

Towards a full characterization of laser ejection as
a scheme for studying stochastic heating

Kevin Caleb Eades

Professor Thomas D. Donnelly, Advisor



Department of Physics

May, 2016

Copyright © 2016 Kevin Caleb Eades.

The author grants Harvey Mudd College the nonexclusive right to make this work available for noncommercial, educational purposes, provided that this copyright statement appears on the reproduced materials and notice is given that the copying is by permission of the author. To disseminate otherwise or to republish requires written permission from the author.

Abstract

Small-scale fusion could provide a fruitful clean energy source in the future. A proposed method of heating electrons in plasma to induce such fusion with a Coulomb explosion is stochastic heating. To investigate this, an isolated nanosphere plasma target must be placed in the focus of a terawatt, femtosecond-laser pulse. Continuing from previous results, we have developed a method of consistently delivering these plasma targets to a precise spatial location in a specified time interval. Our two-stage deposition process of drying and oxygen plasma etching ensures the targets are suitable for ablation, whereby the spheres are scattered into the terawatt laser focus. Experimental data and numerical modeling demonstrate the reliability of our target delivery method.

Acknowledgments

I would like to acknowledge several people without whom this work would not have been remotely feasible.

First and foremost, many thanks to Aaron Rosenthal, a truly phenomenal person whose daily insights taught me more than I can possibly imagine. His work and probing questions inspired this thesis and, more than that, a meaningful friendship I will always cherish. There is no one I would have rather worked with, lived with, and explored the quirks of the wonderful city of Austin, Texas with. Cheers mate.

Second, I would be remiss if I didn't thank my many lab comrades. To my ever-so-talented friend Amber, thanks for putting up with my less-than-standard lab techniques and challenging my every notion without fail; also for making me listen to all that beautiful classical music. Hao, thank you for being my SEM guy and being a great friend outside of lab. Marc, thanks for always asking me every question I didn't want to answer; it made me think much more deeply before jumping to conclusions, which was incredibly valuable.

To the rest of the DonLabs members, Casey, Laura, Sophie and Sakshi, you guys are okay too I guess.

To the Don himself, it has been great working for you. I couldn't ask for a better research advisor, mentor and friend. You are truly an inspiring professor and I honestly don't think I would be in physics if it wasn't for you. Even though I probably don't say it enough, I really appreciate everything you have done for me.

And lastly, but most importantly, to my parents Kevin and Norma Eades. You both pushed me to become the best individual I could be, and for that I will be forever grateful. I would not be where I am today without your love and support.

Contents

1	Introduction	1
1.1	Impact on society	1
1.2	Stochastic heating	2
1.3	Previous Work	4
2	Deposition work	5
2.1	Requirements for stochastic heating studies	5
2.2	Sample preparation, collection, and plasma etching	6
3	The time-of-flight experiments	9
3.1	Setup	9
3.2	Past results	11
3.3	Improving experimental techniques	11
4	Preliminary modeling	14
4.1	Needs for characterization from past time-of-flight results	14
4.2	Novel fitting mechanisms: Monte Carlo and probabilistic	15
4.3	Assumptions	16
4.4	Probabilistic model	17
4.4.1	Maxwell Boltzmann distribution	17
4.4.2	One dimensional model	19
4.4.3	Geometry of the three dimensional model	20
4.4.4	The three dimensional model	23
4.4.5	Results	24
4.4.6	Error analysis	25
4.5	Monte Carlo model	26
4.5.1	Accounting for HCP depositions	26
4.5.2	Results	27
4.5.3	Error analysis	29
4.6	The fitting mechanism	30
4.7	Weaknesses and the need for a better model	32
5	The better model: zero free parameters	33
5.1	Assumptions	33
5.2	Fresnel coefficients	34
5.3	The ablation laser	36
5.4	Absorption	37
5.5	Additional effects: diffusion and Boltzmann radiation	38
5.6	Expansion	39
5.7	Modified finite difference method	39
5.8	The governing coupled partial differential equations	42
5.9	Values used	43

5.10 Results	44
5.11 Error analysis	48
6 Improving the model	50
6.1 Ball lensing	50
6.1.1 Assumptions	51
6.1.2 Spherical symmetry and circular geometry	52
6.1.3 Turning geometry into a model	55
6.1.4 Results	56
6.1.5 Updating the better model for 5 μm spheres	59
6.2 Ejection angles	63
6.3 Sphere transparency	65
7 Outlook: bringing everything together	68
7.1 Comparing the models, with data	68
7.2 What the models tell us	73
8 Future work	78
8.1 Possibly useful experiments	78
8.2 Time-of-flight data with plasma etched spheres	78
8.3 Further model improvements	79
9 Conclusion	80
10 References	81
11 Appendix A: Code	84
11.1 Measuring the focus	84
11.2 Time-of-flight data processor	86
11.3 Probabilistic model	89
11.4 Monte Carlo fitting	95
11.5 Zero free parameter model	99
11.6 Ball lensing	111
11.7 Fitting graphical-user-interface	115

List of Figures

1.2.1	Multipass stochastic heating diagram	3
2.2.1	Comparison of 500nm depositions	7
2.2.2	Collection images revealing bonding post-ablation	8
3.1.1	Experimental setup for time-of-flight experiments	10
3.2.1	Sample time-of-flight trace for 500 nm spheres	12
4.4.1	One-dimensional time-of-flight setup	19
4.4.2	Ejection plume viewed along the scattering beam path	20
4.4.3	Ejection plume viewed from above the scattering beam path	21
4.4.4	Geometry for the probabilistic model	22
4.4.5	Probabilistic model time-of-flight trace	24
4.5.1	HCP packing algorithm for the Monte Carlo simulation	27
4.5.2	HCP algorithm results: intial sphere positions	28
4.5.3	Three sample Monte Carlo simulation time-of-flight traces	28
4.6.1	Time-of-flight data analysis GUI	31
5.2.1	Snell's law setup	35
5.7.1	Setup for computing one-sided numerical derivatives	40
5.7.2	Setup for computing first- and second-order numerical derivatives	41
5.10.1	Surface expansion mesh plot	44
5.10.2	Heat map of the surface expansion	45
5.10.3	Radial trace from Figure 5.10.1.	46
5.10.4	Temporal trace from Figure 5.10.1.	46
5.10.5	Ejection velocities from the first-principles model	47
5.10.6	Simulated PMT signal from the first-principles model	47
5.10.7	Simulated PMT signal from the first-principles model	48
6.1.1	Burn spots on silicon surface post-ablation of 5 μm spheres	51
6.1.2	Geometry for the first lensing scenario	53
6.1.3	Geometry for the second lensing scenario	54
6.1.4	Pulse intensity incident on the silicon surface after lensing	57
6.1.5	Total power incident on silicon substrate	57
6.1.6	Heat map visualization of ball lensing effect	59
6.1.7	Surface expansion with the ball lensing effect	60
6.1.8	Heat map of surface expansion with ball lensing	61
6.1.9	Temporal trace from Figure 6.1.7	61
6.1.10	Radial trace from Figure 6.1.7	62
6.1.11	First-principles ejection velocities with ball lensing	62
6.1.12	Simulated first-principles PMT trace with ball lensing	63
6.2.1	First-principles model θ ejection angle as a function of radius	64
6.3.1	Absorbance spectrum for polystyrene	66
7.1.1	Comparison of probabilistic and simulation models with data	69
7.1.2	Comparison of data with the first-principles model	70

7.1.3	Comparison of data with model including ball lensing	72
7.2.1	THOR laser timing by shooting distance for 500 nm spheres	74
7.2.2	THOR laser timing by shooting distance for 2 micron spheres	75
7.2.3	THOR laser timing by shooting distance for 100 nm spheres	75
7.2.4	THOR laser timing by ablation pulse intensity	76
7.2.5	THOR laser timing by misalignment angle	77

List of Tables

1	Lens parameters for the ablation path at Mudd	10
2	Time-of-flight results: experimental sphere ejection velocities	12
3	Parameters of the first-principles model	43

1 Introduction

This thesis is a continuation of a collaborative experimental effort between Harvey Mudd College and the University of Texas at Austin to study a theorized heating mechanism for producing hot electrons in a high-intensity electromagnetic field from a pulsed, femtosecond laser. The specific goal of the work done at Mudd under Professor Thomas D. Donnelly has been focused on coming up with a mechanism for reliably delivering spherical plasma targets to the focus of a high-intensity laser in order to study this theory. In particular, this thesis analyzes the latest efforts for such target delivery and suggests a route towards the future for our experiment with methods of improving the reliability and likelihood of our mechanism succeeding.

1.1 Impact on society

A lot of work has been done on large-scale fusion such as at the National Ignition Facility (NIF). There, they use 192 lasers incident on a gold hohlraum to attempt to stimulate indirect-drive inertial confinement fusion (ICF). In a somewhat recent experiment, NIF observed fuel gain exceeding unity using a method of manipulating plasma instabilities with laser pulse shapes [19]. This is a major step towards ignition, which is a crucial step in making fusion into a viable alternative energy source. The experiment observed some evidence of the bootstrapping needed to accelerate a fusion burn and cause the process to ignite.

Fusion is not a new concept for energy, having been researched since the early 1950's after the development of thermonuclear explosives with an international cooperative program initiated in 1955 to develop a means of producing controlled thermonuclear energy releases in an effort to create ignition conditions [2]. Since 1969, ICF has been studied extensively with NIF coming online in recent years. Unfortunately, even with the promising results of the fuel gain exceeding unity, a technical report by Tipton in 2015 suggests that the χ parameter must be at least doubled at NIF in order to cross the threshold for ignition [30]. Crudely, χ is the product of the plasma (electron) density n_e and the 'energy confinement time' τ_e . This is a major technological hurdle that must be overcome if large-scale inertial confinement fusion plants are to become feasible. Nonetheless, some estimate that these hurdles will be surmounted eventually with a proof-of-concept ICF power plant as early as 2030 [24]. Granted, this estimate was made in 2004 before many of the major hurdles of NIF were discovered.

There is also still ongoing research into another possible source of fusion energy, magnetic confinement fusion, with international efforts in projects such as the ITER tokamak being heavily invested in as a promising new source.

Overall, fusion is of great international interest as a potential alternative energy source. The International Energy Agency (IEA) based in Paris projects that

between 2009 and 2035, electricity consumption needs will increase from 20,000 to 30,000 TWh while carbon-dioxide emissions will increase from 29 to 43 gigatonnes per year under current policies [9]. As of 2011, less than 5% of the world's energy came from renewable sources with coal, gas, oil and nuclear accounting for over 80% of the total energy picture. Increasing evidence suggests emissions from such energy sources are linked to climate change, such as the rapid increase in atmospheric CO₂ concentrations up to current levels of just over 400 parts per million (ppm). See Hansen et. al.'s paper for an interesting discussion of this, with a detailed analysis of what ppm humanity should aim for [16]. Moreover, the renewable sources that are currently readily available, namely wind and solar, have efficiency problems that inhibit their wider usage. As of 2001, for example, it would take 2,197,800 windmills with 20 meter rotor blades operating at the average wind speed of the North Sea Coast or 33,000 km² of solar panels at a 10% efficiency to meet the energy demand of the United States at that time [25]. Since then, the energy needs have only increased while the technology efficiency has not kept pace.

With fusion, there would be little to no waste in addition to an essentially infinite reservoir of fuel with deuterium being naturally abundant in the oceans and tritium obtainable via neutron capture on deuterium [22]. Further, fusion reactions hold great promise for being very energetically favorable, evidenced for example by the Sun using these reactions as its primary source of energy generation. Fusion could be an integral part of the clean energy future. Hence, it holds more promise than the relatively inefficient current alternative energy options.

While large-scale fusion has been studied extensively with international projects continuing the effort, smaller scale processes have been less well characterized. Stochastic heating has implications for future small-scale fusion studies. If it is shown to be the heating mechanism behind hot electron generation when a plasma target is in the presence of a strong laser field, then this could potentially be exploited to study small-scale fusion possibilities. Such work could be important to advancing the field and stimulating the future of clean energy.

1.2 Stochastic heating

Stochastic heating is a theorized multipass heating mechanism whereby electrons are accelerated to relativistic speeds while shielded by a plasma in the presence of a strong laser field. For a thorough overview of this process, we refer the reader to the work of Briezman [4] [3], the NSF funding renewal proposal of Donnelly and Ditmire [10], and the work of fellow graduating senior Laura Zhang [33]. Here, we will give a very cursory look at the general idea behind the physics of the heating mechanism, taking results straight out of the NSF proposal.

From Figure 1.2.1, the general process of multipass stochastic heating can be

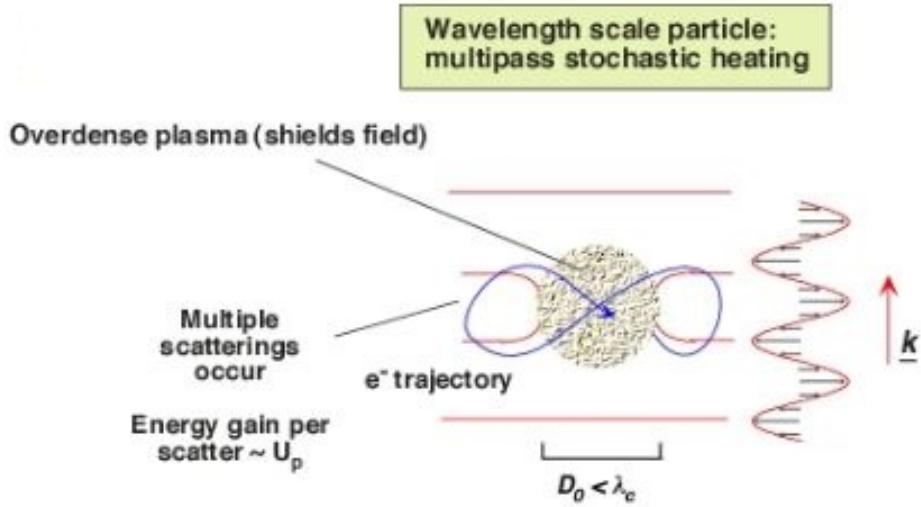


Figure 1.2.1: Basic idea behind multipass stochastic heating. Figure taken directly from Donnelly and Ditmire's NSF proposal [10].

described. When a wavelength-scale plasma target is in the presence of a strong laser field, it becomes an over-ionized plasma. Electrons are then shielded from the electromagnetic fields while inside the target. As an electron goes out into the laser field, it is kicked back into the plasma, which shields it and develops a phase difference as the electron travels through it. By the time it reaches the other side, it is sufficiently de-phased from the strong laser field to be kicked back into the plasma again for another pass. On each such pass, the electron gains energy on the scale of the ponderomotive energy. The hot electrons go into the relativistic regime and gain an energy given by

$$E_{hot} \approx \sqrt{m_e^2 c^4 + \frac{c^2 U_p^2 \tau_{las}^2}{D_0^2}} - m_e c^2, \quad (1.2.1)$$

where U_p is the ponderomotive energy and τ_{las} is the pulse duration of the laser, and D_0 is the plasma structure dimension. Further, the stochastic heating puts the requirement that for relativistic electrons, this heating in Equation 1.2.1 will only work if the plasma structure dimension is greater than the wavelength of the laser but less than the mean free path of the fast electron.

Our stochastic heating experiment seeks to measure the electron energy signatures when a wavelength-scale plasma target is placed in the focus of a high-intensity laser.

1.3 Previous Work

Thus far, the work done at Mudd has been centered on how to deliver plasma targets to the focus of a high-intensity laser. This will be discussed in some detail in Sections 2 and 3 with regards to the most recent efforts at developing a target delivery method. In the past, piezoelectric oscillators were used to generate wavelength-scale droplets in a suspension; former group member Andrew Higginbotham used such devices to deliver heavy water molecules to a high-intensity laser to study fusion yields and found promising results [17] [18].

He also proposed a mechanism for delivering polystyrene nanospheres to the focus of a high-intensity laser via electrostatic ejection. This technique was studied by the group for a while before converting to our current method of laser ejection. Former group members Brendan Folie [12] and Hong Sio [28] laid out the groundwork for why laser ejection is a more viable method than electrostatic as well as providing initial insights into a deposition method for creating samples. Former group members Andrew Yandow [32] and Aaron Rosenthal [26] then developed the laser ejection technique much more on the experimental side, eventually leading to time-of-flight data that shows the scheme is indeed viable for getting polystyrene nanospheres to ablate off a silicon substrate, plume outwards and interact with a crossing laser. Unfortunately, at least part of this data is almost undoubtedly for bonded spheres, which is not permissible for a stochastic heating experiment. Further, the laser used to ascertain interactions post-ablation is orders of magnitude larger in beam width than the high-intensity laser focus used for the experiment.

In this thesis, we develop the laser ejection method even further. For one thing, we assess the progress in eliminating bonds between spheres as well as other experimental improvements building off previous year's iterations. The bulk of our time will be dedicated to determining whether or not time-of-flight data taken at Mudd will be applicable to the stochastic heating experiment with a high-intensity, focused laser. We do this through the development of several models that vastly improve our understanding of the entire laser ejection process.

2 Deposition work

In order to study stochastic heating, there are certain requirements on how wavelength-scale targets are delivered to the focus of a high-intensity laser. These must be satisfied in order to have any possibility of observing energy signatures that are not lost among various undesired signals, as mentioned in Section 1.3. Additionally, this criteria must be weighed against the probability of actually hitting a sphere on any given shot.

2.1 Requirements for stochastic heating studies

As outlined in Section 1.2, stochastic heating is a mechanism for electron energy gain in the presence of a high-intensity oscillating electric field. To successfully couple energy from the laser to electrons, however, the stochastic heating mechanism requires that the electrons must be shielded in a wavelength-scale particle [3]. Again, this is because stochastic heating is a multi-pass process whereby electrons are driven through multiple oscillations of the field, similar to driving a simple harmonic oscillator at resonance [33]. The wavelength-scale particle ensures that the electron's phase on each pass is optimal for such driving. Hence, our first requirement is that the target diameters be on the order of the high-intensity laser's wavelength.

Another key to studying stochastic heating is that a maximal amount of the laser's energy must be coupled directly to the target. That is, lost energy to anything other than the target should be minimized. This means that our targets must be delivered to the focus without any sort of supporting structures. Such structures would tend to act as a sink and divert energy away from being coupled entirely to the electrons. This requirement pretty much leaves two options for the study of stochastic heating: 1) optical trapping and 2) plume target delivery.¹ The former option requires using another laser to trap (at least partially) reflective or transmissive particles in the intensity wells of that laser's profile. Such trapping holds the particle at a certain position with some uncertainty. Many are working on such methods, including a fellow graduating senior in our own group, Casey Cannon, and former group member Jake Fish. See Casey [8] or Jake's [11] thesis for details on the process of optical trapping and efforts to improve its stability. Option (2) requires sending large numbers of targets towards the high-intensity laser focus to have some, ideally high, probability of hitting at least one target that is not in physical contact with any other targets or supporting structures of any sort. Previous experiments have had success in sending particles to the focus using air streams, as with Higginbotham's aerosols [17]. We will take the approach of laser ejection for sending large plumes of targets towards the focus.

¹This actually holds true across a variety of experiments that study interactions between high-intensity lasers and small isolated targets.

Option (1) is quite viable and is indeed used in a variety of experiments, particularly in atomic, molecular and optical (AMO) physics, where trapping is used to obtain novel measurements on ultracold atoms, for example. The experimental utility of optical tweezers has been developed for some time and is fairly well known today. For an overview and a guide to finding everything about optical tweezers, see Lang and Block's Resource Letter [20]. However, its drawbacks, for stochastic heating studies at least, are that only one shot may be taken per vacuum cycle and it requires close attention to detail in the setup and alignment of the trapping beam(s) compared to the high-intensity laser focus.² So we employ option (2), where many more shots can be done per vacuum cycle, at least tens and possibly hundreds if the chamber setup allows for motor drivers along both principle axes perpendicular to the ablation laser's propagation axis. Further, aligning the high-intensity and ablation lasers to be in proper position relative to each other is much more straightforward.

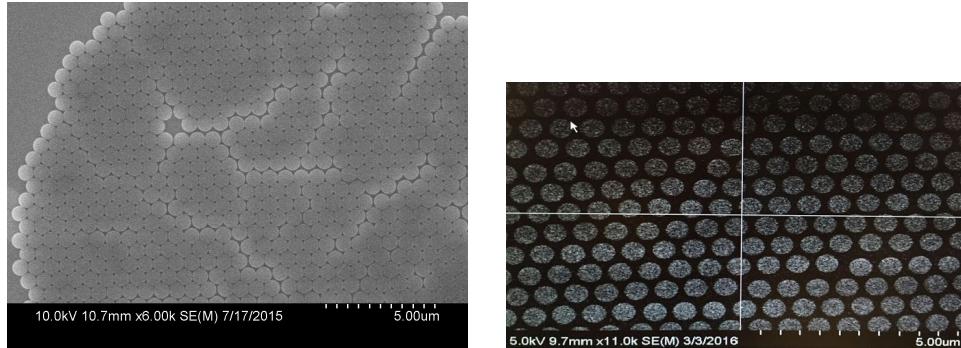
The major drawback to option (2) is that the position of targets is even less well-defined than in option (1). This implies that the statistical probability of getting any particular target to the high-intensity laser focus at a given time is very low. Fortunately, this downside can be overcome by ablating a large number of targets such that the overall probability of getting one to the desired position at the proper time approaches one. This does impose a trade-off, however, as the more targets ablated on any given shot, the fewer total number of shots can be done in a vacuum cycle. However, the ablating more targets side of this trade-off has diminishing returns: spheres that are initially farther from the albtion laser axis will have a lower probability of reaching the high-intensity laser focus that is aligned to focus at a point on that axis some distance from the sample. Hence, there is no need to ablate a spot size above a hundred microns for a ten micron high-intensity laser focal spot size, for example, because the edge spheres being ablated have nearly zero probability of reaching the focus.

With our depositions and time-of-flight characterizations, we can attune our laser ejection technique to both provide a high probability of delivering a target in isolation to the high-intensity laser focus and allow for a large number of shots per vacuum cycle. The former is important for getting useful data on stochastic heating while the latter ensures robustness of that data.

2.2 Sample preparation, collection, and plasma etching

In laser ejection, we deposit the targets onto a substrate. For our experiments, we use polystyrene nanospheres to coat a silicon substrate. These plastic spheres are easily purchased and allow for a fairly precise size scale of the targets. For example, the nominal 500 nm spheres are 498 ± 9 nm for their mean diameter with

²Depending on the experimental setup and requirements, more than one trapping beam may be necessary.



(a) A particularly beautiful HCP region on a deposition.

(b) An etched sample in a region that was formerly HCP with bonds between spheres.

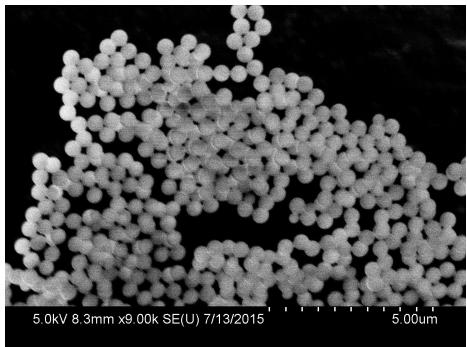
Figure 2.2.1: Side-by-side comparison of 500 nm sphere depositions: one has not been etched (left) and has (right).

a standard deviation of 7.9 nm.³ For the substrate, silicon can be cleaned via pi-ranha etching as detailed in Andrew's thesis [32]. Details of the full deposition process can be found in Aaron's thesis [26] and current, fellow graduating senior Amber Cai's [6]. The key idea is to provide a large number of targets weakly bonded to a surface that can be ejected in a reliable, consistent manner. Trying to maximize the number of spheres ablated per unit area, the tightest packing that can be achieved without layering is hexagonally close packed (HCP).⁴ The goal of our deposition method is therefore to create large regions of HCP targets that can be hit with the ablation laser. Figure 2.2.1a shows a sample HCP region on a deposition.

