensity
133	0.142	50.29
134	0.143	50.05
135	0.145	50.14
136	0.146	50.19
137	0.148	49.76
138	0.149	49.57
139	0.151	49.48
140	0.152	49.24
141	0.154	49.10
142	0.155	48.62
143	0.157	48.19
144	0.159	48.05
145	0.160	48.29
146	0.162	47.48
147	0.163	47.14
148	0.165	47.14
149	0.166	46.67
150	0.168	46.52
151	0.169	46.24
152	0.171	46.05
153	0.172	45.90
154	0.174	46.14
155	0.175	45.86
156	0.177	45.48
157	0.179	44.90
158	0.180	45.00
159	0.182	44.71
160	0.183	44.19
161	0.185	44.24
162	0.186	44.29
163	0.188	43.67
164	0.189	42.86
165	0.191	43.05
166	0.192	43.05
167	0.194	43.05
168	0.195	43.10
169	0.197	43.14
170	0.199	42.10
171	0.200	41.33
172	0.202	41.14
173	0.203	41.14
174	0.205	41.24
175	0.206	41.57

ycord (pix)	Depth (in)	Light Intensity
160	0.183	44.19
161	0.185	44.24
162	0.186	44.29
163	0.188	43.67
164	0.189	42.86
165	0.191	43.05
166	0.192	43.05
167	0.194	43.05
168	0.195	43.10
169	0.197	43.14
170	0.199	42.10
171	0.200	41.33
172	0.202	41.14
173	0.203	41.14
174	0.205	41.24
175	0.206	41.57
176	0.208	41.29
177	0.209	40.81
178	0.211	40.29
179	0.212	40.19
180	0.214	40.43
181	0.215	40.24
182	0.217	40.05
183	0.219	40.00
184	0.220	40.10
185	0.222	39.81
186	0.223	39.33
187	0.225	38.62
188	0.226	38.10
189	0.228	37.86
190	0.229	37.95
191	0.231	38.19
192	0.232	38.10
193	0.234	37.24
194	0.235	36.95
195	0.237	36.81
196	0.239	36.62
197	0.240	36.33
198	0.242	36.33
199	0.243	36.24
200	0.245	35.81
201	0.246	35.81
202	0.248	36.10
203	0.249	35.57
204	0.251	34.81
205	0.252	34.95

ycord (pix)	Depth (in)	Light Intensity
206	0.254	35.52
207	0.255	34.76
208	0.257	33.95
209	0.259	33.81
210	0.260	34.14
211	0.262	34.10
212	0.263	33.81
213	0.265	33.48
214	0.266	33.81
215	0.268	33.95
216	0.269	33.95
217	0.271	34.05
218	0.272	33.86
219	0.274	33.33
220	0.276	32.90
221	0.277	32.95
222	0.279	32.71
223	0.280	32.33
224	0.282	31.95
225	0.283	32.14
226	0.285	32.24
227	0.286	31.29
228	0.288	30.57
229	0.289	30.86
230	0.291	31.48
231	0.292	31.29
232	0.294	30.48
233	0.296	30.33
234	0.297	29.76
235	0.299	29.67
236	0.300	29.67
237	0.302	29.24
238	0.303	29.57
239	0.305	29.76
240	0.306	29.19
241	0.308	29.38
242	0.309	30.10
243	0.311	30.33
244	0.312	29.90
245	0.314	29.62
246	0.316	29.14
247	0.317	29.05
248	0.319	29.29
249	0.320	28.86
250	0.322	27.95
251	0.323	27.71

ycord (pix)	Depth (in)	Light Intensity
252	0.325	28.00
253	0.326	28.00
254	0.328	27.71
255	0.329	27.57
256	0.331	27.67
257	0.332	27.81
258	0.334	27.52
259	0.336	27.14
260	0.337	26.95
261	0.339	26.86
262	0.340	26.86
263	0.342	26.33
264	0.343	26.00
265	0.345	26.38
266	0.346	26.05
267	0.348	25.43
268	0.349	25.05
269	0.351	25.29
270	0.352	25.86
271	0.354	26.10
272	0.356	26.05
273	0.357	26.24
274	0.359	26.10
275	0.360	25.76
276	0.362	25.38
277	0.363	25.10
278	0.365	25.43
279	0.366	25.14
280	0.368	24.29
281	0.369	23.86
282	0.371	24.05
283	0.372	23.90
284	0.374	23.57
285	0.376	23.48
286	0.377	23.24
287	0.379	22.95
288	0.380	23.10
289	0.382	23.52
290	0.383	23.67
291	0.385	24.43
292	0.386	25.00
293	0.388	24.48
294	0.389	23.43
295	0.391	22.95
296	0.392	22.67
297	0.394	22.76

ycord (pix)	Depth (in)	Light Intensity
298	0.396	22.29
299	0.397	21.86
300	0.399	21.95
301	0.400	22.05
302	0.402	21.67
303	0.403	21.14
304	0.405	21.29
305	0.406	21.10
306	0.408	20.67
307	0.409	20.86
308	0.411	20.86
309	0.412	20.86
310	0.414	20.71
311	0.416	20.33
312	0.417	19.57
313	0.419	19.52
314	0.420	19.90
315	0.422	20.48
316	0.423	20.43
317	0.425	19.90
318	0.426	19.95
319	0.428	19.86
320	0.429	19.38
321	0.431	19.48
322	0.432	19.90
323	0.434	20.05
324	0.436	19.52
325	0.437	19.24
326	0.439	19.43
327	0.440	19.62
328	0.442	19.86
329	0.443	19.14
330	0.445	18.95
331	0.446	19.24
332	0.448	19.38
333	0.449	19.14
334	0.451	18.95
335	0.453	18.90
336	0.454	18.67
337	0.456	18.38
338	0.457	18.48
339	0.459	18.86
340	0.460	19.10
341	0.462	18.90
342	0.463	19.00

ycord (pix)	Depth (in)	Light Intensity
344	0.466	17.48
345	0.468	17.14
346	0.469	17.14
347	0.471	17.24
348	0.473	17.38
349	0.474	17.48
350	0.476	17.19

A.6.4 CB1 Developed Conduit Experiment

Table 10: CB1 Developed Conduit Experiment: discharge data

Tag #	Timestamp (hh:mm:ss)	Time (sec)	Mass (g)	Volume (m^3)
60	22:39:41	81581	7	7.0E-06
120	22:39:53	81593	17	1.7E-05
180	22:40:05	81605	24	2.4E-05
240	22:40:17	81617	38	3.8E-05
300	22:40:29	81629	31	3.1E-05
360	22:40:41	81641	32	3.2E-05
420	22:40:53	81653	48	4.8E-05
480	22:41:05	81665	42	4.2E-05
540	22:41:17	81677	86	8.6E-05
600	22:41:29	81689	131	1.3E-04
660	22:41:42	81702	186	1.9E-04
720	22:41:54	81714	254	2.5E-04
780	22:42:06	81726	347	3.5E-04
840	22:42:18	81738	487	4.9E-04
900	22:42:30	81750	642	6.4E-04
960	22:42:42	81762	793	7.9E-04
1020	22:42:54	81774	918	9.2E-04
1080	22:43:06	81786	1033	1.0E-03
1140	22:43:18	81798	1144	1.1E-03
1200	22:43:31	81811	1297	1.3E-03
1260	22:43:43	81823	1383	1.4E-03
1320	22:43:55	81835	1496	1.5E-03
1380	22:44:07	81847	1553	1.6E-03
1440	22:44:19	81859	1634	1.6E-03
1500	22:44:31	81871	1718	1.7E-03
1560	22:44:43	81883	1808	1.8E-03
1620	22:44:55	81895	1884	1.9E-03
1680	22:45:07	81907	1998	2.0E-03
1740	22:45:19	81919	2047	2.1E-03
1800	22:45:31	81931	2073	2.1E-03
1860	22:45:43	81943	2145	2.1E-03
1920	22:45:55	81955	2258	2.3E-03
1980	22:46:07	81967	2406	2.4E-03
2040	22:46:19	81979	2479	2.5E-03
2100	22:46:32	81992	2593	2.6E-03
2160	22:46:44	82004	2665	2.7E-03
2220	22:46:56	82016	2744	2.7E-03
2280	22:47:08	82028	2834	2.8E-03
2340	22:47:20	82040	2948	3.0E-03
2400	22:47:32	82052	3013	3.0E-03
2460	22:47:44	82064	3112	3.1E-03
2520	22:47:56	82076	3227	3.2E-03
2580	22:48:08	82088	3295	3.3E-03
2640	22:48:21	82101	3410	3.4E-03
2700	22:48:33	82113	3459	3.5E-03
2760	22:48:45	82125	3523	3.5E-03
2820	22:48:57	82137	3606	3.6E-03
2880	22:49:09	82149	3709	3.7E-03
2940	22:49:21	82161	3796	3.8E-03
3000	22:49:33	82173	3848	3.9E-03
3060	22:49:45	82185	4117	4.1E-03
3120	22:49:57	82197	#N/A	#N/A
3180	22:50:09	82209	4299	4.3E-03
3240	22:50:21	82221	4383	4.4E-03
3300	22:50:33	82233	4479	4.5E-03
3360	22:50:45	82245	4607	4.6E-03
3420	22:50:58	82258	4765	4.8E-03
3480	22:51:10	82270	4923	4.9E-03
3540	22:51:22	82282	5071	5.1E-03
3600	22:51:34	82294	5231	5.2E-03
3660	22:51:46	82306	5379	5.4E-03
3720	22:51:58	82318	5539	5.5E-03
3780	22:52:10	82330	5711	5.7E-03
3840	22:52:22	82342	5859	5.9E-03
3900	22:52:35	82355	6011	6.0E-03
3960	22:52:47	82367	6174	6.2E-03
4020	22:52:59	82379	6284	6.3E-03
4080	22:53:11	82391	6375	6.4E-03
4140	22:53:23	82403	6458	6.5E-03
4200	22:53:35	82415	6552	6.6E-03
4260	22:53:47	82427	6649	6.7E-03
4320	22:53:59	82439	6742	6.8E-03
4380	22:54:11	82451	6836	6.8E-03
4440	22:54:23	82463	6928	6.9E-03
4500	22:54:35	82475	7017	7.0E-03
4560	22:54:49	82489	7117	7.1E-03
4620	22:55:02	82502	7206	7.2E-03
4680	22:55:17	82517	7308	7.3E-03
4740	22:55:29	82529	7376	7.4E-03
4800	22:55:41	82541	7459	7.5E-03
4860	22:55:53	82553	7665	7.7E-03
4920	22:56:05	82565	7748	7.8E-03
4980	22:56:17	82577	7847	7.9E-03
5040	22:56:29	82589	7948	8.0E-03
5100	22:56:41	82601	8046	8.1E-03
5160	22:56:53	82613	8097	8.1E-03
5220	22:57:06	82626	8195	8.2E-03
5280	22:57:18	82638	8291	8.3E-03
5340	22:57:30	82650	8383	8.4E-03
5400	22:57:42	82662	#N/A	#N/A
5460	22:57:54	82674	#N/A	#N/A
5520	22:58:06	82686	#N/A	#N/A
5580	22:58:18	82698	#N/A	#N/A
5640	22:58:30	82710	#N/A	#N/A
5700	22:58:42	82722	#N/A	#N/A
5760	22:58:54	82734	#N/A	#N/A
5820	22:59:06	82746	#N/A	#N/A
5880	22:59:19	82759	1663	1.7E-03

Tag #	Timestamp (h:m:s)	Time (sec)	Mass (g)	Volume (m^3)
5880	22:59:19	82759	1663	1.7E-03
5940	22:59:31	82771	1807	1.8E-03
6000	22:59:43	82783	2074	2.1E-03
6060	22:59:56	82796	2213	2.2E-03
6120	23:00:08	82808	#N/A	#N/A
6180	23:00:20	82820	2499	2.5E-03
6240	23:00:32	82832	2654	2.7E-03
6300	23:00:45	82845	2815	2.8E-03
6360	23:00:57	82857	2971	3.0E-03
6420	23:01:09	82869	3122	3.1E-03
6480	23:01:21	82881	3278	3.3E-03
6540	23:01:33	82893	3444	3.5E-03
6600	23:01:45	82905	3566	3.6E-03
6660	23:01:58	82918	3714	3.7E-03
6720	23:02:10	82930	3862	3.9E-03
6780	23:02:22	82942	4020	4.0E-03
6840	23:02:34	82954	4185	4.2E-03
6900	23:02:47	82967	4345	4.4E-03
6960	23:02:59	82979	4493	4.5E-03
7020	23:03:11	82991	4641	4.6E-03
7080	23:03:24	83004	4795	4.8E-03
7140	23:03:36	83016	4954	5.0E-03
7200	23:03:48	83028	5124	5.1E-03
7260	23:04:00	83040	5290	5.3E-03
7320	23:04:12	83052	5497	5.5E-03
7380	23:04:25	83065	5638	5.6E-03
7440	23:04:37	83077	5805	5.8E-03
7500	23:04:49	83089	5989	6.0E-03
7560	23:05:01	83101	6186	6.2E-03
7620	23:05:14	83114	6344	6.4E-03
7680	23:05:26	83126	6518	6.5E-03
7740	23:05:38	83138	6751	6.8E-03
7800	23:05:50	83150	6892	6.9E-03
7860	23:06:03	83163	7026	7.0E-03
7920	23:06:15	83175	7166	7.2E-03
7980	23:06:27	83187	7300	7.3E-03
8040	23:06:39	83199	7439	7.5E-03
8100	23:06:51	83211	7521	7.5E-03
8160	23:07:03	83223	7547	7.6E-03
8220	23:07:15	83235	7563	7.6E-03
8280	23:07:27	83247	7620	7.6E-03
8340	23:07:39	83259	7648	7.7E-03
8400	23:07:51	83271	7655	7.7E-03

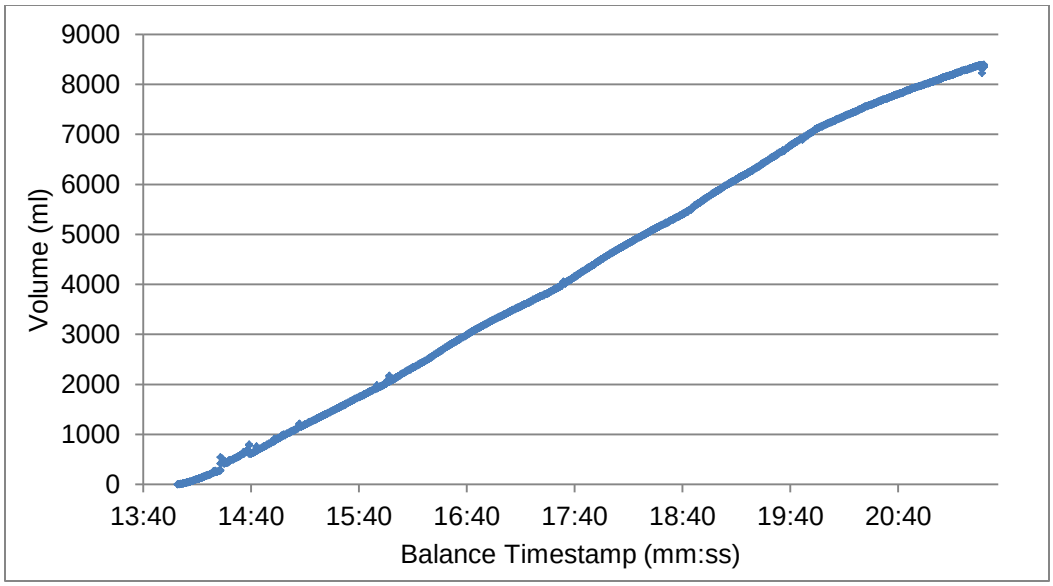

Figure 25: CB1 Developed Conduit Experiment: discharge vs. time

Table 11: CB1 Developed Conduit Experiment: water pressured data

Img #	Time (min)	Water Pressure (Pa)	Effective Pressure (Pa)	Height above Bed (in)
235	12.4	336	-145	1.4
239	12.6	563	82	2.3
243	12.9	678	197	2.7
247	13.1	541	60	2.2
251	13.4	362	-119	1.5
255	13.6	338	-143	1.4
259	13.8	484	3	1.9
263	14.1	632	151	2.5
267	14.3	693	212	2.8
271	14.6	682	201	2.7
275	14.9	724	243	2.9
279	15.2	777	297	3.1
283	15.5	768	287	3.1
287	15.8	757	276	3.0
291	16.1	506	25	2.0
295	16.4	610	129	2.5