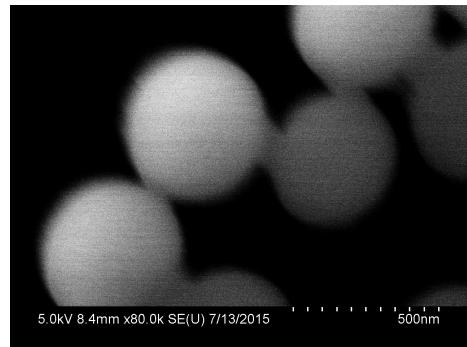
After creating our samples, we ejected them with the ablation laser and subsequently collected them on a charged aluminum foil. According to Brendan's thesis, there is a residual positive charge on the spheres, so with a negative bias on the foil they are attracted via basic electrostatics [12]. They then stick to the foil and can be imaged, revealing characteristics of the ejection. Initially, we were hoping to observe angles of ejection as in Zheng, Lu and Song's paper, which found that the ejected angles resembled a Gaussian in a similar laser cleaning experiment [21]. However, we instead got images such as Figures 2.2.2a and 2.2.2b.

³Note that the uncertainty on the mean diameter means that the spheres have some mean diameter between 489 and 507 nm and the distribution of sizes for spheres in that solution will have a standard deviation of 7.9 nm around whatever that mean is.

⁴Note that layering is undesirable for two reasons. First, the number of layers will vary unpredictably even in small regions, which makes the time-of-flight unreliable. Second, layering implies bonds between the spheres because otherwise the top layers would not remain after depositions. Continue reading in this section for why bonds between spheres are undesirable.



(a) A collection image that shows a network of spheres connected with bonds that remained through ejection.



(b) A close-up view of the bonds between neighboring spheres in the overall structure.

Figure 2.2.2: Images of spheres collected on an aluminum foil revealing the macro-structure (left) and the interspecies bonding (right).

As a result of our collection images, we realized that the spheres were bonding with each other and thereby coming off the substrate in sheets. This is not acceptable given the conditions required for study of stochastic heating outlined in Section 2.1. We attempted a variety of techniques to eliminate the interspecies bonds between targets, but finally settled on plasma etching as in Gogel et. al.'s paper [14] and the wise guidance of Prof. van Ryswyck of Harvey Mudd College's Chemistry Department. The details of plasma etching, along with a procedure, are found in Amber's thesis [6]. After plasma etching, the bonds between spheres are removed on the deposition, as shown in Figure 2.2.1b.⁵ This will allow the spheres to be ejected individually from the substrate rather than in sheets.

⁵This introduces other considerations that have yet to be fully characterized. Namely, since the oxygen plasma we use etches into polystyrene at a uniform rate on all exposed surfaces, the sphere size is reduced in the process. This could change the time-of-flight characteristics. Additionally, the uniform etching rate implies that the surfactant bonding holding the sphere to the silicon substrate could be weakened as well. Previous work by Andrew [32] and Brendan [12] imply the surface bonds are removed on a timescale orders of magnitude faster than the surface expansion ejection process, so any weakening of surfactant bonding may be insignificant anyways.

3 The time-of-flight experiments

With our deposition process consistently producing uniform monolayers on silicon surfaces, the next task is delivering them to the high-intensity laser focus. To do this, several methods have been attempted in the past, as mentioned in Section 1.3. We will use the laser ejection method, which arises as an extension of laser cleaning, whereby particles are removed from a surface using short-pulsed laser irradiation. This technique has been studied since the early 90s as a method of removing micron and sub-micron scale particles from surfaces, overcoming the strong retentive forces binding them together [29]. For our purpose, we hope to eject our polystyrene nanospheres in a predictable, repeatable manner with an ablative laser pulse.

In the stochastic heating experiment, both the ablation laser and the high-intensity laser are pulsed. To successfully deliver a target to the high-intensity laser focus then, we must be able to reliably predict the spheres' time-of-flight post-ablation so that we know exactly when at least one sphere will be in the focal spot. This is the mainstay of our efforts in the lab at Mudd. We seek to determine the ejection velocity of the spheres as a function of the ablation pulse power for all targets of wavelength-scale size. For two years, we took and analyzed experimental data to determine these ejection velocities.⁶ The results of these experiments are detailed extensively in the respective theses of Andrew [32] and Aaron [26]. Here, we will briefly go over the setup and some results so that the modeling approach can be better understood in the experimental context. This will be invaluable in understanding how the model helps transition from our time-of-flight experiments at Mudd to the stochastic heating experiment at UT Austin.

3.1 Setup

The basic setup for our time-of-flight experiments is displayed in Figure 3.1.1 with the focal length of the ablation path lenses found in Table 1. Our ablation laser is a Q-switched, 532 nm tunable system that we use with a 10 – Hz rep rate (blocking all shots except one released for every individual time-of-flight measurement). At this frequency and with the 40 W nominal power setting, each pulse is 20 ns with a 24 μm full-width-half-max (FWHM) focal spot size at the silicon surface and a fluence of 34.5 J/cm² from a total of 434 μJ of energy.

Our scattering beam is a 488 nm, continuous-wave (CW) argon-ion laser that we run at about 0.3 W nominal power for each time-of-flight measurement.⁷ Post-collimation, it has a roughly 200 μm spot size after lens (v).

⁶Of course, all of this data is likely for bonded spheres rather than individuals based on Figures 2.2.2a and 2.2.2b.

⁷As it turns out, the nominal power agrees closely to the actual power for the argon-ion laser, unlike the green Q-switched laser.

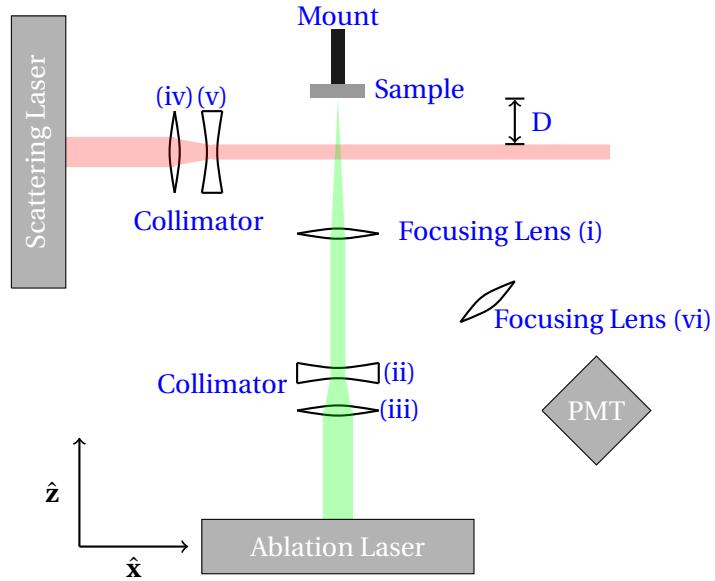


Figure 3.1.1: Pertinent elements of the laser ejection setup. In our scheme, we use a 3 mm x 20 mm sample; D is either 0.5 mm or 0.8mm; lenses (i)-(iii)'s parameters are in Table 1; focusing lens (i) is on a z-translatable stage; the mount is a 3D translatable stage with one stepper motor in the \hat{x} direction.

Lens	(i)	(ii)	(iii)
Focal Length (cm)	100	5	20

Table 1: Lens parameters for Figure 3.1.1. Note for our setup, lens (ii) is actually plano-concave, not concave on both sides.

In a given shot, we let through one pulse of the ablation laser, which is focused onto the sample and ablates the spheres. An oscilloscope is triggered by the pulse along the ablation beam path from a photodiode. This starts data collection on the negative-biased photomultiplier tube (PMT), which takes in focused light through lens (vi) that is scattered by spheres as they pass through the scattering beam. A stronger signal means either more spheres are in the scattering laser or sphere(s) are closer to the beam center where the intensity of scattered light will be greater in the roughly Gaussian profile.

We take data at two distances for D in Figure 3.1.1, which we chose to be 0.5 mm and 0.8 mm. Comparing the peak signal times for these two distances, we can roughly determine the ejection velocity from change in distance (0.3 mm) divided by change in time (temporal distance between the signal peaks).⁸ For a more thorough, detailed explanation of this entire setup, taking data, and analyzing it, please see Aaron's thesis [26].

3.2 Past results

Figure 3.2.1 shows the normalized, negative biased PMT signal as a function of time since ablation for five shots whose traces are summed together (before normalization).⁹ This graph is generated using a technique outlined in Section 3.3. This particular set of data taken on February 19, 2015 is for 500 nm spheres that are 0.8 mm from the scattering beam, shot with the 40W nominal power setting on the Q-switched laser. In other words, $D = 0.8$ mm in Figure 3.1.1.

From traces like these and the techniques mentioned in Section 3.1, Aaron created a table of the ejection velocities as a function of sphere size for the 40W nominal ablation pulse power. This table is shown in Table 2, taken directly out of Aaron's thesis without alteration of the data [26]. The velocities in this table as a function of sphere size in addition to data traces from the lab will be vital for analyzing the models we develop in this thesis.

3.3 Improving experimental techniques

In our lab group, there has been a progression of efforts to improve upon the past experimental techniques in an effort to get more consistent, reliable data. Andrew improved upon the early attempts at delivering targets using laser ejection by starting to analyze the time-of-flight data using a setup almost identical to Figure 3.1.1 [32]. Aaron improved upon Andrew's experimental techniques by developing a more uniform deposition method to create more consistency in the sample quality; Aaron also reduced noise in the signals, determined how to

⁸We call each oscilloscope signal from a time-of-flight measurement a "trace".

⁹The negative bias is to keep in line with how lab data actually looks since the PMT is run at a negative voltage for electronics considerations.

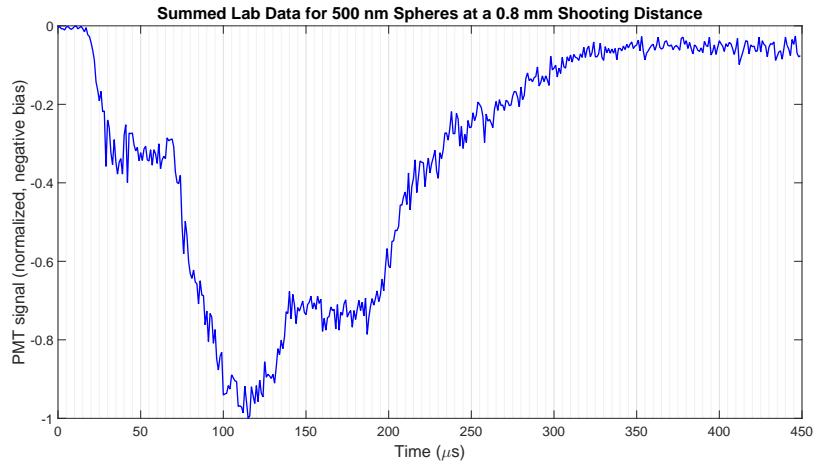


Figure 3.2.1: Sample time-of-flight trace. The PMT signal is plotted, normalized with a negative bias, as a function of time since ablation. Minor grid lines along time help discern times of important features.

Sphere Size Diameter (μm)	Nominal Ablation Power	Velocity (m/s)	Uncertainty (m/s)
.1	40	7.9	2.2
.3	40	9.3	2.0
.5	40	8.3	2.4
.7	40	7.4	1.0
1	40	9.4	2.5
2	40	8.1	2.4
5	40	23.0	6.3

Table 2: Summary of Time of Flight Results. Table and caption taken directly from Aaron's thesis [26].

assess and avoid saturation, and in general improved upon the setup for taking time-of-flight data.

Following this vein, we too sought to improve some of the experimental techniques used by the group in an effort to get better data. There are several areas in which this occurred. The biggest area we improved was detailed in Section 2.2 with the discovery that our spheres are ablating in sheets, stuck together due to undesirable deposition flaws. But again, this is much more thoroughly covered in Amber's thesis [6]. The modeling done to better understand and characterize the ejection mechanism is not exactly laboratory work, and will be covered during the bulk of this thesis.

For lab-specific purposes, we wrote a script that can be used to obtain much more accurate measurements of focal spot sizes than was achieved in Andrew's razor blade method, for example, using only two images: one of the focus and one of the focus with a wire of known, measured size in it. Additionally, to process data, I wrote a series of scripts and functions that take in a number of lab traces (which could be one for a single shot or several to look at aggregate data), set the zero point, normalize, and put in a negative bias to replicate the form of data. This is all operated through a single, user-friendly script that only requires two inputs, the last two digits of the first and last .DAT file numbers.¹⁰ As mentioned before, these scripts and functions were used to generate the graph in Figure 3.2.1 from five lab traces.

There is still work to be done improving experimental techniques. Hao Cao and Marc Finzi will be replacing our metal mirrors with dielectric ones in order to allow the laser to drill more precise holes in aluminum foil for collection experiments instead of poking a sloppy hole of much larger size with a pinhead. This will allow for more accurate collection data, hopefully leading to a more comprehensive analysis of ejection angle characteristics with our improved samples, building off Brendan [12] and Hong's [28] results. Additionally, the optimal plasma etching time is not yet well characterized as a function of sphere size. Our etching method is fairly systematic already and we believe it is capable of reliably producing un-bonded samples starting with our HCP monolayers that have interspecies bonding.¹¹ To do this, though, the optimal etching time must be determined.

¹⁰This is for a consecutive sequence of file numbers such as TEK00003.DAT, TEK00004.DAT and TEK00005.DAT, for example. If needed, the program could be easily modified to take in any set of data files, possibly with a text parser.

¹¹Preliminary evidence of this can be seen in comparing Figures 2.2.1a and 2.2.1b, for example.

4 Preliminary modeling

To be able to analyze our time-of-flight data and, more importantly, assess whether our laser ejection scheme is viable for delivering targets to a high-intensity laser focus, we attempt to model the experiment.

For a first pass at a model, we attempt to simulate the outwards plume of spheres ejected from the silicon substrate by making some assumptions about the net outcome of the ablation pulse's interactions with the spheres and the substrate. From this, we can create time-of-flight traces and compare to those found experimentally by Aaron [26] and Andrew [32].

Note throughout this section that the scattering beam is used as the laser that spheres interact with post-ablation. That is indeed the setup for which the model was developed, although it can and will be run with high-intensity laser parameters in Section 7. In some parts, we bring up the high-intensity laser and really the two are synonymous for the sake of this model except that their radii are different and in Equation 4.4.8, the Δt term is different.

4.1 Needs for characterization from past time-of-flight results

The time-of-flight experiments are all conducted at Harvey Mudd College in Prof. Donnelly's lab. Hence, the data is for a specific set of setup parameters, listed in Table 3. The time-of-flight experiment outlined in Section 3 is not trivial to set up, which could eat into valuable laser time on a bigger facility such as the THOR laser we used previously and the GHOST system that will be used in the coming summer for our stochastic heating experiment.¹² In general, laser shot days are not considered trivial, especially on national scale systems like NIF at Lawrence Livermore National Laboratory and OMEGA at the Laboratory for Laser Energetics. At NIF, the annual operating budget is \$450 million, which makes for an average cost of over a million dollars per day [23].

We cannot assume that the time-of-flight data taken at Mudd will translate to another system unless all parameters are matched exactly, a process which itself could be time intensive and quite costly. Hence, it would be exceptionally useful to have the predictive ability of adjusting time-of-flight results to different setups without repeating the experiment. This would allow us to essentially have the time-of-flight data without spending the days to collect it. This is the motivation for developing a model of the laser ejection experiment.

Further, in the Mudd lab, our scattering beam has a roughly $200 \mu\text{m}$ spot size in the circular $z - y$ cross section with essentially infinite breadth along the x axis, in the frame of a sphere. At UT Austin, the high-intensity laser focus is a roughly

¹²THOR and GHOST are laser systems at the University of Texas at Austin that we have the privilege of using in our collaboration with Professor Todd Ditmire of the Center for High Energy Density Science.

$10\mu\text{m}$ spherical spot. This is orders of magnitude different in scale, which questions the applicability and usefulness of our time-of-flight data. Developing a model for laser ejection could help determine whether or not this data is useful. This question is returned to later in Section 7, and later in this section.

Granted, there are some characteristics which can only be ascertained from the time-of-flight experiments and its related tests. For example, the ablating laser's fluence threshold for damage on the surface is useful and can be matched across systems by simply measuring and tuning the focal spot size of the beam and its energy. These characteristics are used to determine what power level and focusing lens to use on the ablation laser in our stochastic heating experiment. At UT Austin, we last used a Quantel flashlamp laser, as detailed in Aaron's thesis [26]. We determined the power level and focusing lens to use based on trying to match the fluence used for our time-of-flight experiments at Mudd, as detailed in Section 3.1.

Another reason for a model of laser ejection to be developed is simply to understand the process better. Indeed, from three years ago, we had concerns about the laser ejection mechanism with our particular samples that were only verified from probing deeper into understanding how the process works.

4.2 Novel fitting mechanisms: Monte Carlo and probabilistic

Our first take at a model of laser ejection actually turned out to be more useful as a fitting mechanism. With the ejection, we knew our spheres would be coming off the substrate surface with some distribution of velocities based noticing that our time-of-flight traces have some spread in time where there was a large signal, as seen in Figure 3.2.1, for example. Furthermore, from Lu et. al's paper [21] as well as Hong [28] and Brendan's [12] theses, we suspected they would also eject with some angular distribution. Hence, it seemed reasonable to start with distributions for the velocities and angles of ejection and propagate forward in time. When the spheres come across the scattering beam, they produce a PMT signal from the scattered light. How much light is scattered across various angles is determined by Mie scattering, which Michael P. Schubmehl, a former member of our lab group, studied in detail [27].

From the scattered light PMT signals, we simulate time-of-flight data and, more importantly, determine the optimal time to shoot the high-intensity laser after ablation in order to have the highest probability of hitting a sphere.¹³ In the probabilistic model, the output is the percent of spheres that would be hit by the scattering laser as a function of time since ablation. In the Monte Carlo model, the output is the simulated PMT signal as a function of time since ablation.¹⁴

¹³The models will first be run with the scattering time-of-flight parameters in this section and later analyzed with the high-intensity laser in Section 7.

¹⁴This is essentially equivalent to the percent of spheres hit. In general, the bigger the signal, the more spheres hit. See Section 4.4.6 for more detail on this.

Both models take in the most probable velocity of the ejected spheres, which is experimentally determined from the PMT signal peak time after ablation, taken in combination with the separation distance between the sample and the scattering beam, as mentioned in Section 3.2.

4.3 Assumptions

There are several preliminary assumptions that were used to develop the model:

- (i) The velocity distribution for ejected spheres is a Maxwell-Boltzmann with some most probable velocity. This velocity can either be given from experimental data and used to simulate a time-of-flight trace to see how accurate the model is, or it can be left as a free parameter and used to fit the data with a least squares method. The probabilistic model currently operates under the former condition whereas the Monte Carlo model can use both easily.¹⁵ Indeed, for the Monte Carlo model, applying the former condition is covered in Section 4.5 while the latter is covered in Section 4.6.

The justification for using the Maxwell-Boltzmann distribution for the ejection velocities is that during ablation, the spheres could conceivably act similar to a gas in that the contact forces holding the spheres to the surface will inevitably have some variation while the laser beam also has non-uniformity and deviations from a perfect Gaussian. Hence, this could create gas-like conditions where the velocities are selected quasi-randomly from the Maxwell-Boltzmann distribution, with the ‘quasi’ coming from the fact that spheres towards the center of the pulse are likely to eject with a greater speed than those that are farther away due to there being more laser intensity towards the center in the nearly Gaussian distribution. As they plume outwards post-ablation, they resemble a gas-like jet forming a cone with a spherical wavefront.

- (ii) The angular distribution in the θ direction measured from the incoming propagation z -axis of the laser will be a Gaussian. This assumption is justified based on Lu et. al.’s paper [21] as well as an analysis in Brendan’s thesis on particle collection [12]. He discovered that the cutoff θ angle for ejection is at most 20° . Hence, we set this to be three sigma since 99.7% of the values in a Gaussian distribution are found within three standard deviations of the mean on either side. Note that this really means taking the absolute value of the θ angle chosen from a Gaussian distribution centered at zero with a standard deviation of $20/3$. Under this assumption,

¹⁵In fact, the probabilistic model could be made into a fitting mechanism as well, but it would just be significantly more computationally intensive than the Monte Carlo model. Nonetheless, an enterprising future group member could maybe find some interesting results there, although improving the better model outlined of Section 5, as discussed in Section 6, might be a more worthwhile use of resources.

the simulation model is likely more realistic since the probabilistic model assumes a uniform angular distribution over each spherical wavefront.¹⁶

- (iii) The spheres eject with a ϕ angle (in the silicon surface plane perpendicular to the propagation z -axis of the ablation laser) chosen at random in the range $(0, 2\pi)$ for the Monte Carlo model. The spheres are evenly distributed across the angular range for the probabilistic model. Here the probabilistic model is likely much more accurate, although with a large number of spheres the two models will converge in this assumption.
- (iv) For the probabilistic model, all the spheres start at the same position at the center of the ablation laser focus. For the Monte Carlo model, the spheres start in an HCP grid. Those that are within the ablation pulse's focal spot get ablated and have their ejection velocities and angles chosen as detailed in assumptions (i) through (iii). Note that the Monte Carlo model is much more realistic in this assumption. Indeed, the probabilistic model assumption is more or less made because without it, the mathematics would get almost prohibitively difficult when trying to do a probabilistic calculation.

4.4 Probabilistic model

In the probabilistic model, we seek to determine the percent of spheres that would be hit by the scattering laser as a function of the time since ablation. Instead of choosing initial values for each individual sphere to be ejected from the substrate as in the Monte Carlo model, it instead considers what fraction of spheres will lie in each interval within the distributions for the ejection velocity and angle, as noted in Section 4.3.

4.4.1 Maxwell Boltzmann distribution

The Maxwell-Boltzmann equation for the probability of finding a particle to be in a particular velocity, per unit velocity, is given by

$$f(v) = \sqrt{\left(\frac{m}{2\pi kT}\right)^3} 4\pi v^2 e^{-\frac{mv^2}{2kT}}, \quad (4.4.1)$$

where T is the temperature, m is the mass, v is the velocity and k is the Boltzmann constant. Assuming a constant temperature and mass for our spheres throughout the duration of our model (from 0 to 10 microseconds), we can define two new constants.¹⁷ Namely, let $A = \sqrt{\left(\frac{m}{2\pi kT}\right)^3} 4\pi$ and let $B = \frac{-m}{2kT}$. Then,

¹⁶Each spherical wavefront in the probabilistic model is for a different initial velocity since all the spheres start at the same location. See assumption (iv).

¹⁷These are constants only in the sense that T and m are fixed using this as a statistical distribution for the laser ejection behavior, not to replicate the physics behind the mechanism.

Equation 4.4.1 obtains the simpler form of

$$f(v) = Av^2 e^{Bv^2}. \quad (4.4.2)$$

If the parameters A and B are known (which requires knowing m and T), then Equation 4.4.2 may be numerically integrated: the probability of finding a sphere to have a velocity between v and $v + dv$ is given by $f(v)dv$. The total probability of finding a sphere with velocity v between two distinct velocities $v_1 \leq v \leq v_2$ is then

$$\int_{v_1}^{v_2} f(v)dv. \quad (4.4.3)$$

Equations 4.4.2 and 4.4.3 will be the primary equations we use from the Maxwell-Boltzmann distribution in developing our model.

For our experiments, there will be a particular velocity that is most probable for spheres to have as a result of the ablation process. This particular velocity will be denoted v_p for the most probable velocity and is considered a known value. For our particular ablation laser, this most probable velocity is given as a function of sphere size in Table 2.