A.6.5 CB1 Sliding Experiment

Table 12: CB1 Sliding Experimental: time, water pressure, storage, and velocity data

Img #	Time stamp (mm:ss)	Time step (sec)	Water Height (pix)	Water Height (in)	Height above Bed (in)	Storage (ml)	ds/dt (ml/sec)	Surface Velocity (m/sec)	Sliding Velocity (m/sec)
852	23:57.7		686.5	6.88	1.13	36.1			
853	24:05.9	8.2				30.8	-0.65	5.14E-05	6.57E-06
854	24:14.1	8.2	670.5	6.72	0.97	25.1	-0.68		
855	24:22.0	7.9				23.8	-0.17	5.09E-05	7.81E-06
856	24:29.9	7.9	622.5	6.24	0.49	23.2	-0.07		
857	24:37.5	7.6				23.1	-0.01	5.51E-05	6.43E-06
858	24:45.2	7.7	560.5	5.62	-0.13	23.3	0.02		
859	24:53.1	7.9				29.1	0.74	5.66E-05	1.83E-05
860	25:01.0	7.9	781.5	7.84	2.09	35.9	0.86		
861	25:09.0	8.0				35.4	-0.06	6.44E-05	2.12E-05
862	25:17.0	8.0	751.5	7.54	1.79	41.0	0.70		
863	25:24.9	7.9				37.3	-0.46	6.06E-05	2.01E-05
864	25:32.8	7.9	743.5	7.46	1.71	38.6	0.16		
865	25:40.7	7.9				40.3	0.21	5.43E-05	8.72E-06
866	25:48.5	7.8	747.5	7.50	1.75	39.2	-0.13		
867	25:56.3	7.8				35.8	-0.44	5.15E-05	9.06E-06
868	26:04.1	7.8	699.5	7.02	1.27	23.9	-1.52		
869	26:12.3	8.2				22.1	-0.22	5.11E-05	1.14E-05
870	26:20.4	8.2	669.5	6.71	0.96	22.7	0.07		
871	26:28.3	7.8				21.1	-0.21	4.94E-05	6.00E-06
872	26:36.1	7.8	604.5	6.06	0.31	19.3	-0.23		
873	26:44.0	8.0				19.7	0.05	4.50E-05	1.32E-06
874	26:52.0	7.9	724.5	7.27	1.52	19.7	0.00		
875	26:59.5	7.4				25.1	0.74	6.05E-05	2.53E-05
876	27:06.9	7.4	774.5	7.77	2.02	30.9	0.78		
877	27:15.1	8.3				32.7	0.21	6.02E-05	2.36E-05
878	27:23.4	8.3	741.5	7.44	1.69	33.5	0.09		
879	27:31.1	7.7				33.9	0.05	5.54E-05	8.85E-06
880	27:38.7	7.7	743.5	7.46	1.71	35.9	0.27		
881	27:46.3	7.6				32.9	-0.39	5.80E-05	2.21E-05
882	27:53.9	7.6	748.5	7.51	1.76	38.4	0.73		
883	28:02.3	8.4				27.9	-1.26	4.97E-05	7.69E-06
884	28:10.7	8.4	698.5	7.01	1.26	22.4	-0.64		
885	28:18.5	7.8				21.8	-0.09	4.79E-05	6.07E-06
886	28:26.3	7.8	658.5	6.60	0.85	20.8	-0.12		
887	28:34.5	8.2				18.8	-0.24	4.54E-05	6.50E-06
888	28:42.7	8.2	563.5	5.65	-0.10	19.9	0.13		

Img #	Time stamp (mm:ss)	Time step (sec)	Water Height (pix)	Water Height (in)	Height above Bed (in)	Storage (ml)	dS/dt (ml/sec)	Surface Velocity (m/sec)	Sliding Velocity (m/sec)
891	29:06.8	8.2				29.2	0.02	6.14E-05	1.40E-05
892	29:15.0	8.2	751.5	7.54	1.79	44.2	1.84		
893	29:23.3	8.3				34.3	-1.20	5.17E-05	1.75E-05
894	29:31.6	8.3	736.5	7.39	1.64	40.5	0.76		
895	29:39.8	8.2				44.2	0.44	5.54E-05	1.86E-05
896	29:48.0	8.2	745.5	7.48	1.73	42.4	-0.21		
897	29:55.9	7.9				49.3	0.86	5.26E-05	2.10E-05
898	30:03.8	7.9	704.5	7.07	1.32	27.8	-2.72		
899	30:11.8	8.0				24.4	-0.43	4.89E-05	4.17E-07
900	30:19.7	8.0	697.5	7.00	1.25	22.5	-0.24		
901	30:27.7	8.0				19.2	-0.42	4.46E-05	7.57E-06
902	30:35.8	8.1	640.5	6.42	0.67	19.2	0.00	4.85E-05	6.45E-06
903	30:43.5	7.7				18.9	-0.03		
904	30:51.7	8.2	563.5	5.65	-0.10	17.7	-0.15	5.28E-05	1.11E-05
905	31:00.0	8.3				17.6	-0.01		
906	31:08.5	8.5	494.5	4.96	-0.79	18.0	0.05	4.71E-05	4.48E-06
907	31:16.4	8.0				18.4	0.05		
908	31:25.2	8.8	475.5	4.77	-0.98	20.2	0.21	5.06E-05	4.42E-06
909	31:33.3	8.1				15.3	-0.60		
910	31:41.1	7.7	467.5	4.69	-1.06	15.4	0.00	4.77E-05	6.43E-06
911	31:48.9	7.9				15.5	0.01		
912	31:57.0	8.0	1070.5	10.74	4.99	21.5	0.74	6.69E-05	8.76E-06
913	32:05.0	8.0				38.3	2.10		
914	32:12.9	7.9	767.5	7.70	1.95	30.6	-0.97	5.77E-05	3.37E-05
915	32:20.6	7.7				33.4	0.36		
916	32:28.7	8.1	734.5	7.37	1.62	40.1	0.83	6.53E-05	2.52E-05
917	32:36.6	7.9				35.5	-0.59		
918	32:44.3	7.7	740.5	7.43	1.68	34.2	-0.16	5.76E-05	1.55E-05
919	32:52.9	8.6				36.4	0.26		
920	33:00.6	7.7	746.5	7.49	1.74	33.8	-0.34	6.27E-05	2.12E-05
921	33:08.9	8.3				32.6	-0.15		
922	33:16.5	7.6	745.5	7.48	1.73	38.1	0.72	5.68E-05	1.25E-05
923	33:28.2	11.7				35.2	-0.25		
924	33:41.9	13.7	743.5	7.46	1.71	38.9	0.27	5.95E-05	1.45E-05
925	33:53.6	11.7				39.0	0.01		
926	34:03.6	10.0	695.5	6.98	1.23	34.6	-0.44	5.71E-05	1.58E-05
927	34:17.7	14.1				22.8	-0.84		
928	34:31.2	13.6	681.5	6.83	1.08	21.6	-0.09	4.52E-05	1.49E-06
929	34:43.0	11.8				19.5	-0.18		

Img #	Time stamp (mm:ss)	Time step (sec)	Water Height (pix)	Water Height (in)	Height above Bed (in)	Storage (ml)	dS/dt (ml/sec)	Surface Velocity (m/sec)	Sliding Velocity (m/sec)
931	35:07.9	13.8				19.1	0.09		
932	35:20.2	12.3	502.5	5.04	-0.71	16.7	-0.19	4.99E-05	6.95E-06
933	35:32.8	12.6				16.4	-0.02		
934	35:44.9	12.1	468.5	4.70	-1.05	16.3	-0.01	5.24E-05	1.27E-05
935	35:55.5	10.6				27.8	1.08		
936	36:07.9	12.4	767.5	7.70	1.95	32.2	0.35	6.24E-05	2.11E-05
937	36:20.2	12.4				36.1	0.32		
938	36:34.9	14.7	729.5	7.32	1.57	36.5	0.02	6.28E-05	1.76E-05
939	36:47.3	12.5				33.7	-0.23		
940	37:15.9	28.5	743.5	7.46	1.71	37.3	0.13	5.18E-05	1.35E-05
941	37:31.0	15.1				37.0	-0.02		
942	37:45.0	14.0	738.5	7.41	1.66	41.1	0.29	5.39E-05	1.69E-05
943	37:59.2	14.2				27.9	-0.93		
944	38:13.5	14.3	674.5	6.76	1.01	21.9	-0.42	4.96E-05	8.30E-06
945	38:26.1	12.6				20.7	-0.09		
946	38:38.0	11.9	583.5	5.85	0.10	20.2	-0.05	4.41E-05	9.33E-06
947	38:52.0	14.0				17.4	-0.20		
948	39:06.6	14.6	492.5	4.94	-0.81	16.8	-0.04	4.38E-05	5.36E-06
949	39:20.5	13.9				16.3	-0.03		
950	39:32.2	11.7	467.5	4.69	-1.06	17.5	0.10	4.65E-05	8.14E-06
951	39:45.5	13.3				15.5	-0.15		
952	39:58.2	12.7	1068.5	10.72	4.97	22.4	0.54	5.55E-05	1.88E-05
953	40:13.3	15.1				31.6	0.61		
954	40:27.8	14.5	730.5	7.33	1.58	38.6	0.48	5.81E-05	2.11E-05
955	40:42.7	14.9				37.7	-0.06		
956	40:54.8	12.1	730.5	7.33	1.58	38.1	0.03	5.57E-05	1.67E-05
957	41:08.9	14.2				35.2	-0.20		
958	41:24.0	15.1	737.5	7.40	1.65	40.2	0.33	5.15E-05	1.65E-05
959	41:39.8	15.9				39.4	-0.05		
960	41:52.1	12.2	730.5	7.33	1.58	40.3	0.07	5.00E-05	1.42E-05
961	42:05.3	13.2				26.4	-1.05		
962	42:19.0	13.7	664.5	6.66	0.91	22.8	-0.26	4.28E-05	8.21E-06
963	42:33.4	14.4				20.6	-0.15		
964	42:46.2	12.8	533.5	5.35	-0.40	19.3	-0.10	4.07E-05	6.50E-06
965	43:01.1	14.9				18.4	-0.06		
966	43:15.2	14.1	497.5	4.99	-0.76	18.4	-0.01	4.99E-05	1.35E-05
967	43:29.8	14.6				16.2	-0.15		
968	43:42.7	12.9	466.5	4.68	-1.07	17.1	0.07	4.03E-05	7.40E-06
969	43:57.5	14.8				15.7	-0.09		
970	44:10.6	13.1	461.5	4.63	-1.12	14.9	-0.07	4.10E-05	6.88E-06

Img #	Time stamp (mm:ss)	Time step (sec)	Water Height (pix)	Water Height (in)	Height above Bed (in)	Storage (ml)	dS/dt (ml/sec)	Surface Velocity (m/sec)	Sliding Velocity (m/sec)
971	44:25.2	14.6				14.6	-0.02		
972	44:39.2	14.0	459.5	4.61	-1.14	14.6	0.01	4.55E-05	1.12E-05
973	44:51.3	12.1				15.2	0.04		
974	45:02.7	11.4	987.5	9.90	4.15	26.2	0.97	5.39E-05	1.50E-05
975	45:17.0	14.3				32.6	0.44		
976	45:29.9	12.9	729.5	7.32	1.57	41.4	0.68	4.99E-05	1.47E-05
977	45:44.6	14.7				41.9	0.04		
978	45:59.6	15.0	731.5	7.34	1.59	42.7	0.05	5.62E-05	2.10E-05
979	46:13.2	13.6				44.4	0.12		
980	46:28.6	15.4	733.5	7.36	1.61	44.4	0.00	5.28E-05	2.02E-05
981	46:47.5	18.9				46.9	0.13		
982	46:59.1	11.7	732.5	7.35	1.60	45.7	-0.11	4.97E-05	1.39E-05
983	47:13.1	13.9				44.4	-0.09		
984	47:23.4	10.3	726.5	7.29	1.54	43.7	-0.07	5.04E-05	1.55E-05
985	47:38.6	15.2				47.9	0.27		
986	47:50.1	11.5	724.5	7.27	1.52	37.9	-0.88	4.79E-05	1.66E-05
987	48:10.1	20.0				29.2	-0.43		
988	48:29.8	19.7	654.5	6.56	0.81	22.2	-0.36	4.03E-05	2.98E-06
989	48:39.6	9.8				22.8	0.06		
990	48:50.5	10.8	577.5	5.79	0.04	19.7	-0.28	4.01E-05	9.50E-06
991	49:02.9	12.5				19.1	-0.05		
992	49:13.2	10.3	499.5	5.01	-0.74	18.5	-0.05	4.46E-05	1.53E-05
993	49:21.9	8.7				18.1	-0.05		
994	49:30.3	8.4	475.5	4.77	-0.98	17.7	-0.04	3.57E-05	1.02E-06
995	49:39.6	9.3				17.3	-0.05		
996	49:48.5	8.9	468.5	4.70	-1.05	17.2	0.00	4.40E-05	1.24E-05
997	49:57.5	9.0				17.9	0.07		
998	50:06.3	8.8	472.5	4.74	-1.01	17.6	-0.03	3.83E-05	3.49E-06
999	50:15.0	8.7				17.4	-0.02		
1000	50:23.6	8.6	474.5	4.76	-0.99	18.0	0.07		