We can also find the most probable velocity from the Maxwell-Boltzmann distribution itself. Namely, since the Maxwell-Boltzmann distribution gives the probability of finding a particular velocity, per unit velocity, as a function of velocity, the most probable velocity will be when its derivative is zero. Revisiting Equation 4.4.2, the most probable velocity can be obtained from differentiating this equation:

$$\frac{df(v)}{dv} = 2Av e^{Bv^2} + 2ABv^3 e^{Bv^2}. \quad (4.4.4)$$

Setting this equal to zero when $v = v_p$, we can solve for v_p by dividing out common terms and rearranging:

$$v_p^2 = \frac{-1}{B}. \quad (4.4.5)$$

Taking the square root of both sides and re-substituting the value of B into the equation,

$$v_p = \sqrt{\frac{2kT}{m}}. \quad (4.4.6)$$

We see from this derivation of the most probable velocity that if this value is known, then the mass and temperature of the sphere are not necessary to know as well. Hence, since we know the most probable velocity, we do not need to ascertain the temperature and mass of each individual sphere. In other words, we use the distribution to effectively ignore all interactions with the spheres and merely look at the outcome. This is physically invalid as a means of looking at the sphere behavior, but statistically works for developing a model, accurate to the outcome of the ablation mechanism, not necessarily the processes of it.

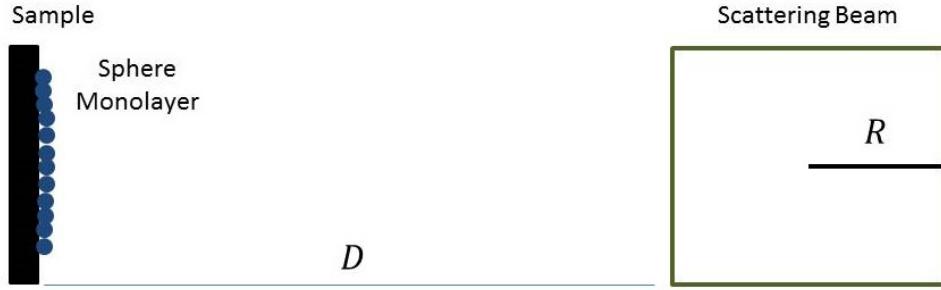


Figure 4.4.1: A one-dimensional visualization of our time-of-flight setup. The polystyrene spheres coat the sample and will be ablated outwards and need to travel a distance D to a scattering beam of radius R .

4.4.2 One dimensional model

In one dimension, the setup will crudely look as represented in Figure 4.4.1. Note that we consider the beam profile to be a square, which is absurd, but will suffice for the general ideas we develop in this one-dimensional model. When we turn to three dimensions, we will be more careful. To create a one-dimensional model, we must also note the basic kinematics equation

$$x = vt, \quad (4.4.7)$$

where x is the distance between the sphere's center and its starting position, v is the velocity of the sphere, and t is the time elapsed from the ablating laser shot. Note here that we neglect any acceleration (due to air resistance, particle interactions, etc.) since we shoot in vacuum, the sphere size is small and the time duration of the model is very short, as preliminarily justified by the peak signal time in Figure 3.2.1 being around $100\mu\text{s}$. Now, if we consider a sphere to be “hit” by the laser if its center is in the beam profile, taking into account the time duration Δt of the high-intensity laser, we can determine the minimum and maximum velocities of a sphere necessary to be hit if the pulse initiates at a time t .¹⁸ Namely,

$$v_{min} = \frac{D}{t + \Delta t}, \quad (4.4.8)$$

$$v_{max} = \frac{D + 2R}{t}. \quad (4.4.9)$$

If we say there are N spheres that get ejected (N being some natural number) when the ablating laser pulse hits, the total number of spheres that would be hit

¹⁸This is really just the sampling time for a PMT and oscilloscope coupled together in our Mudd lab with the CW scattering laser.

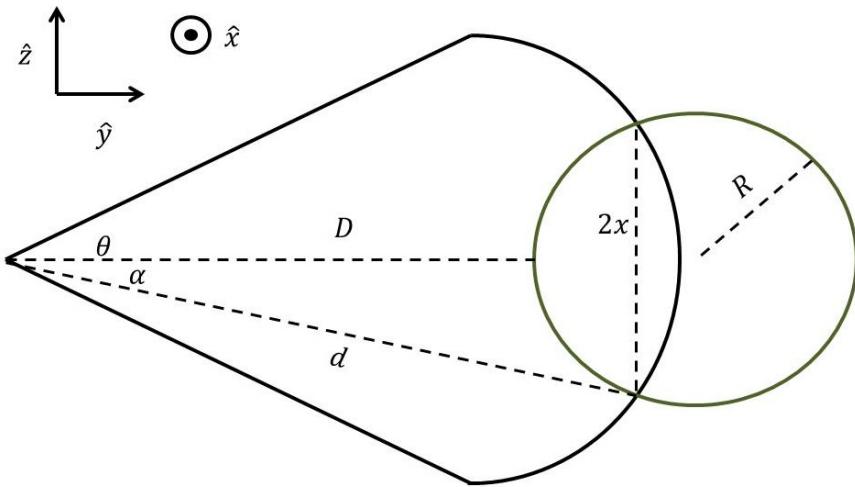


Figure 4.4.2: View along the scattering laser path with the circular beam intersecting the spherical wavefront of our ejected plume of spheres in a cone. Again D separates the sample and the front edge of our scattering beam of radius R .

if the high-intensity laser pulsed starting at a time t , which we will call $N(t)$, is given by applying Equation 4.4.3 to get

$$N(t) = N \int_{v_{min}}^{v_{max}} f(v) dv, \quad (4.4.10)$$

where v_{min} and v_{max} are given by Equations 4.4.8 and 4.4.9, respectively.¹⁹ Setting $N = 100$, Equation 4.4.10 will give the percentage of spheres (out of the total number of spheres that are ejected) that would be hit by the laser at a time t with a distance D between the scattering laser beam front edge and the starting center position of the monolayer.

4.4.3 Geometry of the three dimensional model

In order to build up a three-dimensional model, we must first understand the geometry of it. In the one-dimensional model, if the linear wavefront of spheres propagating outwards has the necessary velocity to be in the scattering beam, then all spheres in the wavefront are for certain hit by it, so there is an “efficiency” of 100%. However, when extending to three-dimensions, we must take into account the fact that the beam only hits a certain fraction of the total area of

¹⁹Again, for the CW scattering beam, “pulsing” at time t really means how many spheres would interact with the laser at that time, either setting Δt to zero in Equation 4.4.8 or the sampling time for a PMT and oscilloscope system as mentioned in the last footnote.

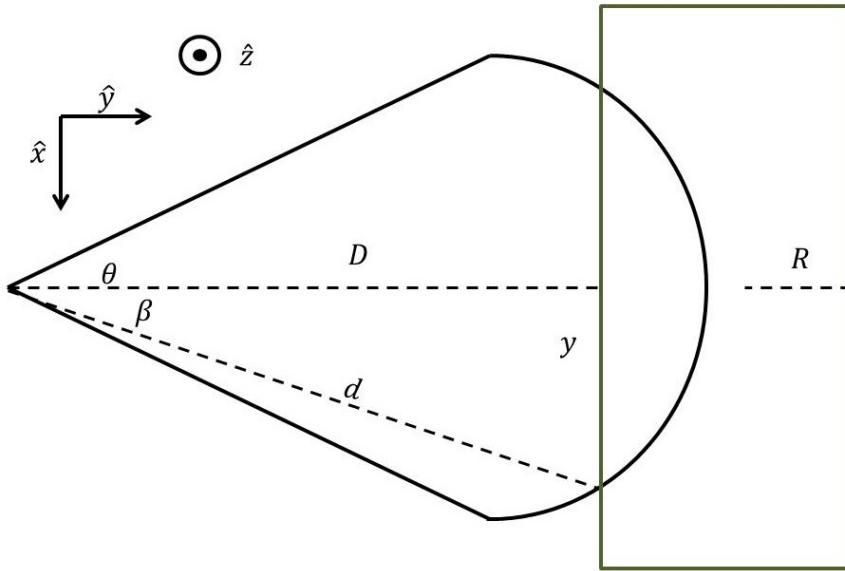


Figure 4.4.3: Top-view looking perpendicularly down on the scattering laser path, where it will look like a rectangle. Again, it intersects the spherical wavefront of the plume of spheres ejected in a cone.

the spherical cone's wavefront. We will call this fraction a “hit efficiency”, which we will denote ζ , where $0 \leq \zeta \leq 1$. To determine the value of ζ , we need to visualize the sphere's interaction with the wavefront. Namely, we need the surface area of intersection between the solid cylindrical beam and the a hollow cone's spherical wavefront as well as the total surface area of that wavefront. To do this, we consider Figures 4.4.2 and 4.4.3.²⁰ From these figures, we first note the total surface area of the cone's spherical wavefront, which is easiest found in Figure 4.4.3. This area is given by the area of a spherical cap, where $\theta = 20$ degrees in the figures. The formula for this area is

$$A_{total} = 2\pi d^2(1 - \cos(\theta)). \quad (4.4.11)$$

We now need to determine the area of intersection between the solid cylindrical beam and the hollow spherical cone wavefront. To do this, we see that we may parameterize the surface of the wavefront by the Cartesian equation for a sphere:

$$x^2 + y^2 + z^2 = d^2, \quad (4.4.12)$$

where $d = vt$. We may parameterize the beam by the Cartesian equation for a solid cylinder:

$$(y - B)^2 + z^2 \leq R^2, \quad (4.4.13)$$

²⁰Note that we choose an unconventional coordinate system for the development of this model in which neither the ablation nor the scattering beam is aligned with the z axis.

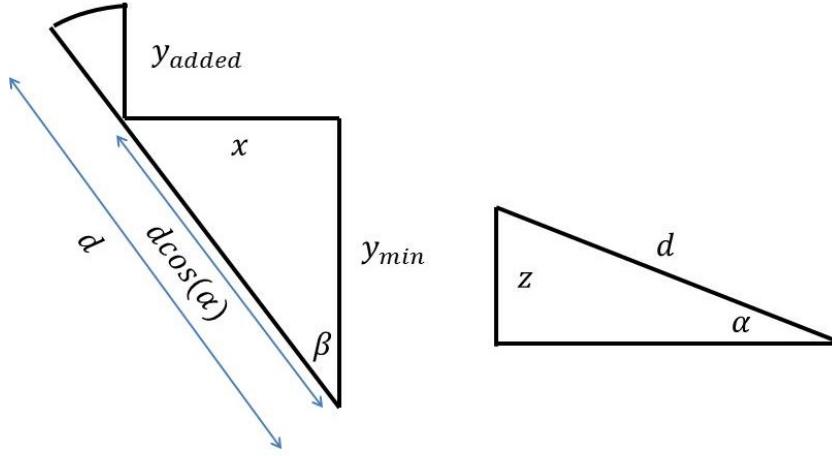


Figure 4.4.4: Geometry of the sphere-cylinder intersection, which relates all the variables from Figures 4.4.2 and 4.4.3. This is based on the three-dimensional nature of the intersection with the spherical wavefront of the plume and our cylindrical scattering beam.

where $B = D + R$. To find the smallest (nearest to $y = 0$) value of y , we subtract Equation 4.4.13 from Equation 4.4.12 to get

$$y_{min} = \frac{d^2 + 2RD + D^2 - x^2}{2(R + D)}. \quad (4.4.14)$$

Now, to find the maximum value of y at a given value of x , we note Figure 4.4.4.

From this, we see that if there is a z -component, then the angle that the radial vector to the sphere-cylinder edge intersection forms with the xy -plane is given by

$$\alpha = \arcsin \frac{z}{d}, \quad (4.4.15)$$

where d is the length of the radial vector. The component of it in the xy -plane is $d\cos(\alpha)$. Then, by Figure 4.4.4, the added distance to y_{min} to get y_{max} is $y_{added} = d(1 - \cos(\alpha))$. It follows that

$$y_{max} = y_{min} + d(1 - \cos(\alpha)). \quad (4.4.16)$$

Now, to find the minimum and maximum values of x , we note once again Figure 4.4.3. Namely, we have that

$$\beta = \arccos \frac{D}{d}. \quad (4.4.17)$$

Then, we immediately have that for a given value of d (determined by v and t),

$$x_{min} = -dsin(\beta), \quad (4.4.18)$$

and

$$x_{max} = d \sin(\beta). \quad (4.4.19)$$

Now that we have the domain over which the surface of intersection occurs in the xy -plane, we can solve for the actually area of this surface. We note the integrand for a surface area integration is $\sqrt{\left(\frac{\partial z}{\partial x}\right)^2 + \left(\frac{\partial z}{\partial y}\right)^2 + 1}$. We then just need to express z as a function of x and y . To do this, we use Equation 4.4.12 to obtain

$$z = \sqrt{d^2 - x^2 - y^2}. \quad (4.4.20)$$

Then, the integrand for the surface area of intersection integral will be given by using this equation for z in the integrand equation above to get

$$f(x, y) = \sqrt{\left(\frac{\partial z}{\partial x}\right)^2 + \left(\frac{\partial z}{\partial y}\right)^2 + 1} = \sqrt{\frac{d^2}{d^2 - x^2 - y^2}}. \quad (4.4.21)$$

Combining all of these results, we then have the following double integral that may be used to obtain a numerical value for the surface area of intersection:

$$A_{hit} = \int_{x_{min}}^{x_{max}} \int_{y_{min}}^{y_{max}} f(x, y) dy dx, \quad (4.4.22)$$

where all the limits of integration and the integrand are given by their appropriate equations derived in this section. This integral can be evaluated readily over the first variable of integration (y) to get the one-dimensional integral

$$A_{hit} = \int_{x_{min}}^{x_{max}} \left[d \arctan \frac{y}{\sqrt{d^2 - x^2 - y^2}} \right]_{y_{min}}^{y_{max}} dx. \quad (4.4.23)$$

We are now equipped with a method to calculate the “hit efficiency” mentioned above, which we here call ζ . Namely,

$$\zeta = \frac{A_{hit}}{A_{total}}, \quad (4.4.24)$$

where A_{hit} is given by a numerical integration of Equation 4.4.23 and A_{total} is given by Equation 4.4.11.

4.4.4 The three dimensional model

Correcting the one-dimensional model we first derived to account for the “hit efficiency”, we must partition the interval from v_{min} to v_{max} for a particular time into small subintervals of velocity length $d v$ so that d may be taken to be roughly

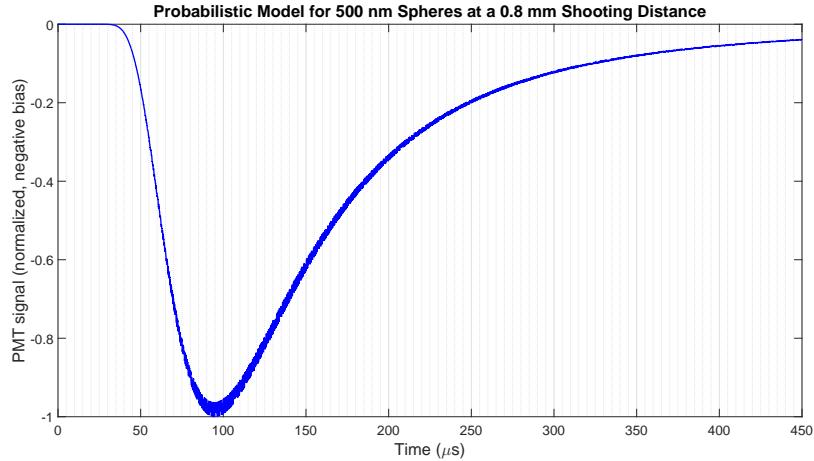


Figure 4.4.5: Sample result of the probabilistic model. This is for the same experimental parameters that were used in collecting time-of-flight data for Figure 3.2.1.

constant over that time in determining the actual value for ζ . Then, updating Equation 4.4.10, we obtain the three-dimensional version:

$$N(t) = N \sum_{k=1}^n \left[\zeta(v_k, t) \int_{v_k}^{v_{k+1}} f(v) d v \right], \quad (4.4.25)$$

where $v_{n+1} = v_{max}$, $v_1 = v_{min}$, n is the number of partitions and $\zeta(v_k, t)$ is the “hit efficiency” evaluated at velocity v_k and time t (such that $d = v_k t$), using Equation 4.4.24 as derived in the preceding section. With this equation, we can now numerically run a model to find the percent of spheres that would be hit by the scattering or high-intensity pulse as a function of time since ablation for various distances D between the sample and the beam edge.

4.4.5 Results

Figure 4.4.5 shows the result of running the probabilistic model with the same parameters as was used in taking the experimental time-of-flight data for Figure 3.2.1: 500nm spheres ablated 0.8mm from the scattering beam, shot with the 40W nominal power setting on the Q-switched laser. The most probable velocity used in the Maxwell-Boltzmann distribution for v_p in Equation 4.4.6 is 8.3 m/s based on Table 2. Again, we normalized for clarity and put in a negative bias to emulate lab traces.

More results from the probabilistic model are in Section 7, in addition to a comparison to the other models and lab data.

4.4.6 Error analysis

Most of the error analysis will come from the assumptions discussed in Section 4.3.

One source of error is introduced from having all the spheres begin at the same location in space. This is of course just a blatantly invalid assumption, but it is necessary in order to be able to create the probabilistic model with generating functions and expressions. Otherwise the math would be exceedingly difficult if not impossible without hundreds of piecewise functions. Having all the spheres start in the same location means some individual ones will be off by as much as about $50\text{ }\mu\text{m}$ along the x or y directions.²¹ This corresponds to the clearing radius Aaron found in his thesis [26]. However, the spheres generating the peak signal will be those between the center and about one to two standard deviations out, where the FWHM and standard deviation of a Gaussian are related by

$$FWHM = 2\sigma\sqrt{2\ln 2}. \quad (4.4.26)$$

The area and therefore the amount of spheres ablated scales as r since $dA = 2\pi r dr$. So since the laser intensity dies off Gaussian-like with radial distance from the ablation pulse center while area increases, the two balance to have the most significant spheres for the leading edge and peak of the time-of-flight signals coming from those between the center and about two standard deviations out. Spheres farther out than this will contribute to the trailing edge of the signal due to their much slower velocities. Hence, realistically, the significant spheres will be off by at most $20\text{ }\mu\text{m}$ in x or y initial position. Given that the scattering beam has a $100\text{ }\mu\text{m}$ radius, this change is nearly insignificant. For the high-intensity laser, however, with a $10\text{ }\mu\text{m}$ spherical spot size, this $20\text{ }\mu\text{m}$ discrepancy will be quite detrimental and the probabilistic model will not be trustworthy as an accurate model. For that, the Monte Carlo simulation will be the analysis tool to use with the first-principles model of Section 5 being the predictive tool.

Another error source is that, technically speaking, the probabilistic model calculates the percent of ejected spheres that will be hit by the scattering beam as a function of time since ablation. This is not exactly the same as the PMT signal, as mentioned in Section 3.1: a stronger PMT signal either means more spheres in the scattering beam or spheres closer to the center where there is greater light intensity. This discrepancy will be small though because the peak signal will more or less correspond to when the most spheres are in the scattering beam, especially when there are thousands of spheres ablated as is the case in these models, although it could be the source of a few microseconds discrepancy between the probabilistic and Monte Carlo models. This does give yet another way in which the probabilistic model will not work well with the high-intensity

²¹Note that we have switched back to a conventional coordinate system. So now y is vertical, z is along the ablation laser and x is along the scattering beam.

laser: with small numbers of spheres (on the order of one) in the focus, the signal will definitely correspond more to where individual spheres are relative to the center of the scattering beam rather than how many are in it.

Of course, with any numerical calculation, there will be some computational error. This is no doubt the culprit behind our fuzzy oscillations about the general curve, as seen in Figure 4.4.5. However, the general curve shape and time signatures stay identical beyond a certain number of partitions in each of the numerical calculations, indicating that a solution has been achieved. Thus, computational error is quite insignificant compared to the previous two errors.

Another major flaw in both the probabilistic and Monte Carlo models will be discussed shortly in Section 4.7.

4.5 Monte Carlo model

In the Monte Carlo model, many of the ideas are similar to the probabilistic model. As mentioned in Section 4.3 and the introduction to Section 4.4, the Monte Carlo model assigns initial values for the ejection velocity and angle of each sphere individually at random, weighted by a distribution. Then, the spheres' positions are updated as they plume outwards and are considered to produce a signal if they interact with the scattering beam.

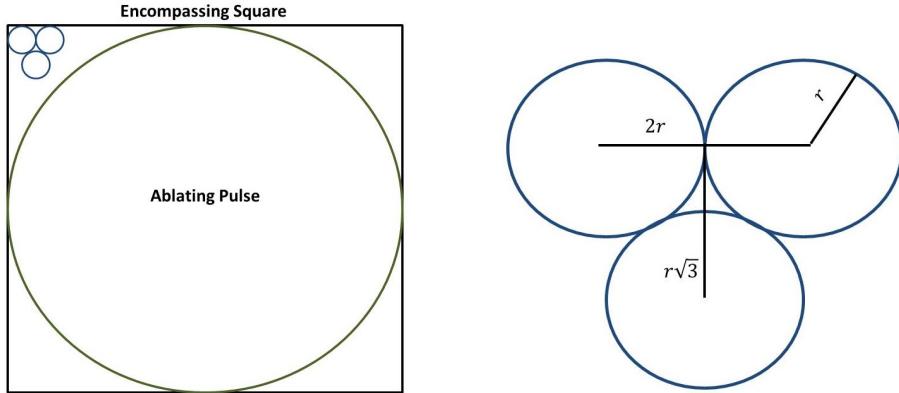
4.5.1 Accounting for HCP depositions

From the depositions, we see that the best target areas are monolayers of the polystyrene spheres which are sorted in a hexagonal-close packing (HCP) arrangement. This arrangement is the tightest possible packing of circles in a 2-D plane, as can be seen for example in Figure 2.2.1a. In order for the simulation to take this into account, there must be a method to determine the x and y positions of the spheres in the 2-D plane just above the silicon surface. In order to do this, we consider Figure 4.5.1a with a zoomed view of the three spheres in the upper left-hand corner, along with some labeled geometry shown in Figure 4.5.1b.

From this, we can construct an algorithm to assign the x and y positions of the spheres that are within the ablating pulse shot.²² Note that, taking the z coordinate of the silicon surface to be at zero, the z position of the center of all of the spheres on the silicon surface is r , the radius of the sphere. Our algorithm is as follows:

- (i) Start in the upper-left-hand corner of the square at the center position of the upper-leftmost sphere (circle in the diagrams), with a known x and y location.

²²We consider a sphere to be “in the shot” if its center is within the FWHM beam width of the ablating pulse about the center of the shot on the slide.



- (a) The picture behind the position-assignment algorithm, with three spheres included.
- (b) The geometry of an HCP packing arrangement. Note that r is the radius of an individual sphere.

Figure 4.5.1: Accounting for HCP in the Monte Carlo simulation.

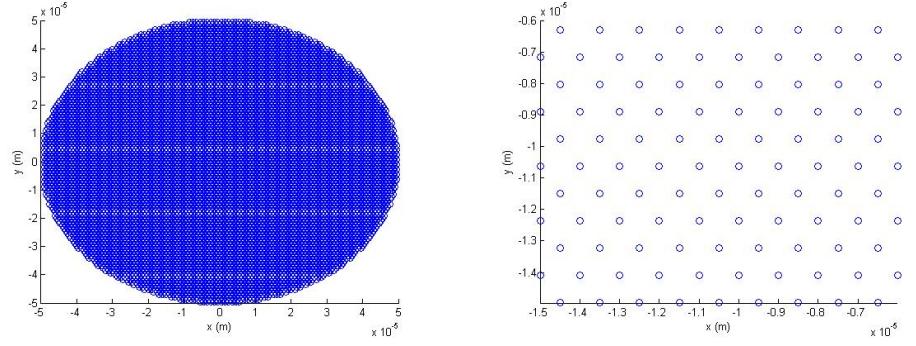
- (ii) Moving to the right in increments of $2r$ (increasing the value of x by $2r$), test to see if the spheres' x and y coordinates satisfy the condition $x^2 + y^2 \leq R^2$, where R is the radius of the ablating pulse²³. If the spheres' coordinates do satisfy this condition, then we record these values in an array.
- (iii) Once finished with a row, reset the x -coordinate to the original value, but change it by $\pm r$ as appropriate to account for the offset introduced by the HCP arrangement. Decrease the y -value by $\sqrt{3}r$. Then, repeat (ii).
- (iv) Once the bottom right-hand corner of the square has been reached, stop.