Table 13: CB1 Sliding Experiment: discharge data

Tag #	Timestamp (hh:mm:ss)	Time (sec)	Mass (g)	Volume (m^3)
1	14:57:35	53855	1	1.0E-06
2	14:57:53	53873	0	0.0E+00
3	14:58:11	53891	67	6.7E-05
4	14:58:29	53909	94	9.4E-05
5	14:58:47	53927	229	2.3E-04
6	14:59:05	53945	0	0.0E+00
7	14:59:23	53963	0	0.0E+00
8	14:59:41	53981	662	6.6E-04
9	14:59:59	53999	0	0.0E+00
10	15:00:17	54017	923	9.2E-04
11	15:00:35	54035	1044	1.0E-03
12	15:00:52	54052	1165	1.2E-03
13	15:01:11	54071	1288	1.3E-03
14	15:01:28	54088	1408	1.4E-03
15	15:01:46	54106	1528	1.5E-03
16	15:02:04	54124	1642	1.6E-03
17	15:02:22	54142	1754	1.8E-03
18	15:02:40	54160	1865	1.9E-03
19	15:02:58	54178	1973	2.0E-03
20	15:03:16	54196	2078	2.1E-03
21	15:03:34	54214	2185	2.2E-03
22	15:03:52	54232	2296	2.3E-03
23	15:04:10	54250	2406	2.4E-03
24	15:04:28	54268	2516	2.5E-03
25	15:04:46	54286	2648	2.6E-03
26	15:05:04	54304	2755	2.8E-03
27	15:05:22	54322	2867	2.9E-03
28	15:05:40	54340	2971	3.0E-03
29	15:05:58	54358	3071	3.1E-03
30	15:06:16	54376	3172	3.2E-03
31	15:06:34	54394	3273	3.3E-03
32	15:06:52	54412	3372	3.4E-03
33	15:07:10	54430	3468	3.5E-03
34	15:07:28	54448	3565	3.6E-03
35	15:07:46	54466		#VALUE
36	15:08:04	54484	3756	3.8E-03
37	15:08:22	54502	3854	3.9E-03
38	15:08:40	54520	3979	4.0E-03
39	15:08:58	54538	4069	4.1E-03
40	15:09:16	54556	4173	4.2E-03
41	15:09:34	54574	4271	4.3E-03
42	15:09:52	54592	4372	4.4E-03
43	15:10:10	54610		#VALUE
44	15:10:28	54628	4565	4.6E-03
45	15:10:46	54646	4657	4.7E-03
46	15:11:04	54664	4755	4.8E-03
47	15:11:22	54682	4852	4.9E-03
48	15:11:40	54700	4946	4.9E-03
49	15:11:58	54718		#VALUE
50	15:12:16	54736	5137	5.1E-03
51	15:12:34	54754	5271	5.3E-03
52	15:12:52	54772	5369	5.4E-03
53	15:13:10	54790	5468	5.5E-03
54	15:13:28	54808	5563	5.6E-03
55	15:13:46	54826	5648	5.6E-03
56	15:14:04	54844	5671	5.7E-03
57	15:14:22	54862	5677	5.7E-03
58	15:14:40	54880	5679	5.7E-03
59	15:14:58	54898	5759	5.8E-03
60	15:15:16	54916	5856	5.9E-03
61	15:15:34	54934	5965	6.0E-03
62	15:15:52	54952		#VALUE
63	15:16:10	54970	6064	6.1E-03
64	15:16:28	54988		#VALUE
65	15:16:46	55006	6075	6.1E-03
66	15:17:04	55024	6159	6.2E-03
67	15:17:22	55042	6271	6.3E-03
68	15:17:40	55060	6382	6.4E-03
69	15:17:58	55078	6438	6.4E-03
70	15:18:16	55096	6449	6.4E-03
71	15:18:34	55114	6459	6.5E-03
72	15:18:52	55132	6494	6.5E-03
73	15:19:10	55150	6599	6.6E-03
74	15:19:28	55168	6714	6.7E-03
75	15:19:46	55186	6819	6.8E-03
76	15:20:04	55204	6856	6.9E-03
77	15:20:22	55222		#VALUE
78	15:20:40	55240	6862	6.9E-03
79	15:20:58	55258	6863	6.9E-03
80	15:21:16	55276	6862	6.9E-03
81	15:21:34	55294	6863	6.9E-03
82	15:21:52	55312	6879	6.9E-03
83	15:22:10	55330	6946	6.9E-03
84	15:22:28	55348	7038	7.0E-03
85	15:22:46	55366	7122	7.1E-03
86	15:23:04	55384	7209	7.2E-03
87	15:23:22	55402	7288	7.3E-03
88	15:23:40	55420		#VALUE
89	15:23:58	55438	7426	7.4E-03
90	15:24:16	55456	7435	7.4E-03
91	15:24:34	55474	7437	7.4E-03
92	15:24:52	55492	7438	7.4E-03
93	15:25:10	55510	7439	7.4E-03
94	15:25:28	55528	7439	7.4E-03
95	15:25:46	55546	7447	7.4E-03
96	15:26:04	55564	7556	7.6E-03
97	15:26:22	55582	7667	7.7E-03
98	15:26:40	55600	7789	7.8E-03
99	15:26:58	55618	7901	7.9E-03
100	15:27:16	55636	7997	8.0E-03

Tag #	Timestamp (hh:mm:ss)	Time (sec)	Mass (g)	Volume (m^3)
101	15:27:34	55654	8115	8.1E-03
102	15:27:52	55672	8186	8.2E-03
103	15:28:10	55690	8195	8.2E-03
104	15:28:28	55708	8198	8.2E-03
105	15:28:46	55726	8199	8.2E-03
106	15:29:04	55744		#VALUE
107	15:29:22	55762		#VALUE
108	15:29:40	55780		#VALUE
109	15:29:58	55798	8243	8.2E-03
110	15:30:16	55816		#VALUE
111	15:30:34	55834	8427	8.4E-03
112	15:30:52	55852	H	#VALUE
113	15:31:10	55870	3696	3.7E-03
114	15:31:28	55888	2511	2.5E-03
115	15:31:46	55906		#VALUE
116	15:32:04	55924	2657	2.7E-03
117	15:32:22	55942	2667	2.7E-03
118	15:32:40	55960	2674	2.7E-03
119	15:32:58	55978	2678	2.7E-03
120	15:33:16	55996	2681	2.7E-03
121	15:33:34	56014	2684	2.7E-03
122	15:33:52	56032	2686	2.7E-03
123	15:34:10	56050	2695	2.7E-03
124	15:34:28	56068	2699	2.7E-03
125	15:34:46	56086	2704	2.7E-03
126	15:35:04	56104	2802	2.8E-03
127	15:35:22	56122	2938	2.9E-03
128	15:35:40	56140	3091	3.1E-03
129	15:35:58	56158	3227	3.2E-03
130	15:36:16	56176	3358	3.4E-03
131	15:36:34	56194	3486	3.5E-03
132	15:36:52	56212	3612	3.6E-03
133	15:37:10	56230	3721	3.7E-03
134	15:37:28	56248	3847	3.8E-03
135	15:37:46	56266	3962	4.0E-03
136	15:38:04	56284	4011	4.0E-03
137	15:38:22	56302	4021	4.0E-03
138	15:38:40	56320	4025	4.0E-03
139	15:38:58	56338	4025	4.0E-03
140	15:39:16	56356	4025	4.0E-03
141	15:39:34	56374	4025	4.0E-03
142	15:39:52	56392	4026	4.0E-03
143	15:40:10	56410	4026	4.0E-03
144	15:40:28	56428	4044	4.0E-03
145	15:40:46	56446	4047	4.0E-03
146	15:41:04	56464	4047	4.0E-03
147	15:41:22	56482	4048	4.0E-03
148	15:41:40	56500	4048	4.0E-03
149	15:41:58	56518	4109	4.1E-03
150	15:42:16	56536	4208	4.2E-03
151	15:42:34	56554	4347	4.3E-03
152	15:42:52	56572	4482	4.5E-03
153	15:43:10	56590	4605	4.6E-03

Tag #	Timestamp (hh:mm:ss)	Time (sec)	Mass (g)	Volume (m^3)
154	15:43:28	56608	4729	4.7E-03
155	15:43:45	56625		#VALUE
156	15:44:03	56643		#VALUE
157	15:44:21	56661	5088	5.1E-03
158	15:44:39	56679	5202	5.2E-03
159	15:44:57	56697	5319	5.3E-03
160	15:45:15	56715	5429	5.4E-03
161	15:45:33	56733	5539	5.5E-03
162	15:45:51	56751		#VALUE
163	15:46:09	56769	5752	5.8E-03
164	15:46:27	56787	5784	5.8E-03
165	15:46:45	56805	5798	5.8E-03
166	15:47:03	56823	5805	5.8E-03
167	15:47:21	56841	5809	5.8E-03
168	15:47:39	56859	5812	5.8E-03
169	15:47:57	56877	5815	5.8E-03
170	15:48:15	56895	5816	5.8E-03
171	15:48:33	56913	5818	5.8E-03
172	15:48:51	56931	5819	5.8E-03
173	15:49:09	56949		#VALUE
174	15:49:27	56967	5821	5.8E-03
175	15:49:45	56985	5823	5.8E-03
176	15:50:03	57003	5843	5.8E-03
177	15:50:21	57021	5844	5.8E-03