Using this algorithm, we obtained a scatter plot of the spheres' starting positions for those spheres that will be hit by the ablating pulse on the slide. This is shown in Figure 4.5.2a with a close-up shown in Figure 4.5.2b.

4.5.2 Results

Implementing the HCP algorithm for initial positions and choosing initial values for the ejection velocities and angles using the same distributions as the probabilistic model outlined in Section 4.3, the Monte Carlo simulation works by updating the positions at each timestep. Then, interactions with the scattering beam are simulated based on the sphere's position relative to the beam with a Gaussian intensity profile. From this, we produce a simulated PMT signal that we then normalize and put a negative bias on for consistency with the lab

²³We choose this radius based on the FWHM of the beam profile for the ablating laser at the focus spot of the silicon surface.



(a) The HCP arrangement reproduced by our algorithm. (b) A close-up of the HCP packing arrangement produced by our algorithm.

Figure 4.5.2: Results of the HCP packing algorithm.

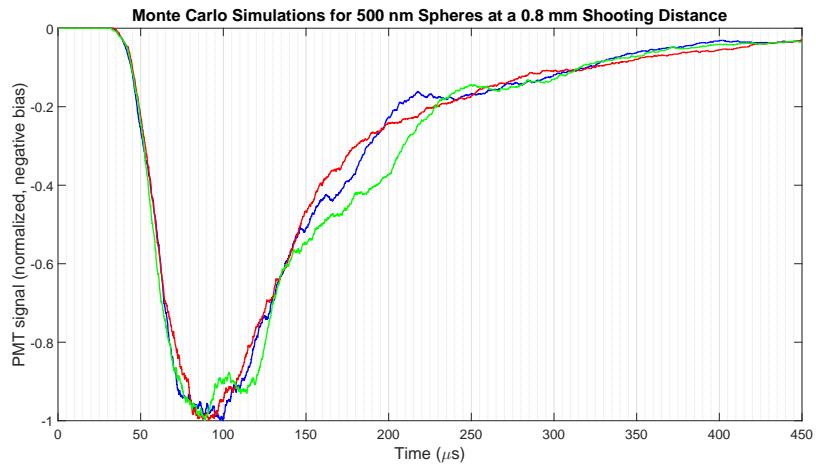


Figure 4.5.3: Sample results from the Monte Carlo model using the same parameters as for Figure 4.4.5 and Figure 3.2.1. A normalized PMT signal with a negative bias is plotted as a function of time since ablation, for three different runs of the code using the same initial conditions.

traces and the probabilistic model. Three sample Monte Carlo simulation traces are shown in Figure 4.5.3, again for the parameters used for the time-of-flight data in Figure 3.2.1, with the same most probable velocity of 8.3m/s used for Figure 4.4.5.

More Monte Carlo simulation results will appear and be discussed in Section 7. Figure 7.1.1 compares Monte Carlo simulation results to the probabilistic model output and experimental time-of-flight data.

4.5.3 Error analysis

As with the probabilistic model, there are several sources of error that must be considered in the Monte Carlo approach. Also in accordance with the probabilistic model, these mostly come from the assumptions made in Section 4.3.

First and foremost, this is a Monte Carlo model. So there is uncertainty and variation in runs with the exact same parameters. In the limit where the number of spheres ablated approaches infinity, such variation would go away, but with our hundreds to thousands of spheres ablated, this variation can be easily seen in Figure 4.5.3. However, there are two ways of putting this more positively. On the one hand, such variation actually makes it slightly more realistic than the deterministic, probability based model where there is a definitive result for a given set of parameters; in lab we have laser fluctuations, changes in the vacuum, non-uniformity of the depositions, and so on, all contributing to slight variations shot by shot in the same run. Further, we can mitigate any serious variations by averaging over a large number of Monte Carlo simulations. Indeed, as will be discussed later in Section 7, the Monte Carlo model results approach the probabilistic model when this is done over a large number of simulations, with surprisingly minor discrepancies.

Another source of error in the Monte Carlo model is the weighted random choices of initial values for the various parameters. This is only an issue because a sphere on the edge of the ablated area has the same probability to have a velocity of 10m/s, say, as a sphere at the center. Of course, this is just blatantly wrong. However, since we have not experimentally determined the ejection velocities as a function of distance from the center of the ablation pulse, this has to do for now.²⁴ From the large number of spheres ablated, this choosing of initial values from weighted distributions will not produce much error along the ϕ and θ angles, although in reality for an individual sphere these will undoubtedly depend on the sphere's initial position as well. Overall, then, from the statistics of large numbers, any error introduced here should be mitigated. The velocity issue will need to be considered as it may be damning for the Monte Carlo model's

²⁴Aaron performed a brief test to confirm that spheres near the edge are ejected with slower velocities, seen from a delay in the peak signal time when ablating an area overlapping a previously cleared section [26]. However, we did not fully flesh this out to get the ejection velocity as a function of radial distance.

accuracy when run with the high-intensity laser parameters.

Again, there is a flaw in both the probabilistic and Monte Carlo models discussed in Section 4.7.

4.6 The fitting mechanism

As mentioned earlier, these models can be used as a useful fitting mechanism for lab data, the time-of-flight traces. At the moment, only the Monte Carlo model is implemented as a fitting mechanism. Here are the steps to how it works:

- (1) Data is read in from a series of lab data traces for the same shot parameters from many shots on a single sample for a specific set of laser powers and a chosen ablation distance D . The sphere radius and the shooting distance are also read in.
- (2) The code first compiles the data traces together into a summed, normalized signal with a negative bias, of course as a function of time since ablation.
- (3) The program then runs through many simulations over a range of possible most probable ejection velocities. It runs a set number of times for each velocity and produces a summed, then normalized overall signal for each velocity to mitigate any Monte Carlo error.
- (4) A least squares fit is performed comparing the simulated traces to the actual data, coming up with a best fit velocity and determining the summed square error between the simulation and experimental trace.

This fitting mechanism is implemented in a Matlab graphical-user-interface (GUI). An example implementation of this GUI is shown in Figure 4.6.1, with the data we have looked at thus far in Figure 3.2.1 and computed the probabilistic and simulation models for. In this particular case, the fit doesn't perform too well since the data is actually not that great. The batch of time-of-flight data in Figure 3.2.1 is displayed because it is unusual with a number of features. Most of our data looks more like the probabilistic model results with sharp leading edges and a smooth tail, as in Figure 4.4.5. An interesting thing to note is that the best fit velocity here of 10 m/s is larger than the 8.3 m/s ejection velocity quoted by Aaron in Table 2 for 500 nm spheres. However, it does not exceed the velocity added with the uncertainty.

With this GUI, the user need only open it up, enter in the sphere radius, ablation distance, and the last two digits of the first and last .DAT file numbers. Then, he or she only has to click the “Analyze DAT File Data” radio button (definitely no humor intended). The program then spits back the best fit velocity and summed square error, in addition to showing time-of-flight plots for the original

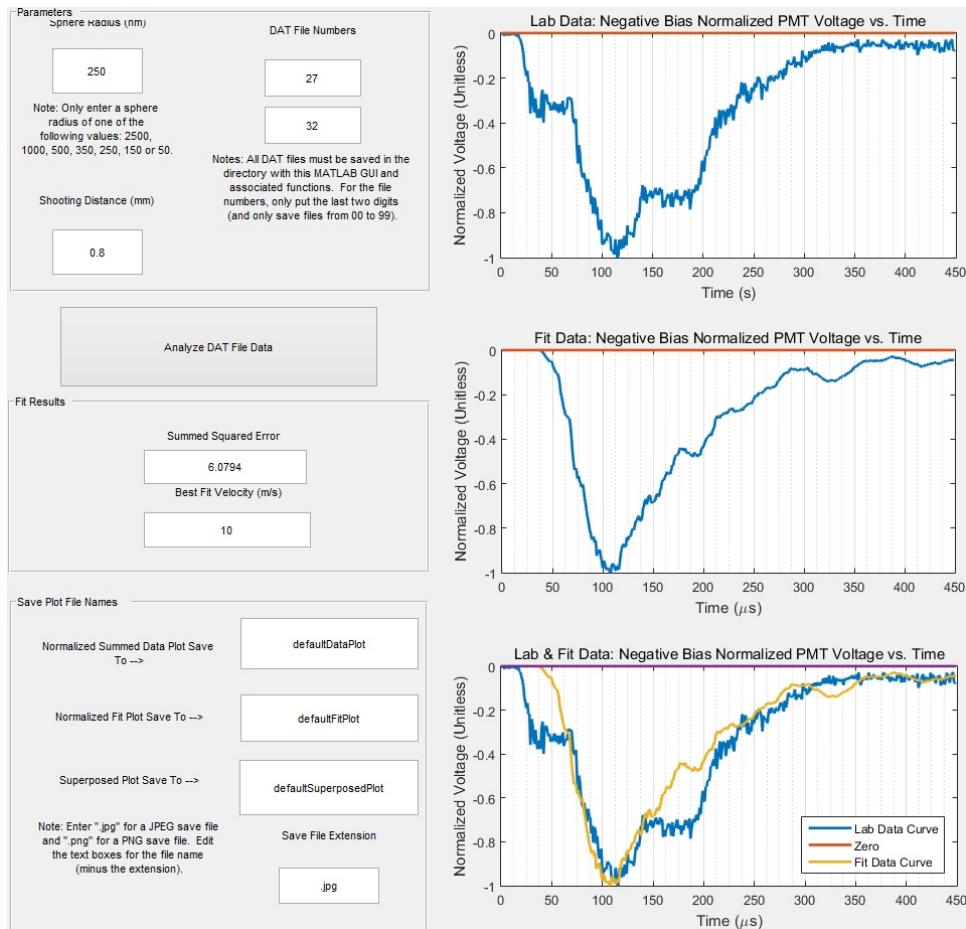


Figure 4.6.1: Image of the GUI after using the data from Figure 4.4.5. Note that it is organized such that all the plots are on the right side while data and information is on the left. On this left side, the user inputs are under the “Parameters” section in the upper part with the radio button for compiling just beneath. Under that is the results of fitting to the Monte Carlo simulation. In the bottom left are the file names and the extension that can be adjusted for saving plots. Note that the “Zero” curve on the “Lab & Fit Data” graph is irrelevant; it was merely required to get a certain GUI plot feature to work.

data, the best fit simulation, and the two superposed. These can be saved to default file names seen in the GUI that can be modified by the user, along with the file type.²⁵

4.7 Weaknesses and the need for a better model

Clearly, from Sections 4.4.6 and 4.5.3, there are flaws to both the probabilistic and Monte Carlo models. The ones outlined there are somewhat negligible for consideration with the Mudd time-of-flight data analysis given the huge size of the scattering beam. However, running with high-intensity laser parameters, these considerations pretty much rule out the probabilistic model and give great hesitation to the Monte Carlo model. The biggest reason for this is that both models effectively ignore the ejection mechanism and just make assumptions about what the result will be and how that translates to the spheres' ejection characteristics. While these assumptions seem decent given the experimental results found by Andrew [32] and Aaron [26], they instilled a lingering desire to understand this mechanism better.

The main flaw with both the probabilistic and Monte Carlo models is that they require an input: the most probable ejection velocity.²⁶ Neither model is by any means a first-principles physics calculation. Rather, this flaw makes them perfect for a data analysis tool, but completely useless as a predictor of the time-of-flight characteristics, unless keeping the same ablation laser settings and adjusting the shot distance D . This should be taken with some emphasis: the models are useless in the standalone sense that they cannot predict anything with respect to time-of-flight or target delivery without experimental data to use as a backbone, much like many stock trading algorithms which have to be careful to distinguish between in-sample and out-of-sample significance. See Campbell and Thompson's paper for an example of this in stock trading [7]. However, these models are extremely useful for analyzing and extending lab data, as outlined in Section 4.6 and shown in Figure 4.6.1, for analysis specifically.

Nonetheless, to fully understand the ejection mechanism and, more importantly, to assess our laser ejection scheme's viability for delivering targets to a high-intensity laser, we need to improve on what we have. For this, we turn to the basics. Namely, we seek a first-principles physical model of laser ejection that has zero free parameters. Its only inputs are the operating specifications: the silicon thickness or the ablation pulse energy and FWHM focal spot size, for example. This is the motivation for the next section in this thesis.

²⁵The default save file type is .jpg, but if the user wishes to edit the figures, for example, he or she could change this to .fig such that it is saved as a Matlab figure with all the tunable properties. Also, an image of the GUI itself can be saved, which defaults to "RunningGUI.eps".

²⁶This is for use in the Maxwell-Boltzmann distribution, and is assumed to be the velocity that Aaron found in his thesis based on the time-of-flight analysis since this velocity is the one the bulk of spheres would need to be ejected with to produce the largest signal collectively [26].

5 The better model: zero free parameters

Building a better model, we will retain some elements of the models already developed, specifically the Monte Carlo simulation. Namely, we will still need to propagate spheres forward in time and simulate their interactions with either the scattering beam or the high-intensity laser.²⁷ But the key will be trying to eliminate assumptions and any free parameters to get as close to an accurate model as possible for our laser ejection mechanism using a pulsed ablation laser incident on our polystyrene-coated silicon substrates. Specifically, we seek to eliminate the one free parameter of the most probable ejection velocity. To do so, we will look at the interactions of light with material. The basic idea is that the silicon substrate will be heated by the ablation pulse, causing the surface to expand outwards and push the spheres off with some ejection velocity that will vary based on how the surface expands at different radial positions.

5.1 Assumptions

Unfortunately, as with any model, some assumptions will still be made, though they are not as basic as some of the assumptions made in developing the Monte Carlo and probabilistic models, outlined in Section 4.3. Here are the assumptions we must make to develop a first-principles model:

- (1) The ablation pulse is a Gaussian both along the radial axis in the $x-y$ plane and along its temporal duration, which corresponds to a spatial width along the z axis. This is the standard assumption for pulsed lasers. Indeed, using the improved focal imaging tools from Section 3.3, we find that the intensity distribution along the radial axis is indeed a Gaussian when the Q-switch is operated at lower intensities at a high rep rate. These conditions are necessary to be able to capture a proper image (mainly to not fry the camera), but the same result will apply for higher powers and low rep rates by simple scaling.
- (2) Most of the various material properties of silicon and polystyrene are constants for the duration of the ablation interaction. This is of course not a valid assumption as many of these “constants” have temperature dependencies, for example. However, when doing preliminary testing as the model was developed, it was determined that many of these dependencies do not significantly affect the values over the range of temperature fluctuations observed with our ablation laser parameters. Furthermore, some dependencies negated each other (in the sense that in an xy multiplication, if x doubles and y halves, the value remains). Of course, the

²⁷By simulate a sphere's interaction with the high-intensity laser, we mean determine whether or not a given sphere would be in the spherical focal spot of that laser, not the stochastic heating effects. For that simulation, see Laura's thesis [33].

model could be marginally improved if the material properties' dependencies are included, but this requires a significant sacrifice to the computation time. Nonetheless, it could be done and would be a potentially valuable endeavor if this code becomes heavily relied upon in the group's future efforts.

- (3) The laser loses negligible intensity from its interactions with the polystyrene spheres. We do account for loss from reflection. So this just means we do not account for any absorptive properties of the polystyrene or interactions with the meniscus or other things bonding the spheres to their silicon substrate. Such forces have been studied extensively by the group in the past and even currently. See the theses of Andrew [32], Aaron [26] and Amber [6] for a look into this characteristic of the deposition process.

For our purposes, we basically assume that the energy required to overcome any such bonding forces between the spheres and their substrate is negligible. Essentially, we take the spheres to be floating on the surface of the silicon such that their ejection velocity will be determined by whatever speed the surface is expanding outwards at before its acceleration decreases. That is, as long as the surface expansion velocity increases, the spheres remain pressed onto the surface, but as soon as the acceleration becomes negative, the spheres eject with whatever velocity they are moving at during that moment.

To consider whether this is a valid assumption, we turn to Brendan's estimates of the adhesion forces between the polystyrene nanospheres and the silicon substrate: he estimated that the largest force would be the capillary force, which would be approximately 2×10^{-6} N for the $5\mu\text{m}$ sphere size [12]. This will be returned to in Section 6.3.

5.2 Fresnel coefficients

With a pulsed laser dumping energy into a silicon substrate, there are several effects we must account for. The two most important when considering just the raw energy transfer will be loss at the interfaces due to reflection and refraction, and absorption of heat into the silicon. To quantify the former, we will turn to the Fresnel coefficients with a brief introduction via Snell's Law.

Snell's law relates the incident θ_i and transmitted θ_t angles of an electromagnetic wave propagating from a material of refractive index n_1 to one with refractive index n_2 , where the angles are measured relative to the normal of the interface between the mediums.²⁸ The specific relation is given by Equation 5.2.1, with a visualization of the setup shown in Figure 5.2.1.

²⁸Incidentally, the reflected angle θ_r is equal to the incident angle θ_i .

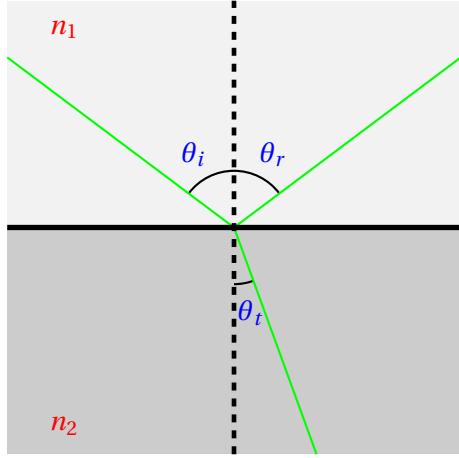


Figure 5.2.1: A incident light ray from quadrant two in a medium with index of refraction n_1 hits an interface with a medium of index of refraction n_2 , causing some of the light to reflect into quadrant one and some to refract into quadrant four.

$$n_1 \sin \theta_i = n_2 \sin \theta_t \quad (5.2.1)$$

With the relation between θ_i and θ_t now given by Equation 5.2.1, we can turn to the Fresnel coefficients. These give the fraction of the incident intensity that is reflected and the fraction that is reflected as a function of these angles. Further, they depend on whether the light is polarized parallel or perpendicular to the plane of the interface. Calling these the p and s polarizations, respectively, the reflectance is given by

$$R_s = \left| \frac{n_1 \cos \theta_i - n_2 \cos \theta_t}{n_1 \cos \theta_i + n_2 \cos \theta_t} \right|^2; \quad (5.2.2)$$

$$R_p = \left| \frac{n_1 \cos \theta_t - n_2 \cos \theta_i}{n_1 \cos \theta_t + n_2 \cos \theta_i} \right|^2. \quad (5.2.3)$$

By conservation of energy, the transmittance can then simply be determined by

$$T_s = 1 - R_s; \quad (5.2.4)$$

$$T_p = 1 - R_p. \quad (5.2.5)$$

In our setup, the incident light will be at $\theta_i = 0$ since the silicon will be taken as perpendicular to the ablation laser's axis of propagation. Further, we will take the ablation laser as an unpolarized light source, in which case the reflectance and transmittance are the averages of their respective s and p polarized values.

When ablating in a high vacuum, $n_1 \approx 1$ and we will let $n_2 = n$ for clarity throughout the rest of the model, with n now representing the refractive index of the silicon substrate. With these assumptions and simplifications, the fraction of the intensity transmitted through the interface into the silicon is

$$\mathcal{T} = \frac{4n}{1 + 2n + n^2}. \quad (5.2.6)$$

This will be used as a scaling factor on the incident intensity of the pulsed ablation laser to determine the energy that remains to be absorbed by the laser in the form of heat.

5.3 The ablation laser

In our experiments, the ablating laser is regrettably not an ideal plane wave of uniform electromagnetic radiation. Rather, it will have both temporal and spatial dependence. For the purposes of the model, we will assume that both of these dependencies are Gaussian, though it is extremely unlikely that such an ideal case could ever be achieved in a lab.²⁹ With these assumptions in mind, we proceed to write down an equation for the intensity of the laser beam as a function of a distance z along its propagating axis, the radius r from that axis and time t :

$$I(r, z, t) = I_0 \frac{1}{\sigma_r \sqrt{2\pi}} \frac{1}{\sigma_z \sqrt{2\pi}} e^{-\frac{(z-z_{avg})^2}{2\sigma_z^2}} e^{-\frac{r^2}{2\sigma_r^2}}, \quad (5.3.1)$$

where I_0 is the intensity of the laser at the focus and center of the beam. The other parameters in the equation are as follows: $\sigma_r = \frac{FWHM}{2\sqrt{2\log 2}}$ is the standard deviation along the radial axis with $FWHM$ measured from the ablation laser, σ_z is the standard deviation along the z axis and z_{avg} is the z axis position of the center of the laser.³⁰ A very important thing to note here is that z_{avg} has time dependence, which is why $I = I(r, z, t)$ in Equation 5.3.1. Also, σ_z is determined from the speed of light and the temporal pulse duration t_p of the laser. In the code, we will set the a time-delay of t_d for clarity until the laser arrives at the surface at an initial distance \mathcal{Z} , from the $\mathcal{Z} = 0$ reference of the back side of the

²⁹It is a very good approximation, however, as noted in Section 5.1. This is because the laser exits the laser cavity as a flattop beam, but it then immediately subjected to numerous imaging optics, which will tend to diverge this flattop profile into a more Gaussian beam.

³⁰This may look invalid as it appears to have units of intensity over distance squared, but when doing the actual code implementation, all distances are made unitless by dividing by appropriate parameters. For example, we divide both r and σ_r by the maximum value of r that we consider, R , to map onto a unitless (-1,1) scale.

silicon substrate.³¹ After this,

$$z_{min} = z_0 - v(t - t_d); \quad (5.3.2)$$

$$z_{max} = z_{min} + vt_p; \quad (5.3.3)$$

$$z_{avg} = \frac{z_{max} + z_{min}}{2}, \quad (5.3.4)$$

where v is the speed of the light in silicon.³²

Equation 5.3.1 may look somewhat unusual, but it is really just making one assumption to differ from the usual intensity profile of a Gaussian beam. Namely, we assume that the laser does not diverge much in its beam waist over the thickness of the silicon. We can justify this quantitatively with the Rayleigh range of our laser. Namely, the z propagation distance from the focus when the beam waist has increased by a factor of $\sqrt{2}$ is denoted z_R and is given by

$$z_R = \frac{\pi w_0^2}{\lambda}, \quad (5.3.5)$$

where w_0 is the beam waist at the laser focus and λ is the wavelength of the laser. Plugging in the values for our green ablation laser, $z_R \approx 10\text{mm}$, which is between two and three orders of magnitude larger than our silicon substrates whose thicknesses are on the order of hundreds of microns, and whose skin depth is an order of magnitude smaller than that. Hence, the laser will essentially retain a constant beam waist and therefore a constant σ_r over the z propagation distance we are concerned with for absorption.

5.4 Absorption

With Equation 5.2.6, we know what percent of the incident laser intensity is transmitted into the silicon wafer to be absorbed.

When traveling through an absorptive material, the intensity of an electromagnetic wave falls off exponentially with distance. The absorption coefficient α is related to the extinction coefficient κ of the complex index of refraction $\underline{n} = n + i\kappa$ by $\alpha = \frac{4\pi\kappa}{\lambda}$. For a given wavelength λ , this absorption coefficient becomes approximately constant and the intensity as a function of depth z in the material is given by

$$I(z) = I_0 e^{-\alpha z}. \quad (5.4.1)$$

³¹We use two coordinate systems here. One is the standard z axis along the ablation laser propagation direction with $z = 0$ at the focus initially at the substrate surface. The other is a dimension convenient for measuring silicon thickness, which is \mathcal{Z} .

³²This may come across as sketchy, but we are really only concerned with these z positions in regards to how much intensity will be absorbed by the laser in each infinitesimal volume in the silicon. Hence, we need to know the light's position as if all space were silicon, provided that we still have the light just becoming incident on the silicon substrate at time $t = t_d$.