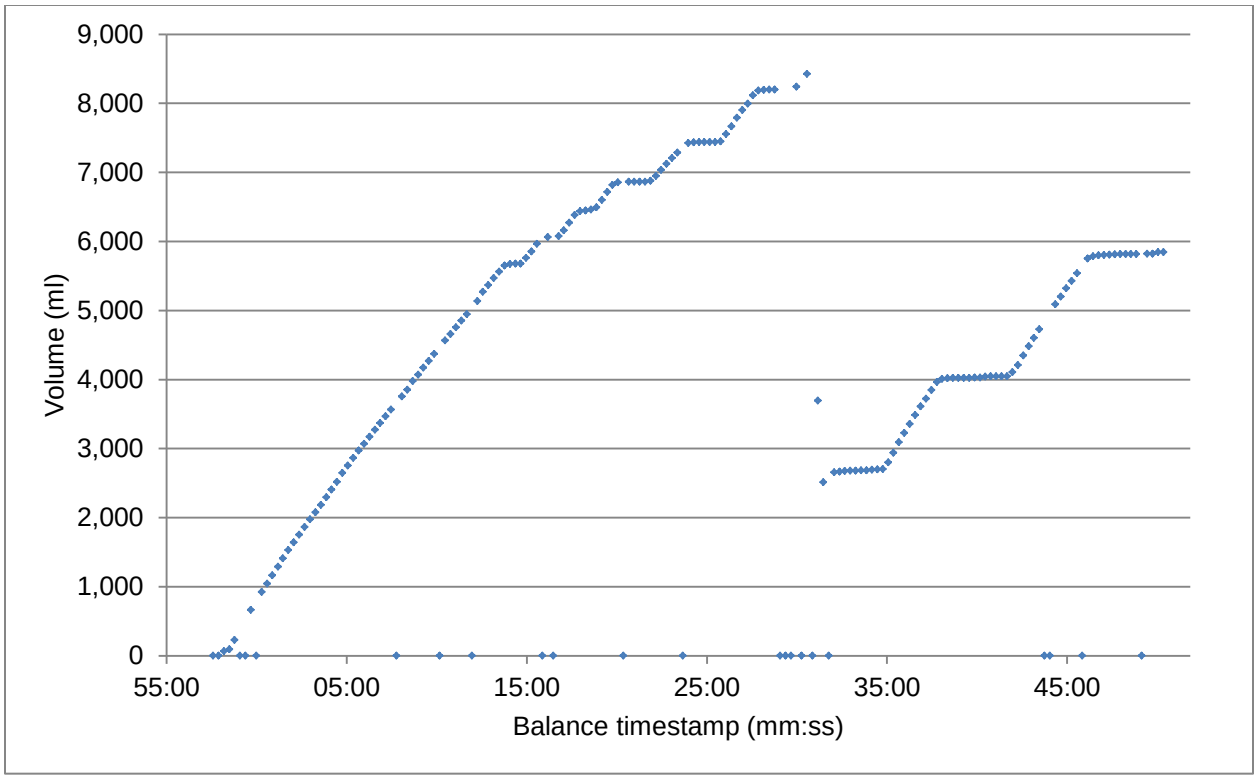

Figure 26: CB1 Sliding Experiment: discharge vs. time

A.6.6 CB2 Sliding Experiment

Table 14: CB2 Sliding Experiment: discharge data

Tag #	Time (h:m:s)	Time (sec)	Mass (g)	Volume (m^3)
90	22:39:47	81587	#N/A	#N/A
180	22:40:05	81605	24	2.4E-05
270	22:40:23	81623	27	2.7E-05
360	22:40:41	81641	32	3.2E-05
450	22:40:59	81659	45	4.5E-05
540	22:41:17	81677	86	8.6E-05
630	22:41:36	81696	167	1.7E-04
720	22:41:54	81714	254	2.5E-04
810	22:42:12	81732	414	4.1E-04
900	22:42:30	81750	642	6.4E-04
990	22:42:48	81768	859	8.6E-04
1080	22:43:06	81786	1033	1.0E-03
1170	22:43:24	81804	1198	1.2E-03
1260	22:43:43	81823	1383	1.4E-03
1350	22:44:01	81841	1523	1.5E-03
1440	22:44:19	81859	1634	1.6E-03
1530	22:44:37	81877	1763	1.8E-03
1620	22:44:55	81895	1884	1.9E-03
1710	22:45:13	81913	2026	2.0E-03
1800	22:45:31	81931	2073	2.1E-03
1890	22:45:49	81949	2195	2.2E-03
1980	22:46:07	81967	2406	2.4E-03
2070	22:46:25	81985	2524	2.5E-03
2160	22:46:44	82004	2665	2.7E-03
2250	22:47:02	82022	2796	2.8E-03
2340	22:47:20	82040	2948	3.0E-03
2430	22:47:38	82058	3048	3.1E-03
2520	22:47:56	82076	3227	3.2E-03
2610	22:48:14	82094	3385	3.4E-03
2700	22:48:33	82113	3459	3.5E-03
2790	22:48:51	82131	3559	3.6E-03
2880	22:49:09	82149	3709	3.7E-03
2970	22:49:27	82167	3821	3.8E-03
3060	22:49:45	82185	4117	4.1E-03
3150	22:50:03	82203	4285	4.3E-03
3240	22:50:21	82221	4383	4.4E-03
3330	22:50:39	82239	4549	4.6E-03
3420	22:50:58	82258	4765	4.8E-03
3510	22:51:16	82276	4989	5.0E-03
3600	22:51:34	82294	5231	5.2E-03
3690	22:51:52	82312	5457	5.5E-03
3780	22:52:10	82330	5711	5.7E-03
3870	22:52:29	82349	5936	5.9E-03
3960	22:52:47	82367	6174	6.2E-03
4050	22:53:05	82385	6329	6.3E-03
4140	22:53:23	82403	6458	6.5E-03
4230	22:53:41	82421	6599	6.6E-03
4320	22:53:59	82439	6742	6.8E-03
4410	22:54:17	82457	6878	6.9E-03
4500	22:54:35	82475	7017	7.0E-03
4590	22:54:56	82496	7165	7.2E-03
4680	22:55:17	82517	7308	7.3E-03
4770	22:55:35	82535	7413	7.4E-03
4860	22:55:53	82553	7665	7.7E-03
4950	22:56:11	82571	7791	7.8E-03
5040	22:56:29	82589	7948	8.0E-03
5130	22:56:47	82607	8062	8.1E-03
5220	22:57:06	82626	8195	8.2E-03
5310	22:57:24	82644	8338	8.4E-03
5400	22:57:42	82662	#N/A	#N/A
5490	22:58:00	82680	#N/A	#N/A
5580	22:58:18	82698	#N/A	#N/A
5670	22:58:36	82716	#N/A	#N/A
5760	22:58:54	82734	#N/A	#N/A
5850	22:59:13	82753	#N/A	#N/A
5940	22:59:31	82771	1807	1.8E-03
6030	22:59:50	82790	2149	2.2E-03
6120	23:00:08	82808	#N/A	#N/A
6210	23:00:26	82826	2579	2.6E-03
6300	23:00:45	82845	2815	2.8E-03
6390	23:01:03	82863	3048	3.1E-03
6480	23:01:21	82881	3278	3.3E-03
6570	23:01:39	82899	3510	3.5E-03
6660	23:01:58	82918	3714	3.7E-03
6750	23:02:16	82936	3944	4.0E-03
6840	23:02:34	82954	4185	4.2E-03
6930	23:02:53	82973	4420	4.4E-03
7020	23:03:11	82991	4641	4.6E-03
7110	23:03:30	83010	4873	4.9E-03
7200	23:03:48	83028	5124	5.1E-03