The intensity that is “lost” in each interval $(z, z + dz)$ can be considered heat (per unit area per unit time) energy to be absorbed by the atoms in the silicon lattice. On the microscopic scale, this will induce electron excitations and increased atomic energies, which will lengthen bonds in the lattice structure. On the macroscopic scale, these atomic energy and physical shifts translate to an increase in temperature and thickness of the silicon substrate. This increase in temperature is characterized by the heat capacity C , which relates infinitesimal temperature changes dT to small amounts of heat energy δQ :

$$C = \frac{\delta Q}{dT}. \quad (5.4.2)$$

The heat capacity itself relates to properties of the silicon wafer. It is often normalized to the mass via the specific heat capacity, sometimes shortened to just specific heat. In all of our laboratory setups, we will have the ablations occurring in near vacuum, so

$$c_p = \frac{C}{\rho V}, \quad (5.4.3)$$

where ρ is the density and V the volume of the material being heated.

5.5 Additional effects: diffusion and Boltzmann radiation

The heat equation is a partial differential equation that governs how the temperature in a material evolves over time. It relates the temporal and spatial flows via the thermal diffusivity D :

$$\frac{\partial u}{\partial t} = D \nabla^2 u \quad (5.5.1)$$

$$= D \left(\frac{\partial^2 u}{\partial x^2} + \frac{\partial^2 u}{\partial y^2} + \frac{\partial^2 u}{\partial z^2} \right). \quad (5.5.2)$$

The thermal diffusivity looks like a constant in this equation, but in a dynamical model such as the one being developed, it is not necessarily so. It is, however, related to material properties by $D = \frac{k}{c_p \rho}$, where k is the thermal conductivity.

While Equation 5.5.2 is the primary source of temperature changes in the material over time, there is also a loss term that will likely be insignificant in scale, but that we will include anyways. This term is Stefan-Boltzmann radiation, which is present regardless of pressure as is evident in astrophysics. It describes the total power radiated from an object across all wavelengths in relation to the temperature and surface area of the object via

$$P = \epsilon A \sigma T^4, \quad (5.5.3)$$

where A is the total surface area, P is the total power radiated, T is the temperature, $\sigma = \frac{2\pi^5 k^4}{15 c^2 h^3} \approx 5.67 \times 10^{-8} \text{ W m}^{-2} \text{ K}^{-4}$ is the Stefan-Boltzmann constant and ϵ

is the emissivity of the object. Note that emissivity is not to be confused with the electrodynamic term, but is rather a measure of how “perfect” of a blackbody the object is on a 0 to 1 scale, with 1 signifying a “perfect” blackbody, an ideal material that absorbs all incident electromagnetic radiation.

5.6 Expansion

The key to laser ablation is the ejection of the particles from the substrate with some non-trivial ejection velocity at an angle reasonably close to the normal of the surface. The latter condition has been discussed in Section 4, even though it will be partially accounted for in the model itself later in Section 6.2. The former is the thrust of our model.

Expansion of materials in response to temperature changes is a well-known and studied effect. For a material of length L and coefficient of linear thermal expansion α_L , the change in length dL relates to a change in temperature dT by

$$\alpha_L = \frac{1}{L} \frac{dL}{dT}. \quad (5.6.1)$$

This could get fairly complicated if α_L depends on temperature, but again over the ranges observed in the preliminary testing, this coefficient remains roughly constant. See Glasov and Pashinkin’s paper for details about the temperature dependencies of α_L and the specific heat c_p of silicon [13].

At this point, one could also question the use of the linear expansion equation in the model. This is a valid concern considering the three-dimensional nature of the silicon media that we are working with. However, in the case of our specific depositions, we cut out silicon slides that are 3 mm × 20 mm on the deposition/ablating surface and are roughly 500 μm thick. We can see that the deposition area dimensions are roughly two to three orders of magnitude larger than the thickness dimension. Moreover, most of the energy will initially be deposited in a very shallow depth into the silicon of somewhere between one and ten microns into the surface, as per Green’s paper [15]. Considering the rigidity of a silicon lattice versus the comparatively enticing low pressure of our near vacuum, the silicon will use the heat energy to expand outward rather than flexing the silicon lattice. That is, it is easier to stretch the bonds in the lattice than it is to compress them given an energy flux.

In the model, then, we will consider the linear expansion of the silicon outward into the vacuum rather than a volumetric expansion.

5.7 Modified finite difference method

Combining everything discussed thus far in this section, a mathematical formulation can be developed for our physical model, as will be displayed in Section 5.8. These are very nonlinear equations and any attempt to solve them analytically would be a futile effort. As mentioned in Section 1.3, Andrew solved for

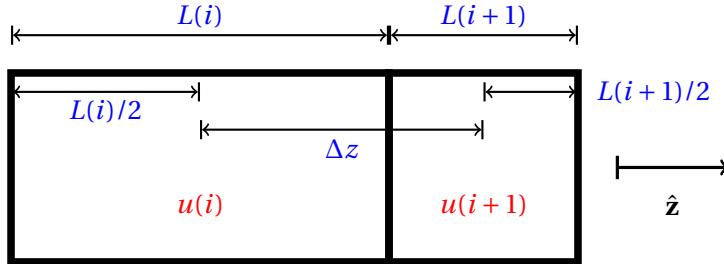


Figure 5.7.1: Diagram for computing a one-sided first-order numerical derivative. Shown are two consecutive boxes in a column along the z axis, i and $i + 1$, with exaggerated length differences $L(i)$ and $L(i + 1)$. $u(i)$ and $u(i + 1)$ are their respective temperatures.

analytical solutions to the heat equation under a set of conditions and assumptions that are far more restrictive with many more approximations than we felt comfortable with for trying to generate a first-principles model [32]. Nonetheless, he did produce an estimate for the expansion timescale of the silicon substrate, which was his main purpose.

For our purposes, we seek a full solution considering all of the pertinent effects. But we must solve this numerically. To do so, we implement a modified version of the finite element method. This may scream incompetency to numerical physicists everywhere, but we found it to work quite well with minimal error, as discussed in Section 5.11. For a thorough analysis of numerical methods, see Butcher's text with emphasis on solving differential equations [5].

The basic idea of the finite difference method arises from the fact that taking derivatives is difficult when there is no analytical function to take the derivative of. It requires discretizing both space and time. Hence, we must partition our universe into a grid, assigning values to each box in the grid. Luckily, from angular symmetry with a Gaussian ablation pulse, we need only consider two physical dimensions, the radial distance in the $x - y$ plane from the ablation pulse center and the z axis along the laser propagation direction. This creates a 2D array of boxes, each of which we assign a temperature and a length along the z axis. The cross-sectional area dA of the boxes will not change given the considerations in Section 5.6. Using the governing equations, the length and temperature of each box are updated at every timestep. But in order to use the governing equations, numerical derivatives must be calculated. Here is where the usual finite difference method is modified: we take the derivative with respect to a changing length dimension. That is, in computing $\frac{du}{dz}$, even if Δu is the same between consecutive timesteps, the length Δz may change, resulting in a differential numerical derivative. To compute our numerical derivatives in this modified finite difference method, then, consider Figure 5.7.1. From this, we compute the right-

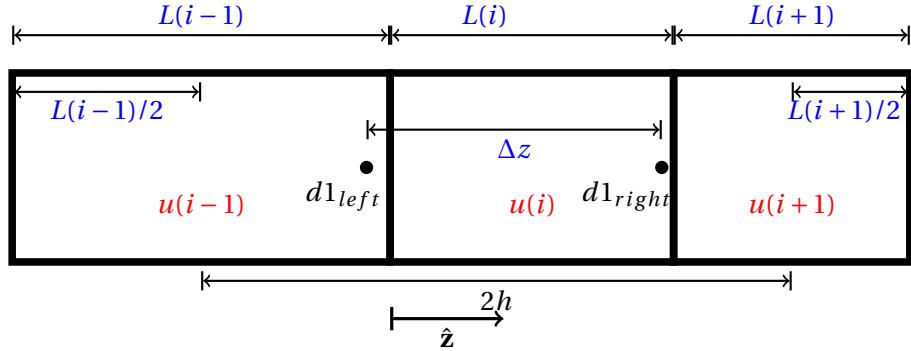


Figure 5.7.2: Diagram for computing the first-order and second-order numerical derivatives. The sites marked $d1_{left}$ and $d1_{right}$ are where the left- and right-sided first derivatives are calculated.

sided numerical derivative at box i in a z axis column to be

$$\begin{aligned} \frac{du}{dz}_{right} &= \frac{u(i+1) - u(i)}{\Delta z} \\ &= 2 \frac{u(i+1) - u(i)}{L(i) + L(i+1)}. \end{aligned} \quad (5.7.1)$$

This and its corresponding equation for a left-sided derivative are needed for our governing equations since they involve a second-order derivative in spatial dimensions. For a first-order derivative, it can be computed using idea behind one of the limit definitions. Namely, for a function $f(x)$,

$$\frac{df}{dx} = \frac{f(x+h) - f(x-h)}{2h}, \quad (5.7.2)$$

where the analytical derivative is computed taking $h \rightarrow 0$. Considering Figure 5.7.2, we find that the distance $2h$ between where $f(x-h)$ and $f(x+h)$ are sampled is really $L(i) + (L(i+1) + L(i-1))/2$. Then, the first order numerical derivative is

$$\frac{du}{dz} = 2 \frac{u(i+1) - u(i-1)}{L(i-1) + 2L(i) + L(i+1)}. \quad (5.7.3)$$

Similarly, considering again Figures 5.7.1 and 5.7.2 in conjunction with Equation 5.7.1, we can compute the numerical second derivative as a difference in the right-sided and left-sided first derivatives divided by the distance between where these values are computed. This yields

$$\begin{aligned} \frac{d^2u}{dz^2} &= \left(2 \frac{u(i+1) - u(i)}{L(i) + L(i+1)} - 2 \frac{u(i) - u(i-1)}{L(i-1) + L(i)} \right) / \Delta z \\ &= 8 \left(\frac{u(i+1) - u(i)}{L(i) + L(i+1)} - \frac{u(i) - u(i-1)}{L(i-1) + L(i)} \right) / (L(i-1) + 2L(i) + L(i+1)), \end{aligned} \quad (5.7.4)$$

since $\Delta z = (L(i-1) + 2L(i) + L(i+1))/4$.

5.8 The governing coupled partial differential equations

We now generate a mathematical formulation for the model. Essentially, we wish to solve coupled partial differential equations governing the temperature and length of the silicon substrate. The latter corresponds to the surface expansion, which can be differentiated with respect to time to determine the surface expansion velocity. As mentioned in Section 5.1, this will determine the ejection velocity of our spheres as a function of radial distance from the center of ablation pulse.

Putting everything in terms of the fundamental material properties of silicon, we can express our model mathematically with N partitions along the z axis. We let \mathcal{Z} denote the silicon thickness. We also have M partitions along the radial axis in the $x - y$ plane. Then the temperature and z lengths look something like $u(r, z, t) = u(m, n, t)$ and $L(m, n, t)$, where $m \in (1, M)$ and $n \in (1, N)$ and t is the usual time, which is also discretized. The silicon thickness will vary as a function of radial distance, so we will have $\mathcal{Z}(r, t) = \mathcal{Z}(m, t)$.

First, the initial conditions (ICs) are given:

$$\begin{aligned} u(m, n, 0) &= T_{amb}; \\ L(m, n, 0) &= \frac{Z_0}{N}; \\ \mathcal{Z}(m, 0) &= Z_0. \end{aligned} \tag{5.8.1}$$

The governing coupled partial differential equations are given next:

$$\begin{aligned} \frac{\partial u(m, n, t)}{\partial t} &= \frac{kZ_0}{c_p N \rho_0 L(m, n, t)} \nabla^2 u(m, n, t) \\ &\quad + \frac{4n}{1+2n+n^2} \frac{E_p}{t_p A_p} [H(t) - H(t - t_p)] \times \\ &\quad \frac{Z_0}{c_p L^2(m, n, t) N \rho_0} \left[e^{-\alpha \sum_{j=1}^{n-1} L(m, j, t)} (1 - e^{-\alpha L(m, n, t)}) \right]; \\ \frac{dL(m, n, t)}{dt} &= L(m, n, t) \alpha_L \frac{du(m, n, t)}{dt}. \end{aligned} \tag{5.8.2}$$

In this pair of coupled partial differential equations, it is very clearly nonlinear. There is a Laplacian for the heat equation term along with two Heaviside step functions and even an exponential of a summation. For the boundary term where $n = 1$, there is an additional term

$$\frac{-A\epsilon\sigma Z_0 u^4(m, 1, t)}{c_p N \rho_0 L^2(m, 1, t)} \tag{5.8.3}$$

that is added into the partial differential equation for $\partial u(m, 1, t)/\partial t$.

With these governing equations, we wish to solve for $\partial \mathcal{Z}(m, t)/\partial t$ so that we may determine at what time and with what speed the spheres are ejected from

Name	Variable	Value	Dependencies
Pulse Duration	t_p	20 ns	N/A
Index of Refraction	n	3.42	λ
Density of Silicon	ρ	2330 kg/m ³	T
Ablation Laser Wavelength	λ	532 nm	N/A
Ablation Area	A_p	$1.77 \times 10^{-8} \text{ m}^2$	N/A
Total Ablation Pulse Energy	E_p	$3.25 \times 10^{-4} \text{ J}$	N/A
Full-Width-Half-Max	$FWHM$	24 μm	N/A
Linear Expansion Coeff	α_L	$4.86 \times 10^{-6} \text{ K}^{-1}$	T
Thermal Conductivity	k	131 W m ⁻¹ K ⁻¹	T
Emissivity	ϵ	0.5	λ, T
Specific Heat of Silicon	c_p	700 J kg ⁻¹ K ⁻¹	T
Initial Silicon Thickness	z_0	500 μm	N/A
Ambient Temperature	T_{amb}	294.5 K	N/A

Table 3: Values and dependencies of various “constants” used in the model.

our expanding silicon substrate. Here, $\mathcal{Z}(m, t) = \sum_{n=1}^N L(m, n, t)$. As mentioned in Section 5.1, when the acceleration (the derivative of $\partial \mathcal{Z}(m, t)/\partial t$) turns negative for the first time, the spheres eject with whatever velocity $\partial \mathcal{Z}(m, t_{eject})/\partial t$ the surface is expanding at the ejection time t_{eject} .³³

5.9 Values used

This section is intended to provide full information on the values of the “constants” used in the model and to give context for all the terms in Equation 5.8.2. They are listed in Table 3.

Note that in certain cases, such as for the “Density of Silicon” constant, the value listed pertains only to the initial density of the silicon in vacuum at the ambient vacuum temperature. Another key to understanding this table is a brief explanation of the “Dependencies” column: this column describes what other parameters the listed “constant” depends on. For example, the index of refrac-

³³This is only strictly true if the acceleration stays negative once it switches. Fortunately, from Figure 5.10.1, we see this is the case for laser ejection.

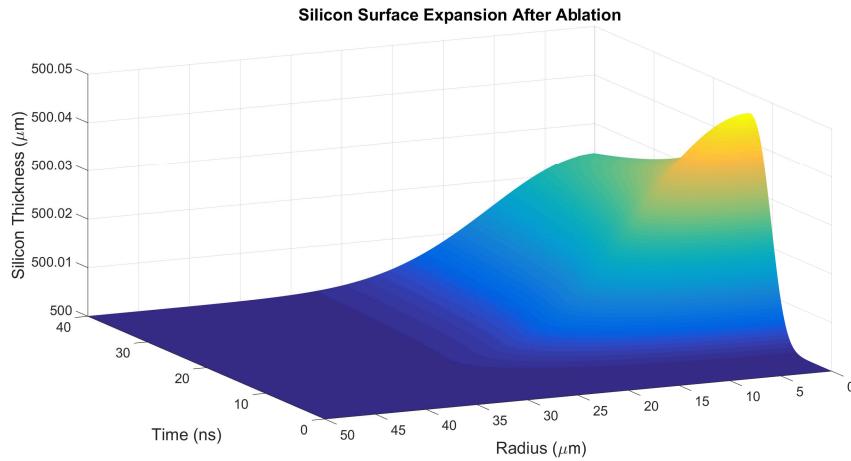


Figure 5.10.1: Silicon surface expansion. The silicon thickness is plotted as a function of time and radial position from the ablation pulse's center. “Hotter” colors indicate greater thickness, as does height.

tion n is dependent on the wavelength of light that is incident upon the material. In such cases where one constant depends on (potentially many) other parameters, I have quoted the relevant value given the other values. Many of the constants have other dependencies such as on temperature. Again using the index of refraction n , there is a definite temperature dependence and temperature will definitely not remain constant throughout the laser heating of the silicon substrate. However, n will stay within a percentage point or so of the quoted value over the temperature ranges observed in preliminary runs of the model.³⁴

5.10 Results

Running our model code with values corresponding to our ablation laser, we find the surface expansion characteristics of our silicon substrate when ablated with our laser. Again, the parameters are easily adjustable to any system knowing just a few characteristics of the ablation laser. For our system, the surface expansion of the silicon is visualized in Figure 5.10.1. In this mesh plot of the surface expansion, “hotter” colors indicates greater thickness, or larger outwards expansion in other words. We can see that in general with time, the surface expands before reaching some peak thickness and relaxing due to cooling and diffusion. From

³⁴As mentioned already, analyzing such preliminary runs during the development of the code was fundamental to determining which parameters could be approximated as constant and what other simplifying assumptions or approximations could be made. The development was done painstakingly slowly and deliberately so as to test the dependence on every variable, observe behavior in limiting cases and slowly work up to the full-fledged three-dimensional model.

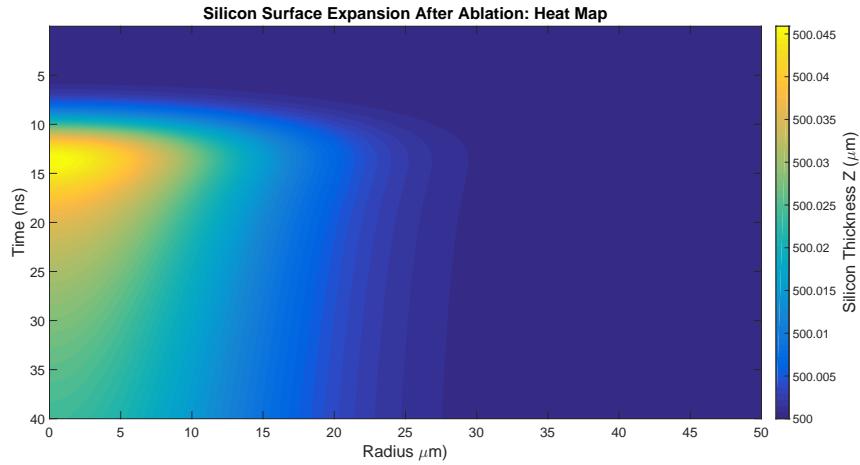


Figure 5.10.2: Two-dimensional heat map of Figure 5.10.1. Here, the silicon thickness is exclusively denoted by the color, with “hotter” colors indicating larger expansions.

the heat map in Figure 5.10.2, we explicitly see the diffusion by tracking a given color curve to see that the farther radial distances expand more slowly, after the inner distances have already expanded outwards. Moreover, if the ejection times are actually computed, it is found that the ejection time increases with radial distance from the center of the ablation pulse.³⁵

Radial and temporal traces of the surface expansion are shown in Figures 5.10.3 and 5.10.4, respectively. The former shows how the peak thickness of the silicon varies with radial distance from the center of the pulse while the latter displays the time evolution of the silicon thickness at the pulse center. It is noteworthy that the silicon surface only expands by just over 0.05 microns. Further, the timescale for this expansion before sphere ejection is about 10 nanoseconds. In Andrew’s thesis, he computed a timescale for expansion of 150 picoseconds, which he claimed was also a good estimate of the relaxation time of the silicon back to equilibrium with T_{amb} of the surrounding; his estimate for the magnitude of thermal expansion was 33.4 nanometers [32]. Clearly, from the results of this first principles model, Andrew’s timescale estimate is off by about two orders of magnitude. His magnitude of the expansion estimate, though, is in the same order of magnitude. A good check that the model’s timescale of 10 nanoseconds is actually reasonable is to note that the pulse duration is 20 nanoseconds. The largest intensity will be in the temporal middle of this pulse, so it makes sense that the most expansion would occur around that timeframe.

³⁵This has little effect on the time-of-flight properties because these ejection time differences are on the order of at most nanoseconds whereas the peak PMT signal times are on the order of tens to about a hundred microseconds.

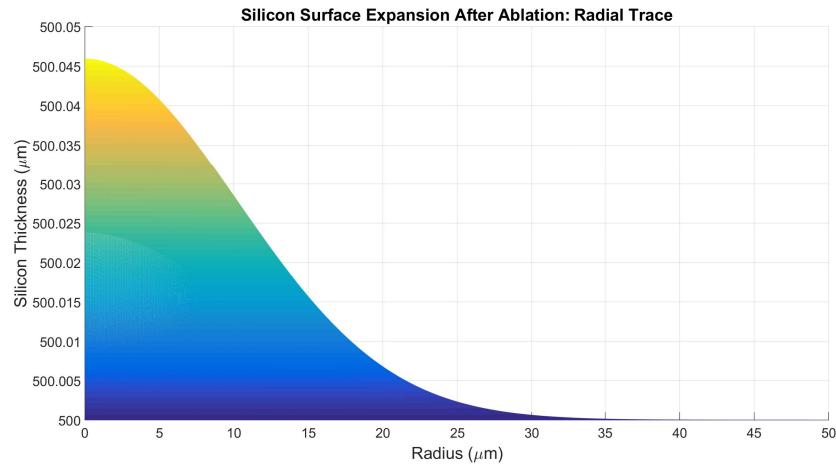


Figure 5.10.3: Radial trace from Figure 5.10.1.

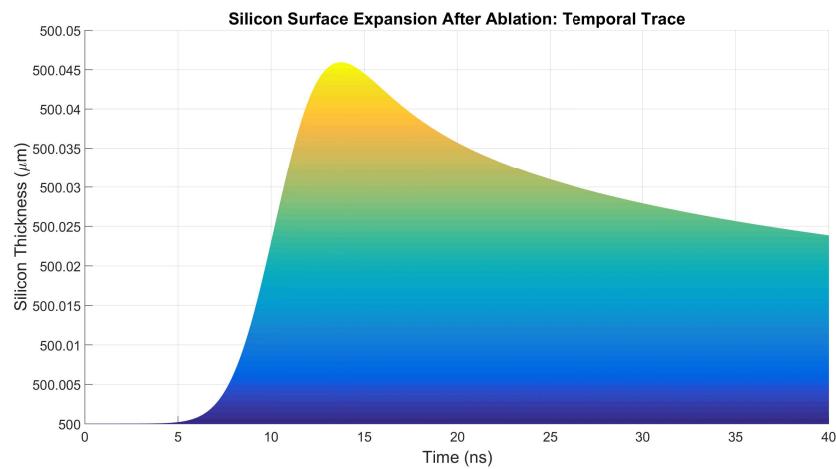


Figure 5.10.4: Temporal trace from Figure 5.10.1.

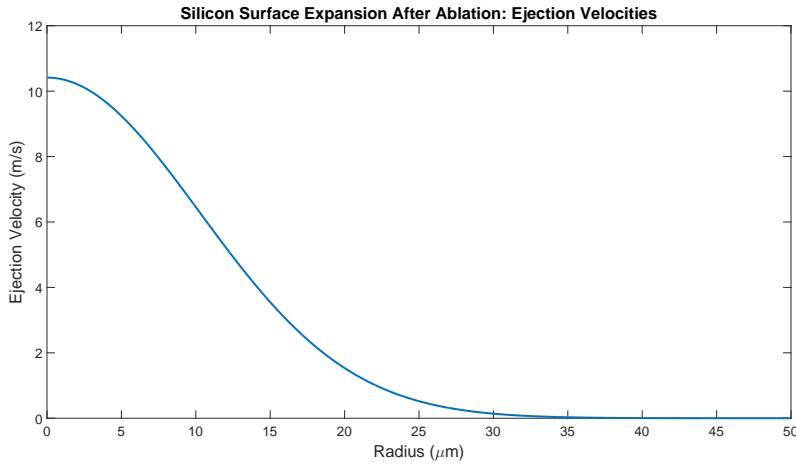


Figure 5.10.5: Ejection velocities corresponding to Figure 5.10.1. This is the velocity with which spheres will be ejected from the silicon substrate as a function of radial distance from the ablation pulse center.