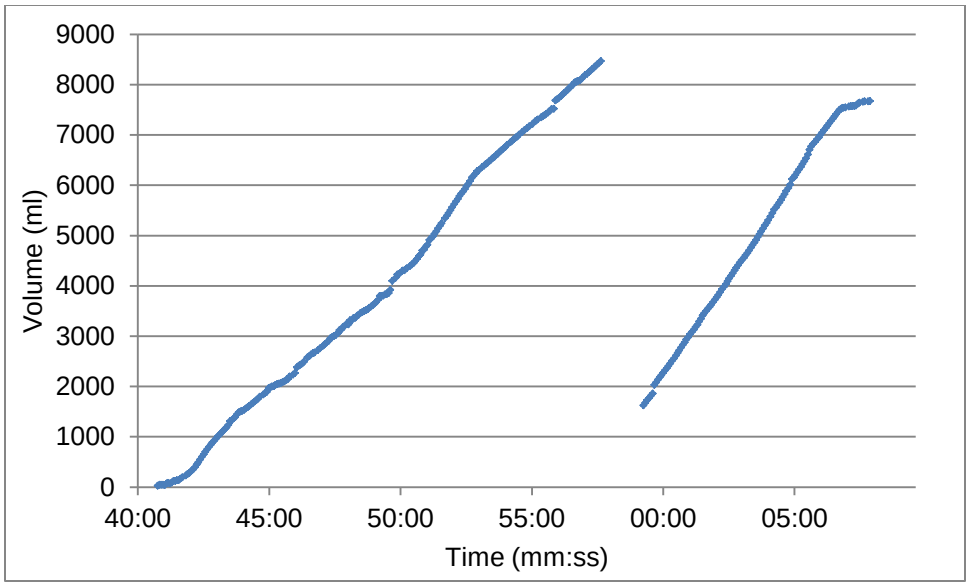

Figure 27: CB2 Sliding Experiment: discharge vs. time

Table 15: CB2 Sliding Experiment: water pressure data

Img #	Timestamp (mm:ss)	Tstep (sec)	Height above bed (in)
128	26:28	0.0	2.67
146	27:28	30.0	2.69
164	28:28	30.0	2.43
182	29:28	30.0	2.69
200	30:31	32.8	2.74
218	31:37	32.8	2.72
236	32:42	32.8	2.67
254	33:51	36.0	2.68
272	35:03	36.0	1.02
290	36:15	36.0	1.80
297	36:45	8.4	1.76
301	37:01	8.4	1.77
305	37:18	8.3	1.71
309	37:35	8.3	1.68
327	38:54	39.4	0.95
345	40:13	39.4	0.95
363	41:33	41.1	1.02
381	42:57	42.1	0.97
399	44:22	42.1	1.01
409	45:09	4.9	0.98
413	45:28	9.8	1.23
417	45:48	9.8	0.98
421	46:07	9.8	0.97
425	46:27	9.7	0.91
429	46:46	9.8	0.91
433	47:06	9.9	0.79
437	47:26	9.9	1.08
441	47:46	9.9	0.99
445	48:06	9.9	0.86
449	48:26	10.0	0.84
453	48:45	9.9	0.88
457	49:06	10.3	0.97
461	49:27	10.3	0.96
465	49:47	10.3	0.71
469	50:08	10.3	1.01
473	50:28	10.3	0.90
477	50:49	10.3	1.00
481	51:10	10.5	0.77
485	51:31	10.5	0.99
489	51:52	10.5	0.95
493	52:13	10.6	0.94
497	52:34	10.6	0.46
501	52:55	10.6	0.53
505	53:16	10.4	0.56
509	53:37	10.4	0.70
513	53:58	10.4	0.80
517	54:18	10.0	0.80
521	54:38	10.0	0.80
525	54:58	10.0	0.84
529	55:18	10.1	0.80
533	55:38	10.1	0.82

Img #	Timestamp (mm:ss)	Tstep (sec)	height above bed (in)
537	55:59	10.1	0.79
541	56:19	10.2	0.88
545	56:39	10.2	0.76
549	57:00	10.2	0.77
553	57:20	10.3	0.77
557	57:41	10.3	-2.35
561	58:02	10.3	-2.35
565	58:22	10.4	-2.35
569	58:43	10.4	-2.35
573	59:04	10.4	-2.35
577	59:25	10.4	-2.35
581	59:46	10.4	-2.35
585	00:07	10.4	-2.35
589	00:27	10.4	-2.35
593	00:48	10.4	-2.35
597	01:09	10.4	-2.35
601	01:30	10.4	-2.35
605	01:51	10.4	-2.35
609	02:12	10.4	-2.35
613	02:32	10.4	-2.35
617	02:53	10.4	-2.35
621	03:14	10.4	-2.35

Table 16: CB2 Sliding Experiment: velocity and storage data

Img #	Timestamp (mm:ss)	Tstep (sec)	Surface Velocity (m/sec)	Sliding Velocity (m/sec)	S (ml)	ds/dt (ml/sec)
302	37:06		6.5E-05	1.1E-05		
335	39:25	140	6.2E-05	1.2E-05		
490	51:57	752	5.3E-05			
495	52:23	26	5.2E-05			
502	53:00	37	5.2E-05			
509	53:37	37	4.7E-05			
514	54:03	25	5.1E-05			
532	55:33	91	4.6E-05			
548	56:54	81	4.7E-05	6.0E-06	50.6	-0.14
552	57:15	21	5.0E-05	1.0E-05	47.8	-0.53
556	57:36	21	4.6E-05	2.9E-06	36.8	-0.04
560	57:57	21	4.8E-05	7.2E-06	35.9	-0.18
564	58:17	20	4.5E-05	4.1E-06	32.2	-0.05
578	59:30	73	4.6E-05	6.5E-06		
596	01:08	38	4.4E-05			

A.6.7 CB2 Patchy Sliding Experiment

Table 17: CB2 Patchy Sliding Experiment: discharge data

Tag #	Timestamp (hh:mm:ss)	Time (sec)	Mass (g)	Volume (m^3)
1	20:58:49	13	38	3.8E-05
2	20:59:03	27	85	8.5E-05
3	20:59:17	41	#N/A	#N/A
4	20:59:32	56	62	6.2E-05
5	20:59:46	70	521	5.2E-04
6	21:00:00	84	517	5.2E-04
7	21:00:14	98	542	5.4E-04
8	21:00:28	112	583	5.8E-04
9	21:00:42	126	744	7.5E-04
10	21:00:56	140	694	7.0E-04
11	21:01:10	154	763	7.6E-04
12	21:01:24	168	786	7.9E-04
13	21:01:38	182	817	8.2E-04
14	21:01:52	196	942	9.4E-04
15	21:02:06	210	964	9.7E-04
16	21:02:20	224	1012	1.0E-03
17	21:02:34	238	1125	1.1E-03
18	21:02:48	252	1242	1.2E-03
19	21:03:02	266	1358	1.4E-03
20	21:03:16	280	1457	1.5E-03
21	21:03:30	294	1524	1.5E-03
22	21:03:45	309	1633	1.6E-03
23	21:03:59	323	1731	1.7E-03
24	21:04:13	337	#N/A	#N/A
25	21:04:27	351	1887	1.9E-03
26	21:04:42	366	1987	2.0E-03
27	21:04:56	380	2065	2.1E-03
28	21:05:10	394	2186	2.2E-03
29	21:05:24	408	2278	2.3E-03
30	21:05:39	423	2404	2.4E-03
31	21:05:53	437	2492	2.5E-03
32	21:06:08	452	2571	2.6E-03
33	21:06:22	466	2615	2.6E-03
34	21:06:36	480	2648	2.7E-03
35	21:06:50	494	#N/A	#N/A
36	21:07:04	508	2726	2.7E-03
37	21:07:18	522	2783	2.8E-03
38	21:07:32	536	2792	2.8E-03
39	21:07:46	550	2829	2.8E-03
40	21:08:00	564	2873	2.9E-03
41	21:08:14	578	2938	2.9E-03
42	21:08:29	593	#N/A	#N/A
43	21:08:43	607	3079	3.1E-03
44	21:08:57	621	3165	3.2E-03
45	21:09:11	635	3223	3.2E-03
46	21:09:25	649	3282	3.3E-03
47	21:09:39	663	3317	3.3E-03
48	21:09:53	677	#N/A	#N/A
49	21:10:07	691	3342	3.3E-03
50	21:10:22	706	3356	3.4E-03
51	21:10:36	720	3369	3.4E-03
52	21:10:50	734	3381	3.4E-03
53	21:11:04	748	3394	3.4E-03
54	21:11:18	762	3406	3.4E-03
55	21:11:33	777	#N/A	#N/A
56	21:11:47	791	3434	3.4E-03
57	21:12:01	805	3445	3.5E-03
58	21:12:15	819	3474	3.5E-03
59	21:12:29	833	3484	3.5E-03
60	21:12:43	847	3494	3.5E-03
61	21:12:58	862	3505	3.5E-03
62	21:13:12	876	3515	3.5E-03
63	21:13:26	890	3525	3.5E-03
64	21:13:40	904	3541	3.5E-03
65	21:13:54	918	3558	3.6E-03
66	21:14:08	932	3565	3.6E-03
67	21:14:22	946	3575	3.6E-03
68	21:14:36	960	3591	3.6E-03
69	21:14:50	974	3611	3.6E-03
70	21:15:04	988	3645	3.7E-03
71	21:15:18	1002	3666	3.7E-03
72	21:15:32	1016	3684	3.7E-03
73	21:15:47	1031	3709	3.7E-03
74	21:16:01	1045	3738	3.7E-03
75	21:16:15	1059	#N/A	#N/A
76	21:16:29	1073	3816	3.8E-03
77	21:16:43	1087	6848	6.9E-03
78	21:16:54	1098	6822	6.8E-03