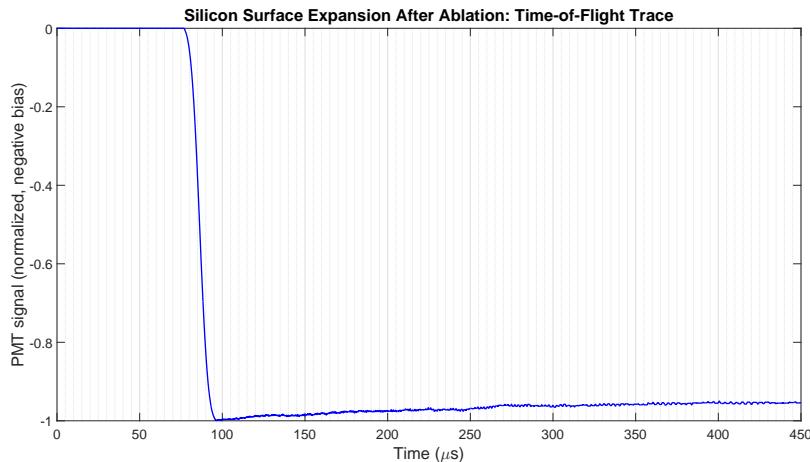


Figure 5.10.6: Simulated PMT signal. This uses the velocity distribution from Figure 5.10.5 and the HCP packing from the simulation model. The normalized PMT signal with a negative bias is plotted as a function of time since ablation, for the same duration as for lab data.

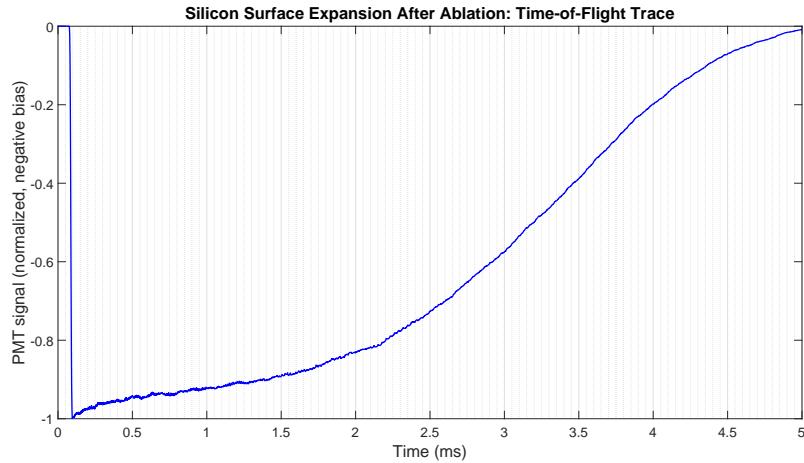


Figure 5.10.7: The first-principles simulated PMT signal of Figure 5.10.6 plotted for a longer time since ablation.

We use the time-of-flight analysis from the Monte Carlo model with its initial HCP sphere positions and the simulated interactions with the scattering beam, outlined in Section 4.5. Using this with the velocity distribution from Figure 5.10.5, we get a simulated PMT signal, shown in Figure 5.10.6. This will be compared to the other models and to data in Section 7.1. Qualitatively, though, we see there is the usual leading edge, but the peak signal lingers for quite some time, a couple of milliseconds in Figure 5.10.7. This is due to the interactions of the ablation pulse with the spheres, specifically their bonding to the substrate, being effectively ignored, causing a large number of spheres near the edge of the ablation disk to be ablated with very slow velocities. We will return to this issue in Section 6.3.

5.11 Error analysis

Similar to the analysis of the probabilistic and simulation models, most of the error comes from the assumptions we made.

One assumption is that the ablation pulse is a Gaussian in both radial space on a given cross section and in time. Every real laser beam is some combination of laser light modes. For a perfect Gaussian, the only propagating mode would be the 00 one. As mentioned earlier, however, from imaging the focal spot of our ablation laser at a higher frequency and lower intensity, we find that it resembles a Gaussian to good approximation along the radial dimension in a cross section. Unfortunately, it is much harder to verify the temporal distribution due to imaging and detector difficulties on the nanosecond resolution level. This temporal Gaussian assumption is the one that is more questionable, but from relay

imaging our initially flattop beam along the ablation path, it should be quite accurate. For an analysis on how a near flattop beam spreads into a Gaussian through free space propagation, see Trover's paper [31]. Overall, this assumption will likely not be the source of much error in the model. The only really presumptive part of using it is making an estimate of the spatial Gaussian standard deviation along the z propagation axis, σ_z , although this is accomplished fairly easily with knowledge of the pulse duration t_p and the speed of light.³⁶

Also with the laser beam, there is bound to be some amount of astigmatism, as with any laser system. This makes our assumption that the expansion characteristics will be azimuthally symmetric slightly questionable. However, as with the Gaussian approximation, it is likely a very good one. When observing the imaged focal spot, the beam was quite azimuthally symmetric with only slight discrepancies. These would maybe affect spheres local to that region, but certainly not the bulk time-of-flight characteristics. This is because the majority of the substrate's thermal expansion is from the incident laser intensity being directly absorbed rather than by heat diffusion from nearby silicon material. This is not true of course for spheres radially distant from the ablation pulse's center, but these spheres do not contribute to the leading edge and peak times that we are primarily concerned with.

We are again solving things numerically, which introduces some error regardless of how the calculations are carried out. Fortunately, the timestep and grid box dimensions we use are quite small due to the short timescale for surface expansion: our timestep is $dt = 10\text{ps}$, our radial box length is $dr = 50\text{nm}$, and our z dimension length of each box initializes at $dz = 1\mu\text{m}$. So even though the basis for our numerical solution is the finite element method, the results will be quite accurate. Indeed, testing the model with adding even more spatial partitions and making shorter timesteps does not change the result, indicating that a quasi-stable numerical solution has been reached. Hence, we use those parameters because the run-time scales quite harshly with the number of partitions due to the use of arrays in optimizing the code.

The biggest source of error that remains untested or unverified is the dependencies of all the constants listed in Table 3 and used in Equation 5.8.2. Again, preliminary testing showed that these dependencies should not make a meaningful difference in the temperature ranges experienced during ablation, but none of the dependencies have been put explicitly into the code. If the model is to be trusted without doubt and used as a predictive tool when transferring the laser ejection setup to a new experiment with different equipment, these dependencies should probably be included into the code such that it is an even more accurate first-principles model with minimal approximations.

³⁶A full imaging or other measurement of the temporal distribution of an ablation pulse to get an experimental measurement of σ_z would be a great future endeavor if this model is to become heavily relied upon.

6 Improving the model

In this section, we discuss several paths towards improving our first principles model of the laser ejection mechanism. First, motivated by Aaron's thesis and his brief look at spherical lensing, we calculate this effect with another model that traces rays through a sphere and calculates the intensity distribution that would be incident on the silicon surface. Then we turn our attention to further calculations that could be processed in the first principles model, specifically determining the ejection angles of the spheres. Finally, we address one of the big assumptions and approximations that went into developing the first principles model in looking at the ablation laser's interactions with the polystyrene nanospheres and their bonds to the substrate.

6.1 Ball lensing

Spheres that are transparent to light can roughly act as a lens. This effect is well known and the “effective focal length” is a well-defined quantity based on the diameter and index of refraction of the sphere. It is easily found with simple ray tracing, which determines the regime for which this applies. Namely, ball lensing will only be present for sphere sizes whose diameters are large compared to the wavelength of the laser. For smaller sphere sizes that are on the order of the wavelength, Maxwell equations are necessary to determine how the light propagates through the sphere. Doing so would be a bit tedious because the general result will be that for such sphere sizes, individual spheres diffract the light across the silicon surface, effectively producing no significant changes to the spatial intensity distribution between when the laser pulse is incident on the spheres and when it is incident on the silicon substrate.

In our experiment, however, we are not interested in the effective focal length of the sphere but rather in how the light hits a surface immediately following the sphere. Specifically, we are interested in how polystyrene spheres of micron scale radii and index of refraction around $n_s = 1.615$ deposited on a silicon surface act as lenses for incident light. We want to determine why we sometimes observe burn spots on the silicon surface after $5\mu\text{m}$ spheres are ablated with such a laser, as seen in Figure 6.1.1. Aaron studied this effect with some approximations that usually go into finding the focal length of a spherical lens in order to determine an effective fluence on the surface after going through the sphere [26]. However, he wound up finding a formula that depends on the angle of incidence, where we want to look at the aggregate effect of all rays passing through the sphere; he suggested using ray tracing software to perform an exact measurement. So naturally we try to create our own version of a program that will calculate this for us.

To consider the problem of how a ball sphere acts as a lens, we take a mathematical approach. Namely, this problem can be considered a mapping prob-

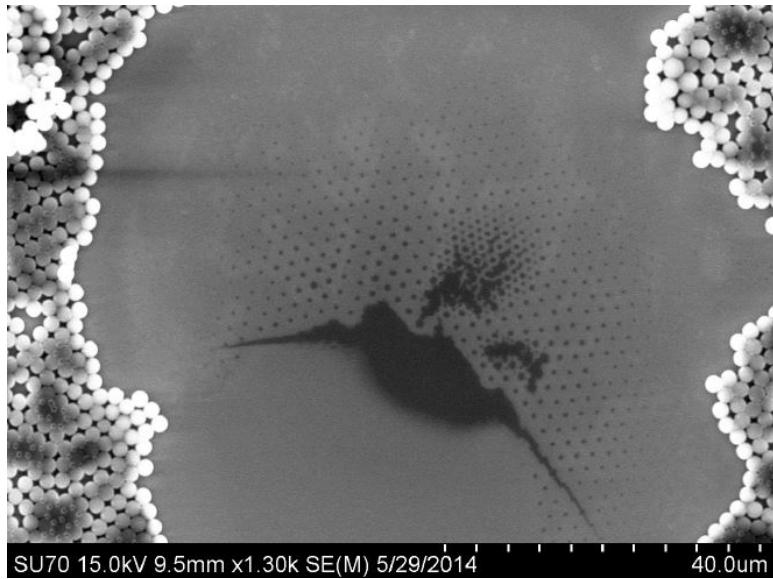


Figure 6.1.1: Burn spots on the silicon surface. This picture is of an area of $5\ \mu\text{m}$ spheres that have been ablated at 40W. Taken directly out of Aaron's thesis without alteration [26].

lem. The sphere acts as a function that maps one set of polar coordinates to another. That is, the sphere acts as some function f_s such that we have a mapping $(r, \theta) \rightarrow (f_s(r), \theta)$. Note that θ does not change because of the spherical symmetry: Snell's Law will not alter the angle θ . Here, the first set of coordinates is in the circle plane parallel to the surface, but located at the top of the sphere. This circular plane is the projection of the sphere onto the plane parallel to the surface at the top of the sphere. The second set of coordinates is in the projection of the sphere onto the surface. Essentially, we are trying to determine where a given light ray ends up on the surface as a function of where it starts on the sphere.

6.1.1 Assumptions

We make the following assumptions in order to make this problem somewhat similar. These assumptions are reasonable, however, as will be justified.

- (i) We assume that there is no other material other than a perfectly round sphere, the perfectly flat surface and the perfectly uniform vacuum that all physics is done in. In the context of polystyrene spheres on a silicon surface, this is a reasonable assumption. The polystyrene spheres are manufactured with some care, so one could reasonably expect them to be close to perfect spheres; the silicon wafers we deposit the sphere solution on are laser etched and are therefore of good quality and our system alignment is

off by at most one degree from a perpendicular ablation beam to the silicon, so our surface as seen by the laser is pretty nearly flat; and our actual ablation shots are done in high vacuum with both a roughing and turbo pump.

- (ii) Assume for a given sphere that the electric field from the incoming light source is uniform so the intensity over our initial circle space is uniform. In our experiment, the spheres are small compared to the beam waist of the focused ablation laser, so even though the beam is roughly Gaussian, we can reasonably treat the laser light as uniform over the initial circle space.
- (iii) Finally, we assume that light rays that are incident from reflections or transmissions from other nearby spheres are negligible. On our depositions, the spheres on a good slide are in a hexagonal close packing (HCP) structure in a single monolayer, so there are six spheres immediately adjacent to any given sphere. Hence, there will be some light that is incident on the sphere from these adjacent spheres, but it will be small as only a small percentage of light is reflected off the adjacent spheres and only for very large values of θ near $\frac{\pi}{2}$ will any transmitted light make it from an adjacent sphere to our sphere. Hence, a very small proportion of the incident light on the adjacent spheres is sent to our sphere, meaning it is negligible for our purposes.

6.1.2 Spherical symmetry and circular geometry

From the spherical symmetry present in this problem, we can restrict our attention to two-dimensions as Snell's Law interactions will not affect the ϕ angle unless by sending r to $-r$. Hence, we can apply the ray tracing geometry ignoring ϕ entirely and looking at where each light ray at a given radial distance from the sphere's center axis (perpendicular to the silicon surface) is sent to on the silicon surface and then multiply the intensity by the area on the sphere's surface that corresponds to the radial distance. To do this, we split the interval of all possible θ values for the incoming light ray into N evenly spaced test values of θ . We then use the smaller θ value at each interval to apply the calculation. This is a variation of a left-Riemann summation. The last data point at $\theta = \frac{\pi}{2}$ has no area corresponding to it and is therefore considered to carry no energy.

There are two possible scenarios for the ray tracing. One is that the ray ends up on the same side of the center axis on the silicon surface as it was as an incident ray. The other is that it is sent to the other side of the center axis. We will analyze the first scenario to begin with. This is shown in Figure 6.1.2. Note first off that for any chord on a circle, the angles of intersection between the chord and the circle are the same at both ends. This is why the Ψ angle is labeled the same on both ends of the chord. This in turn implies that the angle between the

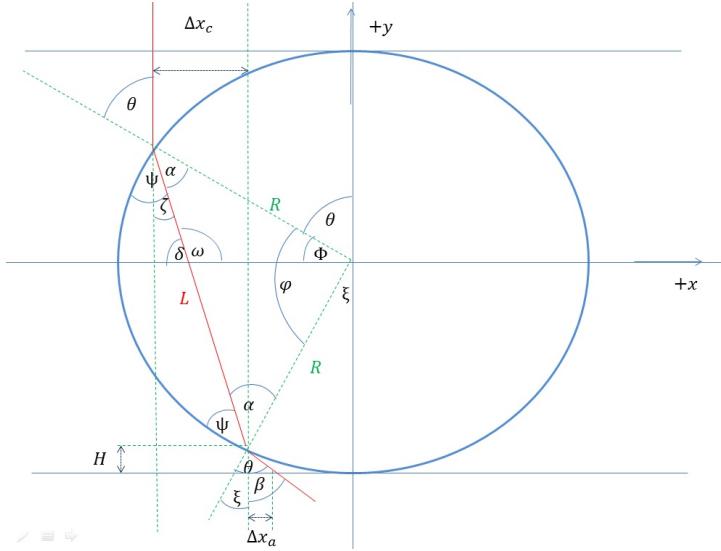


Figure 6.1.2: Geometry for the first scenario; view along the silicon surface. Here, this silicon surface is located at the bottom of the circle at $y = -R$, extending in and out of the page and along x in a two-dimensional plane. The incident ablation pulse comes from positive y downwards towards the sphere. The x axis is in the same plane of the silicon surface, perpendicular to the ablation laser.

chord and the normal to the circle, α , is the same on both ends as well. Snell's Law relates the angles of incidence, reflection and refraction via Equation 5.2.1. Hence, the exiting angle from the circle must also be θ since the equation is the same for the second interface as the first and α remains the same inside the circle at both interfaces. Now, from geometry, chord lengths, and Snell's Law, we can establish the following relations:

$$\begin{aligned}
 \alpha &= \sin^{-1} \left(\frac{n_a}{n_c} \sin(\theta) \right), \\
 L &= 2R \sin\left(\frac{\varphi}{2}\right), \\
 \varphi &= \pi - 2\alpha, \\
 \Phi &= \frac{\pi}{2} - \theta, \\
 \omega &= \pi - \alpha - \Phi, \\
 \delta &= \pi - \omega, \\
 \zeta &= \frac{\pi}{2} - \delta, \\
 \Delta x_c &= L \sin(\zeta).
 \end{aligned} \tag{6.1.1}$$

When these are put together, they yield the result of $\zeta = \theta - \alpha$ and therefore $\Delta x_c =$

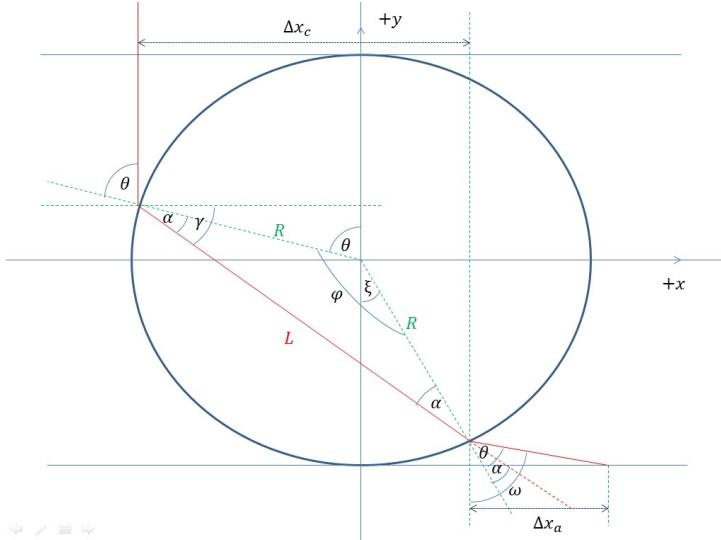


Figure 6.1.3: Geometry for the second scenario; view along the silicon surface. Again, this silicon surface is located at the bottom of the circle at $y = -R$, extending in and out of the page and along x in a two-dimensional plane. The incident ablation pulse comes from positive y downwards towards the sphere.

$2R\cos(\alpha)\sin(\theta - \alpha)$. To calculate Δx_a , we use geometry to obtain even more angle relations:

$$\begin{aligned} \xi &= \pi - \theta - \varphi, \\ \beta &= \theta - \xi, \\ H &= R - R\cos(\xi), \\ \Delta x_a &= H\tan(\beta). \end{aligned} \tag{6.1.2}$$

Putting these together, the result is $\Delta x_a = R(1 - \cos(\beta))\tan(2\alpha)$. The total x displacement from the starting location is the sum of Δx_c and Δx_a . Note that we chose a negative x starting value in both scenarios for convenience of calculation, as will be seen in Section 6.1.3 shortly. The second scenario is shown in Figure 6.1.3. In this scenario, the light passes across the center axis of the sphere (the axis perpendicular to the surface of the silicon that still passes through the center of the sphere). This occurs when $\theta > 2\alpha$. A lot of the geometry and calculations follow very closely those of scenario one, so we will just summarize the

relations and the results for Δx_c and Δx_a . The relations are as follows:

$$\begin{aligned}
\varphi &= \pi - 2\alpha, \\
\gamma &= \frac{\pi}{2} - \theta + \alpha, \\
\Delta x_c &= L \cos(\gamma), \\
\theta &= \frac{\pi}{2} - \gamma + \alpha, \\
\omega &= (\theta - \alpha) + \theta - 2\alpha, \\
\xi &= \omega - \theta, \\
H &= R - R \cos(\xi), \\
\Delta x_a &= H \tan(\omega).
\end{aligned} \tag{6.1.3}$$

The last relation is only true provided $\omega < \frac{\pi}{2}$. Otherwise, the ray does not hit the silicon surface and is instead directed upward or horizontal in the figure. The results of putting all of these relations together is that $\Delta x_c = 2R \cos(\alpha) \sin(\theta - \alpha)$ as before but now $\Delta x_a = R(1 - \cos(\theta - 2\alpha)) \tan(2\theta - 3\alpha)$, provided that $2\theta - 3\alpha < \frac{\pi}{2}$.

Note that we put emphasis on calculating Δx_c and Δx_a because, respectively, they represent the perpendicular distance from the laser axis that a ray of light is displaced in the circular cross-section of the sphere and between the circle and the surface of the silicon. Hence, these two values determine the focusing behavior because they enable us to find where each light ray ends up in the perpendicular plane to the ablation axis, relative to where it started. In other words, Δx_c and Δx_a physically tell us how the spatial laser profile changes between being incident on the top of a sphere and hitting the silicon surface.

6.1.3 Turning geometry into a model

In order to actually determine how the incident light is mapped onto the surface of the silicon, we utilize the results of the previous section in implementing a numerical computation.³⁷ To create this numerical computation, we must consider that what we really want to know is the amount of power that is hitting the surface. So we will now translate our mapping of the x -coordinates into a mapping of the intensity.

To do this is fairly straightforward. Based on assumption (ii), the incident intensity σ is uniform over the plane tangent to the top of the sphere. For the computation, we split the angular range $(0, \frac{\pi}{2})$ into N points with $M = N - 1$ intervals of even angular length. For the n^{th} angle point θ_n , we can calculate the

³⁷Note that by noting that $\sin(\theta) = \frac{|x|}{R}$, we could create a direct polynomial function f that maps the x -coordinate of where the light hits the sphere initially to the x -coordinate where the light hits the surface. This function, though a polynomial, has numerous terms and is fairly long, so we will instead implement the results from the geometry.

corresponding x_n coordinate from the relation $x_n = -R \sin(\theta_n)$.³⁸ Then, the area in our quasi-Riemann summation that that sample point represents is given by $A = \pi(x_{n+1}^2 - x_n^2)$. The total energy contained by all the points on the sphere in the angular range (θ_n, θ_{n+1}) is then given by $E_n = A\sigma$. All these points will be mapped to the same radial distance from the center axis of the sphere on the silicon surface (again, the axis that is perpendicular to the silicon surface). This radial distance is calculated using the results of the previous section.

Once we have the energies at each radii on the silicon surface, we go back through again and split the sphere's circular projection into N discrete radial points (since ϕ coordinates again will not be affected) and sum the energy contained in each interval. This is converted again to the intensity by dividing by the total area between concentric circles represented by consecutive radial points. Once this has been done, we have our numerically calculation of the energy mapping from the incident light on the plane tangent to the top of the sphere to the silicon surface tangent to the bottom of the sphere.

6.1.4 Results

The numerically calculated mapping results in knowing the resultant intensity as a function of the radial distance from the center axis of the sphere perpendicular to the silicon surface. This can be represented in several ways. Two of these that we will now see are as a graph of intensity as a function of radius and as a heat map of the intensity on the silicon surface as compared to the incident intensity on the sphere from our laser pulse.

The results are displayed using a radius of one for convenience of knowing the scale of the x axes in relation to the radius of the sphere. Throughout, we use a total incident power of one watt such that the incident intensity is $1/\pi \approx 0.3183 \text{ W/m}^2$. We will always normalize to the incident intensity for intensity graphs and the sphere radius for distances. Perhaps surprisingly, our results scale to any sphere size for which ray optics are appropriate when the light wavelength is much smaller than the sphere diameter ($\lambda \ll R$), which is why we normalized everything. It also justifies saying our incident intensity is $1/\pi \text{ W/m}^2$ because we might as well consider the sphere to have a radius of one meter. However, it really doesn't matter since we always normalized the intensity and power to the incident values.

Figure 6.1.4 shows the intensity on the surface of the silicon. There are several key features to note in this graph. The first is that it displays the normalized intensity as a function of the normalized radial distance from the center of the sphere. This means that 300 on the y-axis means 300 times the incident intensity while 0.3 on the x-axis means at a radial distance of 30% of the sphere's radius from the center of the sphere or 30% of the way to the edge of the sphere.

³⁸Note again that we choose the negative value for x simply for convenience in calculations.