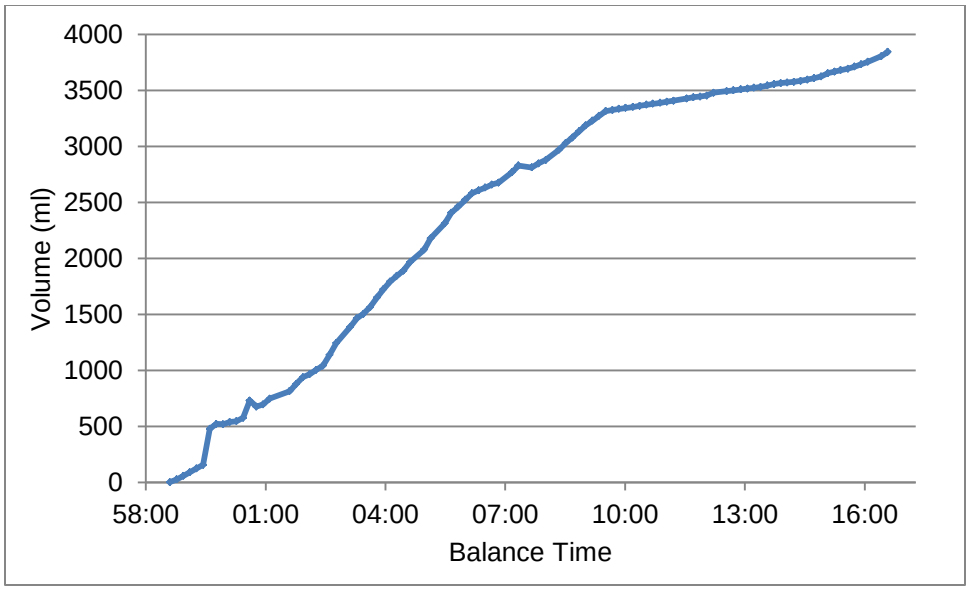

Figure 28: CB2 Patchy Sliding Experiment: discharge data

Table 18: CB2 Patchy Sliding Experiment: water pressure data

Img #	Timestamp (mm:ss)	Water Height (in)	Floatation Fraction
96	04:11	3.25	1.62
106	04:44	3.00	1.50
116	05:16	2.99	1.49
126	05:49	2.94	1.47
136	06:22	2.95	1.47
146	06:55	3.02	1.51
156	07:29	3.20	1.60
166	08:04	3.39	1.70
176	08:39	3.11	1.56
186	09:14	3.04	1.52
196	09:49	2.90	1.45
206	10:28	3.01	1.51
216	11:08	2.99	1.49
226	11:47	3.05	1.53
236	12:26	2.96	1.48
246	13:06	2.95	1.47
256	13:49	2.86	1.43
266	14:32	2.82	1.41
276	15:15	2.70	1.35
286	15:58	2.77	1.38
296	16:41	2.73	1.37
306	17:26	2.76	1.38
308	17:35	2.77	1.38
311	17:50	2.55	1.27
314	18:04	2.69	1.35
317	18:19	2.99	1.50
320	18:33	2.90	1.45
323	18:48	2.87	1.43
326	19:03	2.80	1.40
329	19:17	2.59	1.30
332	19:32	2.34	1.17
333	19:37	2.62	1.31
334	19:41	2.90	1.45
335	19:46	3.12	1.56
336	19:51	3.20	1.60
337	19:56	3.14	1.57
338	20:00	3.07	1.54
339	20:05	3.10	1.55
340	20:09	3.06	1.53
341	20:14	3.07	1.54
342	20:18	3.01	1.50
343	20:23	2.99	1.50
344	20:28	2.93	1.47
345	20:33	2.88	1.44
346	20:38	2.73	1.37
347	20:43	2.41	1.21
348	20:47	2.52	1.26
349	20:52	2.79	1.39
350	20:56	3.06	1.53
351	21:01	3.26	1.63
352	21:06	3.49	1.75
353	21:11	3.56	1.78
354	21:16	3.45	1.72
355	21:21	3.31	1.66
356	21:26	3.42	1.71
357	21:30	2.88	1.44
358	21:35	2.72	1.36
359	21:41	2.65	1.33
360	21:46	2.62	1.31
361	21:51	2.64	1.32
362	21:56	2.37	1.19
363	22:01	2.70	1.35
364	22:06	3.01	1.50
365	22:11	3.05	1.52
366	22:17	3.03	1.52
367	22:22	3.01	1.50
368	22:27	2.95	1.47
369	22:32	2.91	1.45
370	22:37	2.47	1.24
371	22:42	2.87	1.43
372	22:47	3.15	1.57
373	22:52	3.12	1.56
374	22:57	3.03	1.51
375	23:03	2.93	1.47
376	23:08	2.92	1.46
377	23:14	2.88	1.44
378	23:19	2.90	1.45
379	23:25	2.54	1.27
380	23:30	2.88	1.44
381	23:35	1.36	0.68
382	23:41	1.53	0.77
383	23:46	1.66	0.83
384	23:51	1.75	0.88
385	23:57	1.76	0.88
390	24:24	1.84	0.92
395	24:51	2.10	1.05
400	25:18	2.15	1.07
405	25:45	2.14	1.07
410	26:11	2.00	1.00
415	26:38	1.93	0.97
420	27:05	1.86	0.93
425	27:32	1.79	0.89
430	27:59	1.78	0.89
435	28:26	1.79	0.89
440	28:53	1.80	0.90

Table 19: CB2 Patchy Sliding Experiment: surface velocity data

Img #	Timestamp (mm:ss)	Vsurface (m/sec)
321	18:39	5.95E-05
323	18:49	5.74E-05
325	18:58	5.57E-05
327	19:08	6.53E-05
329	19:17	6.02E-05
331	19:27	
333	19:37	
335	19:46	5.91E-05
337	19:55	5.69E-05
341	20:14	5.55E-05
345	20:33	5.06E-05
347	20:43	6.43E-05
349	20:52	6.63E-05
351	21:01	5.83E-05
352	21:06	6.63E-05
354	21:16	5.39E-05
356	21:26	6.10E-05
358	21:36	6.66E-05
360	21:46	5.44E-05
362	21:56	5.95E-05
364	22:06	5.67E-05
366	22:16	6.08E-05
368	22:26	5.79E-05
371	22:42	5.96E-05
373	22:52	5.86E-05
375	23:03	5.57E-05
377	23:13	5.93E-05
382	23:39	5.77E-05
393	24:35	5.72E-05
411	26:11	5.25E-05
429	27:48	4.86E-05