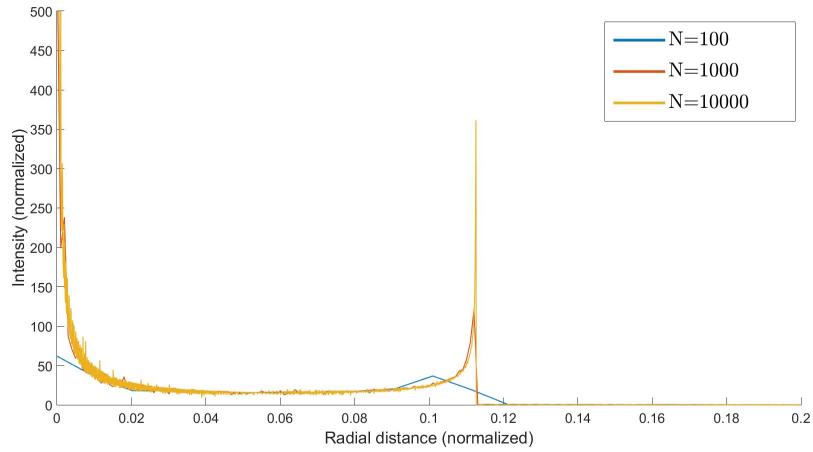


Figure 6.1.4: Numerical calculation of the ball lensing effect. The intensity of light (normalized to the incident intensity on the sphere) on the silicon substrate is plotted as a function of normalized radial distance from the sphere's center, for different numbers of partitions N .

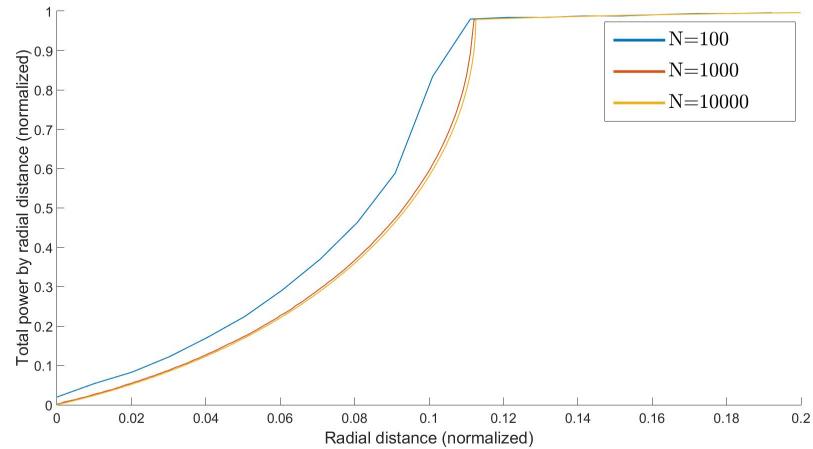


Figure 6.1.5: Another way to view Figure 6.1.4. The total power (normalized to a total incident power of one watt) up to the normalized radial distance from the sphere's center if plotted, again for different numbers of partitions N .

The second thing to notice is that the curve becomes smoother with more high-frequency noise and more pronounced, distinguishing features as the number of data points used increases. For example, the y-intercept for the $N = 10000$ graph is actually in the thousands. It seems as if the intensity increases as the number of data points increases. This of course doesn't make sense logically as the intensity should be some value independent of the number of iterations used in the numerical computation. However, as the number of data points increases, the distance between them decreases and since intensity goes as one over distance squared, it gets blown up arbitrarily near the radial distance of zero as the area between any two concentric circles is $\pi(2r\Delta r + \Delta r^2)$, where r is the radius of the smaller one and Δr is the difference in their radii. But the total power between any two concentric circles does in fact approach a constant value as the number of iterations increases, as can be seen in Figure 6.1.5. This plot shows the total power within a radial distance from the sphere center, as a function of that radial distance. Both are normalized to one based on the total incident power and the radius of the sphere.

The curve shape in Figure 6.1.4 is also very interesting to consider. We have already explained why the intensity blows up arbitrarily at the origin, but there is still the second peak with the sharp drop after and the small undulations at even larger radii. These are also easily explained by the physics of how Snell's Law interacts with the unique sphere geometry. As the angle θ increases, the radius on the surface of the silicon that the light ray is sent to also increases, to a point. There, the radius starts to decrease again towards zero and even crosses zero to go to the other side, where it proceeds to grow again at an increasing rate as θ approaches $\frac{\pi}{2}$. Where the turning point occurs when the radius starts to decrease, there are several rays that are sent to that location as there is always a build-up in graphs of polynomial functions near where the first derivative is zero. After the radius goes to zero again and begins increasing on the other side, it increases rapidly and eventually goes past this turning point, causing the small undulations seen after the sharp drop.

Another way to visualize the data is through the use of heat maps, which allow us to visually see where the intensity is large and small directly on the silicon surface itself. Again, we normalize to the incident intensity. However, we set a baseline of one for the surface intensity such that the Log doesn't blow up. These are displayed in Figure 6.1.6, and perhaps provide a more useful sense of the scale of the lensing effect. These were calculated using $N = 1000$ partitions.

Using a numerical ray tracing scheme causes the undulations that produce the artistic effects seen in slight blue variations in Figure 6.1.6b and the high-frequency oscillations about the general curve shape in Figure 6.1.4. Do not let these distract from the key takeaways from the data, namely the overall intensity scaling and the area to which the incident light is focused.

Here's the really amazing part. We model ball lensing in order to determine

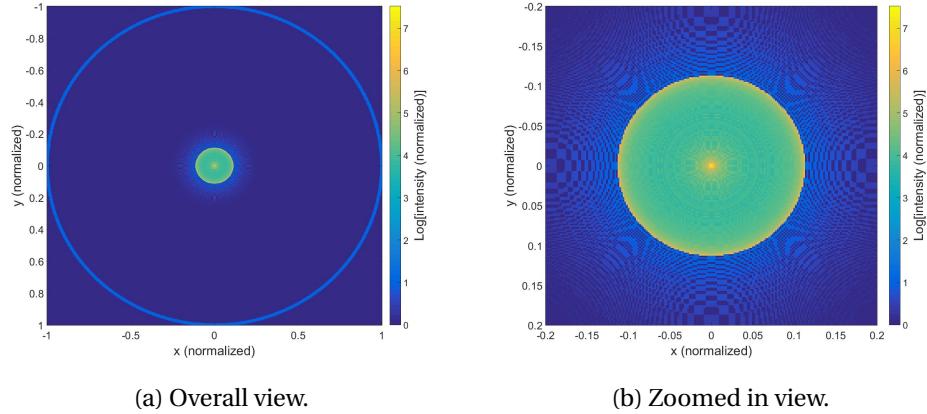


Figure 6.1.6: A heat map visualization of Figure 6.1.4. The Log of surface intensity (normalized to the incident intensity) is plotted against normalized position. In the overall view, the light blue ring is the projection of the sphere's cross-sectional area onto the surface.

why there are burn spots on the silicon surface after ablating $5\mu\text{m}$ spheres, as seen in Figure 6.1.1. Analyzing the pixels in this picture with MATLAB's imtool function, we find that the dark centers are either 2×1 or 2×2 pixels big while the centers are separated by an average of about 14.6 pixels in the HCP regions. Dividing the two yields 0.137. In our ball lensing model, we found that the incident light is mapped to a much higher intensity in an area on the silicon surface whose radius is 0.113 of the original sphere radius. Given that we essentially reached the pixel resolution limit in analyzing our experimental burn photo (the dark center could be anywhere from 1.5 to 2.5 pixels across yielding an uncertainty of 0.034 in the 0.137 measurement), we find the results of our calculation compared to the experimental data sufficiently close to suggest that ball lensing is the effect responsible for the burn spots in our $5\mu\text{m}$ ablation experiments since the normalized mapped area radius is within the uncertainty of the measured value from Figure 6.1.1. That is, the ball lensing effect causes incident light on a sphere to be focused onto the silicon substrate beneath with a much higher intensity in a smaller area. This has profound effects on how the spheres are ejected from the surface.

6.1.5 Updating the better model for $5\mu\text{m}$ spheres

We can improve the first-principles surface expansion model for $5\mu\text{m}$ spheres by including the ball lensing effect. Doing so, we effectively place spheres on an axis with even spacing as in an HCP monolayer and use that to determine where the incident laser pulse becomes focused. From the ball lensing model, we know

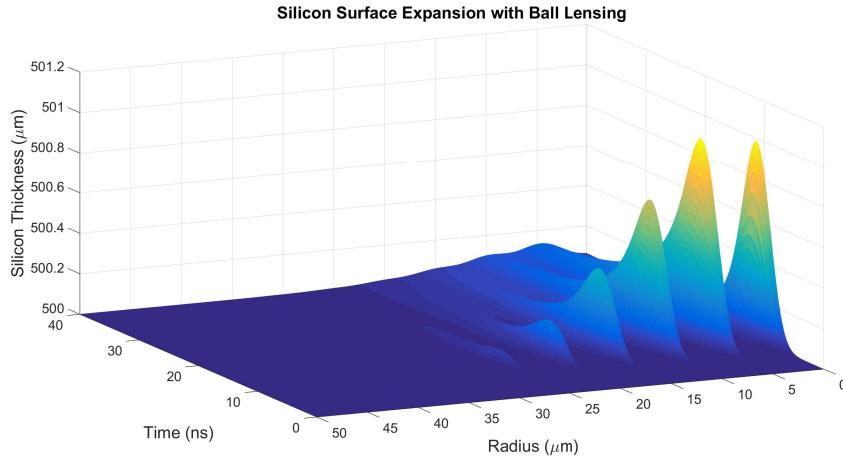


Figure 6.1.7: Surface expansion of the silicon substrate with the ball lensing effect included. The silicon thickness is plotted as a function of time and radial distance from the ablation pulse center.

how much area it is focused into and the amplitude scaling of the intensity. Propagating through the better expansion model with these new conditions, we find the results displayed in Figures 6.1.7 through 6.1.10 for the surface expansion of the silicon.

Note that for these figures, we are looking at the surface expansion as a function of time and radial distance from the ablation pulse center for one line of $5\text{ }\mu\text{m}$ spheres in an HCP deposition with the first sphere being in the exact center of the pulse. Hence, to actually determine the surface expansion characteristics and ejection velocities for the entire HCP deposition area, we would need to create a full-fledged three-dimensional model since azimuthal symmetry is no longer present as was the case for Section 5. However, since the main peaks are determined mostly from the absorption of laser heat directly into the silicon rather than diffusion, we can emulate such a model by merely running this model with the sphere positions as they would be along any radial axis from the center. Going over all possible radial axes with spheres on them, we can get an overall picture of the surface expansion with the ejection velocities of each individual $5\text{ }\mu\text{m}$ sphere in the original HCP deposition. Figure 6.1.11 shows the ejection velocity as a function of radial distance from the ablation pulse center for the line of $5\text{ }\mu\text{m}$ spheres that are packed as tightly as possible on the line.

In order to compare the time-of-flight traces, we need to propagate through the Monte Carlo method of simulating PMT traces, as was used to produce Figure 5.10.6 from the surface expansion in the absence of ball lensing. Doing this, we obtain the simulation PMT trace from the first-principles model shown in Figure 6.1.12. Note this signal has much more high-frequency components in,

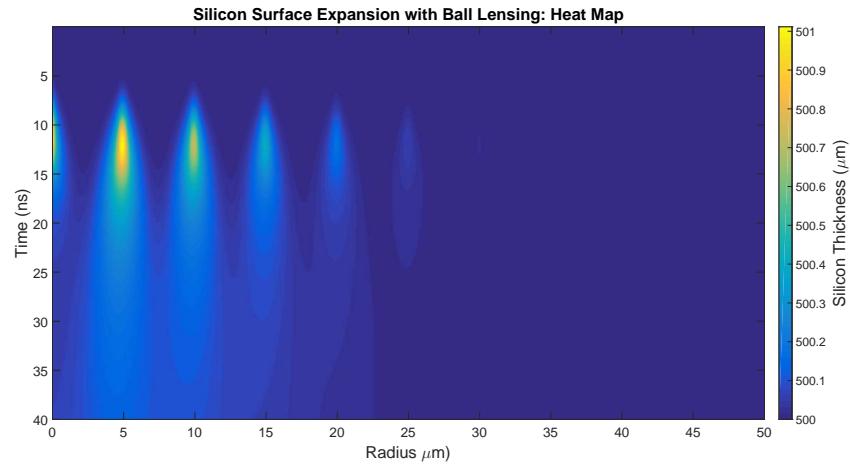


Figure 6.1.8: Heat Map view of Figure 6.1.7.

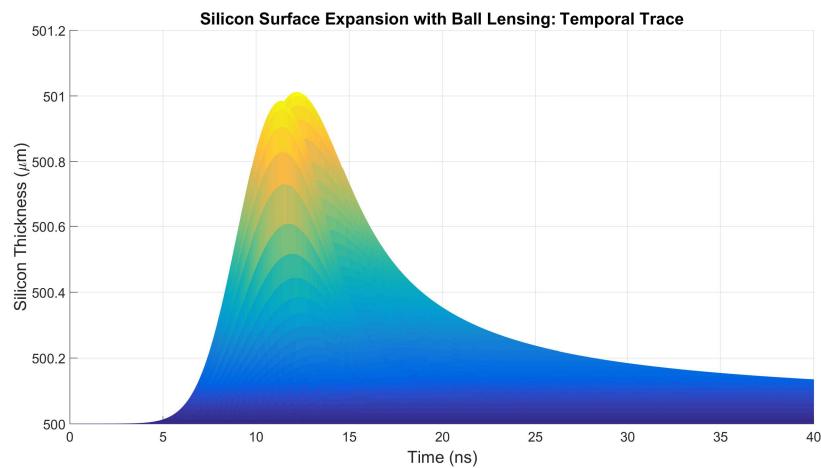


Figure 6.1.9: Time axis trace of Figure 6.1.7, showing the time evolution of the largest peak (and the first secondary barely visible with slightly different lining).

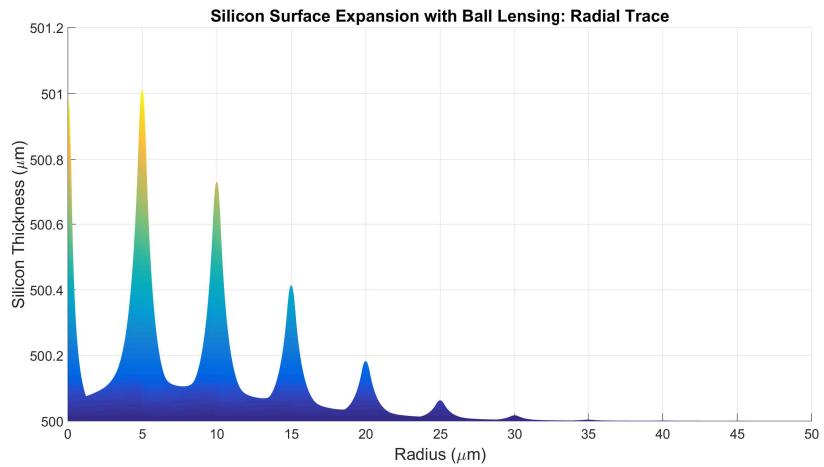


Figure 6.1.10: Radial axis trace of Figure 6.1.7, showing the peak silicon thickness at each $5\mu\text{m}$ sphere position and their relative heights.

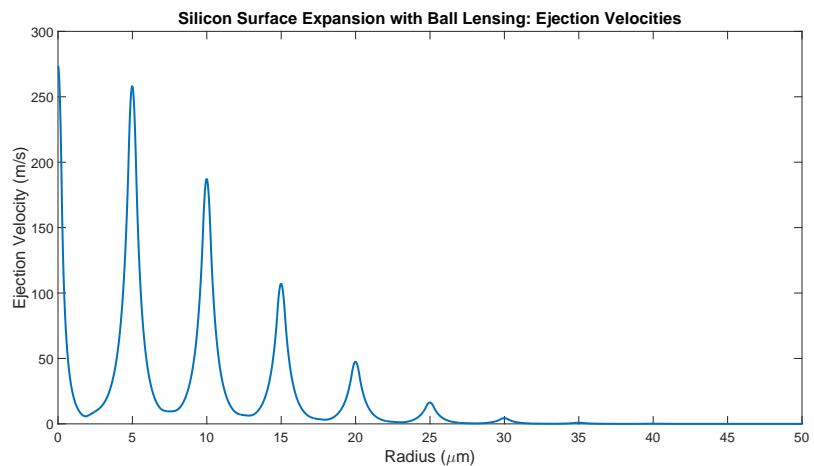


Figure 6.1.11: Ejection velocity as a function of radius with the ball lensing effect for $5\mu\text{m}$ spheres. Note that only the peaks correspond to actual ejection velocities.

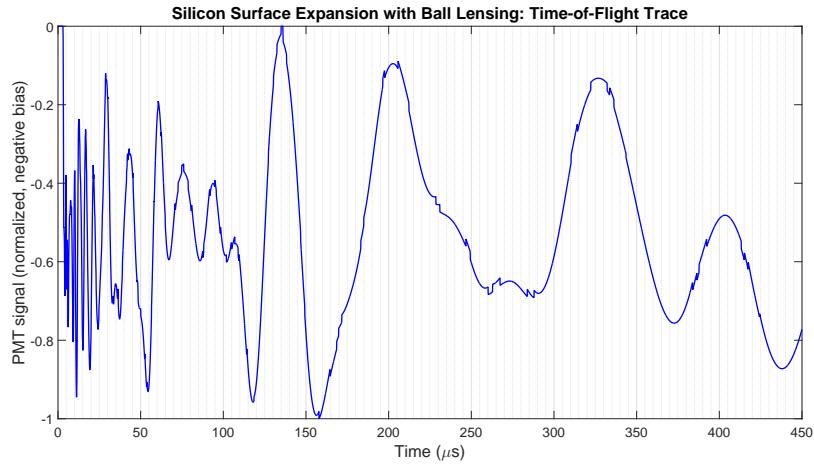


Figure 6.1.12: Simulated PMT signal including the ball lensing effect for $5\text{ }\mu\text{m}$ spheres.

it, making it seem much more erratic and unpredictable. This is due to the small number of spheres that get ablated from $5\text{ }\mu\text{m}$ depositions.

6.2 Ejection angles

One of the assumptions when building up the probabilistic and Monte Carlo models was that the spheres are ejection from the silicon substrate at some angles as well as with some velocity. The angle distribution along ϕ , the angle in the $x - y$ plane perpendicular to the laser propagation axis, was assumed to be even by azimuthal symmetry and indeed we will not change that. However, the angle distribution along θ , the angle from the z laser propagation axis, was taken to be uniform up to a maximum ejection angle of 20° in the probabilistic model; for the Monte Carlo simulations, it was taken to be a Gaussian distribution with three standard deviations corresponding to 20° ; in the first principles model, we assumed they would eject straight outwards from the surface.

With the first principles model, we can actually determine what the θ ejection angle is based on the normal to the silicon surface at the moment of ejection. Comparing this to the original normal of the unheated surface along the z propagation axis of the laser, we can determine the angle between them. The result of this analysis is shown in Figure 6.2.1. We see that the ejection angle is actually quite small for all spheres being ablated. The biggest ejection angle is about 1.8 mrad , which corresponds to 0.10° . Hence, our assumption in developing the first principles model that the spheres just eject straight outwards turns out to be much more accurate than our assumptions in developing the other two models.

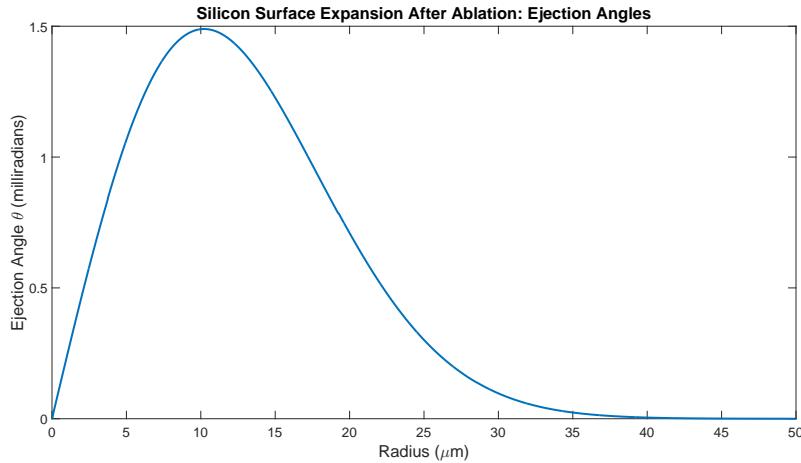


Figure 6.2.1: The θ ejection angle as a function of the radial distance from the ablation pulse center.

We can offer a couple explanations as to why this result differs from the assumptions we made in developing the other models. In Brendan's thesis, he actually noted that he suspected the spheres to eject more or less straight outwards as well based on the results of his collection efforts; he was inconclusive however due to a small data set and weird asymmetries that he couldn't explain, and therefore though they would likely be ejecting in a distribution as previous literature had found, with a maximum angle of 20° [12]. Indeed, in Lu et. al.'s paper, they found the ejection of particles from laser cleaning to most closely follow a Gaussian distribution [21]. Their setup is similar to ours, so we initially expected similar results, which is the justification for using a Gaussian distribution of θ ejection angles in the Monte Carlo simulations. We cannot account for the huge difference between that result and our model with certainty without knowing much more about their experiment and having significantly more data to back up our own model. However, we will proffer one explanation.

In a laser cleaning setup, there are particles of different sizes that require different amounts of input intensity from an ablative laser in order to be removed from the surface. Such a laser is likely to be Gaussian in its spatial profile perpendicular to its propagation axis. Hence, there will be less intensity in the tails of this distribution, implying that fewer particles will be cleaned from the silicon at farther distances from the pulse center. Hence, in their collected distribution of particles, it is plausible that the Gaussian result is merely a signature of the laser, assuming that the number of particles ablated from the surface is in some manner directly proportional to the intensity. Furthermore, the particles that do get ablated from distances farther from the ablation pulse center will be ejected slower than corresponding particles in the center, and hence will be less likely

to make it a set vertical distance above the substrate to the collection device, as was the setup in Lu et. al.'s experiment, causing the Gaussian distribution with θ [21]. Also in their experiment, the laser fluence is about two orders of magnitude below what we use for our sphere ablations. This could imply a very wide focus (they did not state their focus, merely that it went through a lens), and it would not be an unrealistic assumption that their beam could have a focal width on the order of the width of their observed Gaussian for particles as a function of θ . This would mean that the particles are ejecting straight outwards and any distribution in position when collected is simply due to a distribution in position on the surface.

6.3 Sphere transparency

When the ablation laser pulse is sent in, the light will either pass directly to the silicon surface or it will first interact with the spheres that coat the substrate. With laser-sphere interactions, we assumed in Section 5.1 that it would be essentially negligible as it would be governed by diffraction or Mie scattering, both of which would essentially send the light unperturbed through to the silicon surface along its original path. However, as seen with the ball lensing effect for large sphere sizes in Section 6.1, these interactions may not be so negligible. It is still likely that the diffraction and Mie scattering regimes cause the light to behave as if just incident on the silicon substrate, but this could be quantified and possibly included in a rework of the model.

Another issue we address is how much of the incident ablation pulse energy is lost on the spheres before interacting with the silicon surface. In Section 5.2, we discussed how the Fresnel coefficients determine the amount of light lost from reflection when light propagates across an interface between two media with different indexes of refraction. A crux of the better model is the absorption of electromagnetic energy in silicon. This applies to all materials. The light lost due to absorption in air is indeed negligible for two reasons: 1) our lab is kept at a low humidity and with minimal dust such that there is little for the laser pulse to interact with in air, and 2) we measure the fluence, laser intensity, and all other such values immediately before it enters our high vacuum system.³⁹

But this still leaves the interaction with our polystyrene nanospheres. To quantify how much light is absorbed by these spheres, we use absorption spectroscopy. Figure 6.3.1 shows the results of this measurement, indicating that with our 532 nm wavelength ablation pulse, the absorbance of fresh spheres will be about 2.5. Letting this be denoted A , we now try to relate this to the intensity loss due to absorption in the polystyrene nanospheres. The absorbance is directly proportional to length with $A \propto l$. Since our photospectrometry was done with a 1 mm path length, we must scale this to our nanospheres. At most, the far-

³⁹In a vacuum, there is no absorption of light because there is no material for it to interact with.

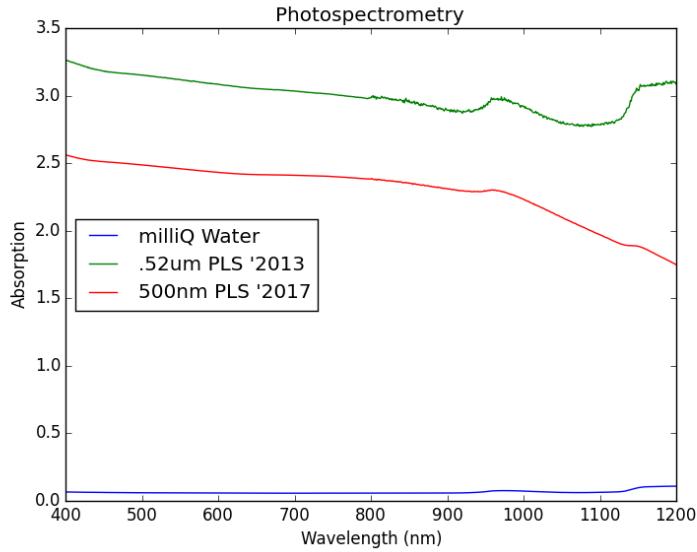


Figure 6.3.1: Absorbance of polystyrene to incident light as a function of wavelength for two of similar sphere solutions, taken with a 1 mm path length. This suggests that absorption increases with solution age as the green curve is from an expired sample.

At this distance the ablation pulse light would travel through polystyrene would be across the $5\text{ }\mu\text{m}$ of our largest sphere size. Hence, we will have the 2.5 value by the ratio of these lengths. Now, the absorbance is fundamentally related to intensities by

$$A = \log_{10} \left(\frac{I_0}{I} \right), \quad (6.3.1)$$

where I_0 is the incident intensity and I is the transmitted intensity. Solving this equation and making sure to scale for our lengths, we find that

$$\begin{aligned} I &\leq I_0 \times 10^{-2.5 \times 0.005/1} \\ &\approx 0.97 I_0. \end{aligned} \quad (6.3.2)$$

This is nontrivial and is therefore taken into account in the model. For smaller sphere sizes such as 500 nm, however, this quickly becomes less significant as we get $I \approx 0.997 I_0$.

The other major assumption we used in developing our models regarding the ablation laser pulse's interactions with the polystyrene nanospheres was that any surface forces are negligible. As mentioned in Section 5.1, Brendan estimated the largest surface force to overcome (by orders of magnitude) would be approximately $2 \times 10^{-6}\text{ N}$ for our $5\text{ }\mu\text{m}$ spheres [12]. The density of our spheres

is 1.05 g/cm^3 . From $F = dp/dt$ and the timescale for expansion of about 10 ns that can be seen in Figure 5.10.4, we can calculate the approximate minimum change in velocity needed to eject these spheres to get $v_{min} \approx 0.3\text{ m/s}$. If we impose this cutoff with our first principles model, the results actually do not change appreciably since most spheres that are ejected in the model actually eject with velocities just over 0.3 m/s at the edges, which suggests our model is very accurate. Using the edge of the observed experimental ablated area, we get velocities just above the cutoff velocity for the last few spheres ablated. This is pretty remarkable.

7 Outlook: bringing everything together

In this section, we bring together all the models developed thus far. We first compare them to one another and to actual data. Then, we examine the utility of the models for their created purpose, translating time-of-flight data taken on a large CW scattering beam to determine the likelihood of laser ejection delivering a target to a high-intensity laser focus with any reliability and better timing.

7.1 Comparing the models, with data

To compare the models, we will use a particularly nice set of three time-of-flight data traces with 500 micron spheres taken February 19, 2015 at a shooting distance of 0.8 mm between the sample and scattering beam edge. This data has a clear leading edge with a fairly smooth decay in signal strength afterwards. An overlay of the data with the results of our probabilistic model and three Monte Carlo simulations is shown in Figure 7.1.1. An overlay of the data with the result of our first-principles model is shown in Figure 7.1.2. From these, we can discern several features of our models and their ability to characterize the data.

However, before any discussion, we must emphasize one point again: all the experimental time-of-flight data to date very likely includes bonded spheres that are sticking together post-ablation. This is likely true on the grounds that we have not changed the initial deposition method much since Andrew and Aaron's days. The current deposition method was developed with Aaron and used for preparing the samples used to take all scattering data in his thesis that we use here. Hence, all the lab time-of-flight data may include different ejection characteristics than the ones our models assumed with un-bonded spheres that freely ejected independent of neighboring spheres. As a hypothesis into how these characteristics are different, we propose that when bonded spheres are ejected in a sheet, they will eject with a velocity near that of the more central spheres (with the higher ejection velocities based on Figure 5.10.5). There will most probably be several sheets that are ejected due to non-uniform HCP depositions. However, the bulk of them will eject with velocities near the central ejection velocity of about 10 m/s with smaller outlying sheets ejecting with less velocity, causing the slow, exponential-like decay in the PMT signal after the peak. Hence, the spheres are not ejected with the velocity distribution from Figure 5.10.5, but rather have a couple sheets ejecting with velocities near the maximum and outlying sheets ejecting slightly slower. As the first wave of sheets with high velocities passes into the signal, this produces the sharp leading edge with the peak corresponding to those fast groups about to exit the scattering beam on the back side as more outlying sheets have entered the argon-ion. Again, this is a hypothesis, but seems to explain the data. The key thing to note is that we cannot directly compare the first-principles model to our lab data as it is for a slightly different process.

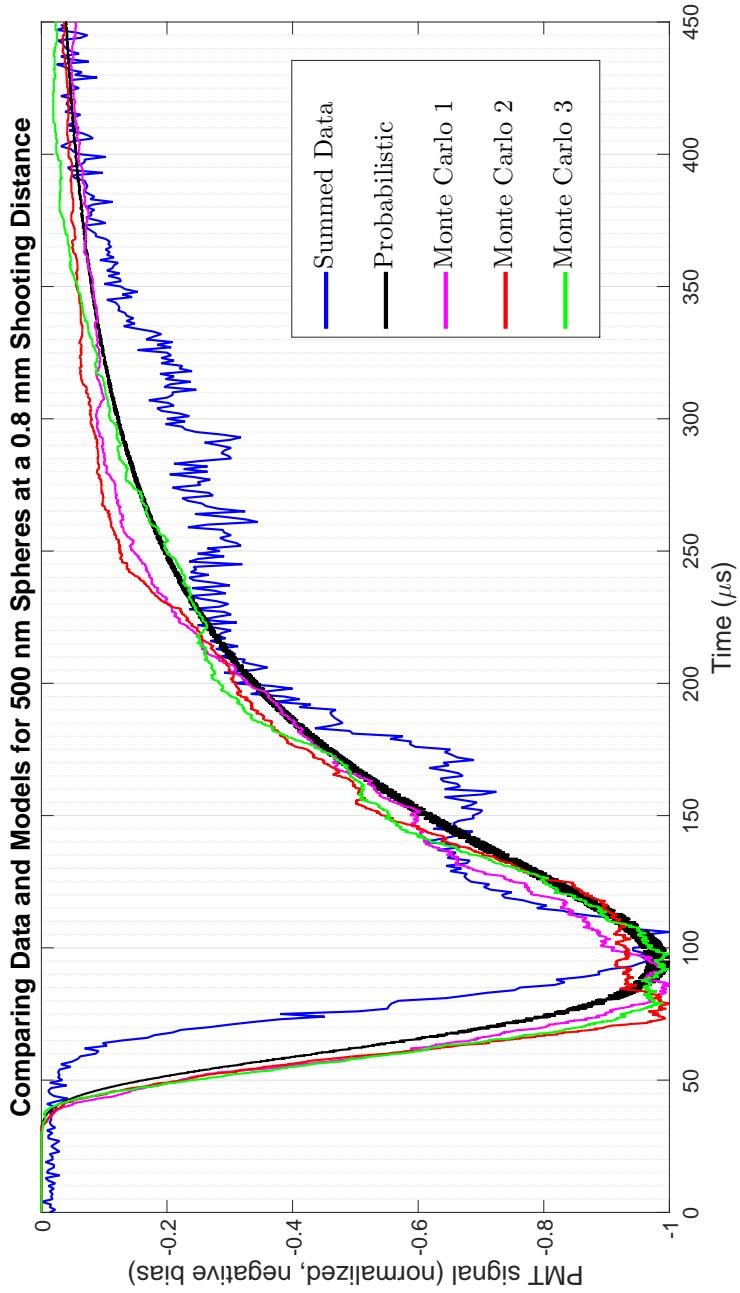


Figure 7.1.1: Overlay of summed data time-of-flight trace with the probabilistic result and three Monte Carlo simulations.

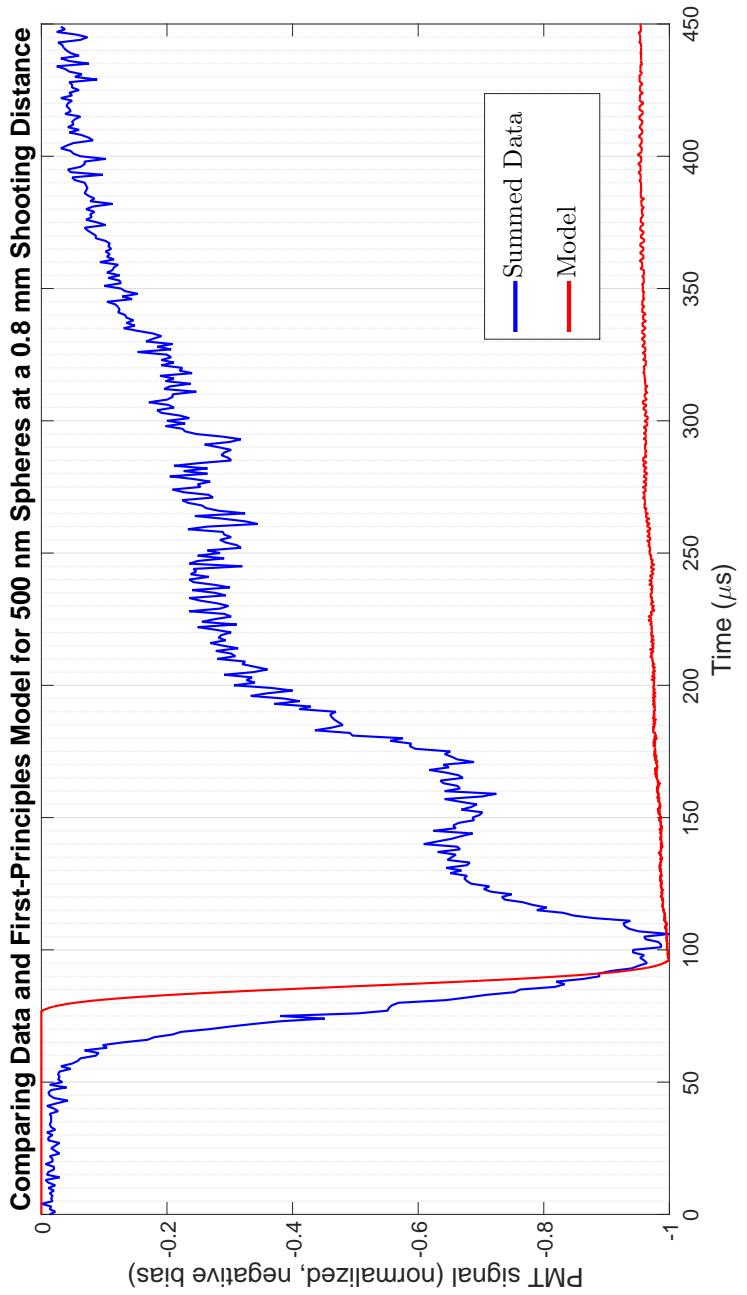


Figure 7.1.2: Overlay of summed data time-of-flight trace with the first-principle model's result.

One may question this argument given how good the probabilistic model and Monte Carlo simulations seem to replicate lab data. Indeed, from Figure 7.1.1, their peaks are within $5\mu s$ of the data's signal peak. Since the peak duration is on the order of $10\mu m$, this is a fairly good result. The signal decay after the peak of both probabilistic and Monte Carlo simulations are remarkably similar to the data. However, we must consider the velocity distribution that was used for both of these models: the Maxwell-Boltzmann. With this distribution, the vast majority of spheres will be ejected close to the most probable ejection velocity, with few spheres being ejected either slower or faster. Since we input the most probable velocity based on lab data from Aaron's thesis in Table 2, it makes sense that the signal peaks from these models should be close to the experimental ones [26]. Moreover, the fact that some spheres are ejected faster than the most probable velocity explains why the leading edges of the probabilistic model and Monte Carlo simulations are all earlier than the lab data by about $10\mu s$, because this cannot happen in reality. That is, the ablation pulse can only induce some maximum ejection velocity and no higher. Some spheres will be ejected more slowly however, even when in sheets, which is of course the case in a Maxwell-Boltzmann distribution as well, which makes the trailing edges of the signal match each other fairly nicely as well.

So then for the first-principles model, there are several things to consider. Based on these analyses, we should primarily look for agreement in the peak signal time. From Figure 7.1.2, we see this is the case with peak signal times corresponding nicely between the data and our model. The leading edge's beginning is slightly delayed in the model from the data. This is because in the experiments, with sheets there are many spheres ejected with higher velocities whereas in the model there are comparatively few that eject with such high velocities since area increases with radius in a circular geometry. Further, this explains why there is almost no decay in signal for the model whereas there is in the experimental data. This is one flaw of normalization: the total amount of signal should be nearly the same between our model and lab data because the same number of spheres are ablated in both and they interact with the same laser.⁴⁰ However, the data signal is primarily seen in the time window shown of $450\mu s$ whereas the model lasts for about a millisecond at the same relative strength more than 0.9 of the peak signal. Hence, in actuality, the peak model signal is much smaller than the peak data signal, as must be the case for the areas under the curve to be equal, to a good estimate. This again points to the fact that all the time-of-flight data thus far is for bonded spheres ejecting in sheets. The model's simulated PMT signal lingers in time because many spheres are ejected with slower velocities and the near uniform simulated signal strength across a wide time range points to the

⁴⁰Virtually all ejected spheres interact with the argon-ion since our ablation radius is approximately 50 microns and the scattering beam has a diameter of about 200 microns and all spheres are ejected close to the perpendicular as seen in Figure 6.2.1.

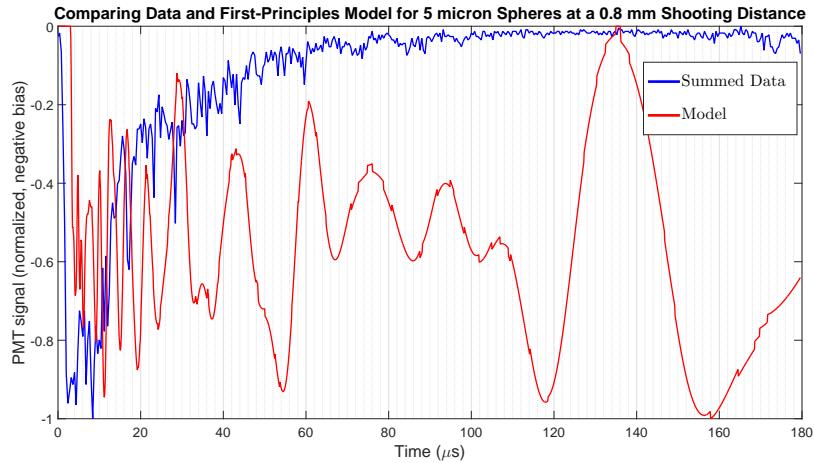


Figure 7.1.3: Overlay of summed data time-of-flight trace with the first-principle model's result including ball lensing for large spheres.

number of spheres being ablated scales linearly with distance from the center of the ablation pulse while the velocity decreases more rapidly with this distance. The two balance to some degree.

Thus, there are two main takeaways from comparing lab data to the models. First is that the probabilistic model and the Monte Carlo simulations agree so closely to the data in part because they take in a most probable ejection velocity from analyzed data; discrepancies are due mostly to unrealistic properties of a Maxwell-Boltzmann distribution. This lends to the conclusion of Section 4 that these models are much more useful as fitting mechanisms for the data, including assessing the most probable ejection velocity. For a predictive tool, the first-principles model is much more appropriate. Its discrepancies from the data are almost certainly due to the bonds between spheres and the undesirable sheets they are currently ejecting in. Nonetheless, the fact that the peak signal times are fairly close is a reasonably good indication of the model's accuracy. To better assess its reliability as a predictive tool, its results should be compared to experimental time-of-flight traces once the plasma etching technique has been refined and a comprehensive new set of scattering data has been taken by Hao and Marc. One caution to them is that the signal may be smaller than for past scattering experiments based on the area under the curve argument from above, so less neutral density (ND) filters may have to be used on the PMT to get a distinguishable trace.

For completeness, we also include a comparison of our first-principles model's result for 5 micron spheres with experimental time-of-flight data in Figure 7.1.3. Here, the result is much less predictable, so the comparison should not be taken as seriously as it was for Figure 7.1.2. For one thing, the 5 micron spheres had

very inconsistent time-of-flight traces with wildly varying peak times when we took data in the past, as noted in Aaron's thesis [26]. For another, there are so few 5 micron spheres ejected—on the order of hundreds—that group statistics are not exactly possible compared to data for the 500 nanometer spheres, say, with on the order of tens of thousands ejected. This, along with the ablation laser profile overlapping differently with the HCP lattice between shots accounts for the much more exotic fluctuations visible in both the data and our first-principles model. The important takeaway from this comparison is that the ball lensing effect discussed in Section 6.1 is indeed seen in the data with much faster ejection velocities and our model accounts for that to some degree.

7.2 What the models tell us

From the comparisons to data and the various other tests of our model's accuracy that have been noted throughout thus far, we can conclude that it is likely a pretty good predictive tool. It has no free parameters that must be specified and can hence be adjusted to any system by simply changing the values in Table 3 as appropriate to match the specifications of any experimental setup that might be used. This could translate across experiments and is not necessarily only applicable to studies of stochastic heating. As mentioned in the motivations for developing the models, this is incredibly valuable.

Also as mentioned in the motivations, we perform the time-of-flight scattering experiments in order to know when to fire a high-intensity laser after the ablation pulse is fired in order to successfully hit a sphere in the high-intensity laser focus. This poses a slight problem since our scattering beam is about 200 microns in diameter whereas the high-intensity laser focus we last used on UT Austin's THOR system is about 10 microns in diameter. Hence, we need to be able to translate our time-of-flight results at Mudd to the UT Austin system's parameters, and our first-principles model allows us to do this.

We will no longer simulate PMT signals, but will instead look at how many spheres would be in the high-intensity laser focus if it were fired at a given time. This is plotted as a function of that time, which is measured from the moment at which the ablation pulse is fired. For information on how to time these systems together, see Aaron's thesis' reference to a timing diagram we created and discussion of it [26]. All of the model runs in this section will be using 500 nm spheres and ablation laser specifications of our Q-switched one at Mudd, as quoted in Table 3. So, of course, this whole analysis will have to be re-run with whatever ablation laser is used for the stochastic heating experiment.

The first plot for timing the high-intensity laser based on our model's results is shown in Figure 7.2.1. Here, the number of spheres that would be hit by the high-intensity laser at a given time since the ablation pulse is plotted as a function of that time. The various curves are for different shooting distances, or separations between the sample surface and the high-intensity laser focus. We

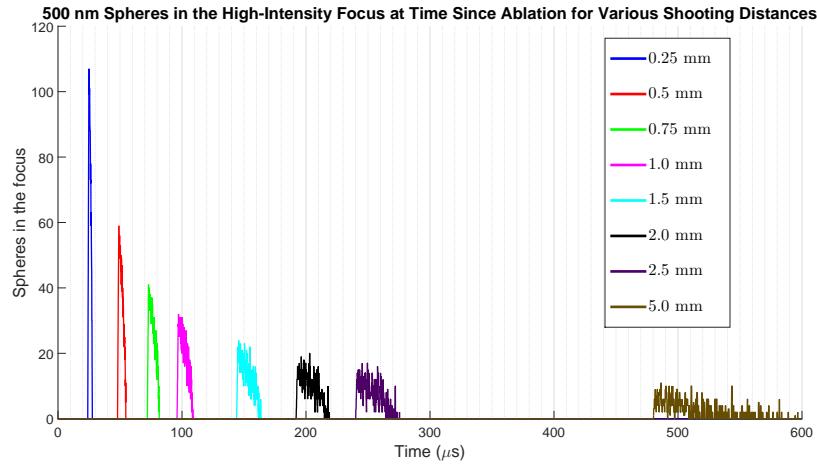


Figure 7.2.1: The number of 500 nm spheres that would be in the high-intensity laser focus at a given time after the ablation pulse is plotted as a function of that time for a variety of shooting distances. The shooting distance is the separation between the sample surface and the front edge (from the spheres' perspective) of the high-intensity focus.

can see that for greater shooting distances, the time width in which at least one sphere could be expected to be in the focus increases. However, this comes at the expense of the total number of spheres in the focus, so there is a trade-off. Additionally, beyond about 2.0 mm, some of the increased breadth in time when a sphere could be hit actually has significant undulations where there are brief moments where no spheres would be in the focus. Hence, while the greater duration of time when a sphere would be in the focus is desirable, it is only desirable to a point.

In Figures 7.2.2 and 7.2.3, we see similar analyses as in Figure 7.2.1, except with larger and smaller sphere sizes, respectively. We can see that the timing remains almost identical across all three plots for every shooting distance. However, the number of spheres that could be hit in each varies radically from at most 8 of the 2 micron spheres to at most 2700 of the 100 nm spheres.

In Figure 7.2.4, we use a fixed shooting distance of 0.5 mm and vary the laser intensity. It is wishful thinking to assume that any pulsed laser system will retain the same intensity on a shot-by-shot basis. This has been studied by many, such as Bechtel and Lee who studied the variation in pulse duration for an ND:Yag laser system all the way back in 1975 [1]. To get an idea for the sensitivity of timing the high-intensity laser to any intensity fluctuations of the ablation pulse, we run the model for intensities 10% above and below the intensity used throughout this thesis. As we can see from the plot, the timing does change significantly with such a large swing in intensity, but even then there is overlap between the

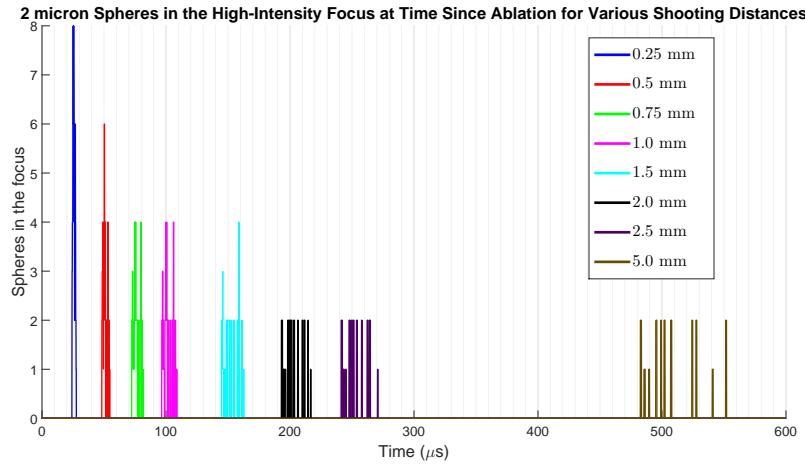


Figure 7.2.2: Same analysis of sphere targeting by shooting distance as in Figure 7.2.1, except for 2 micron spheres.

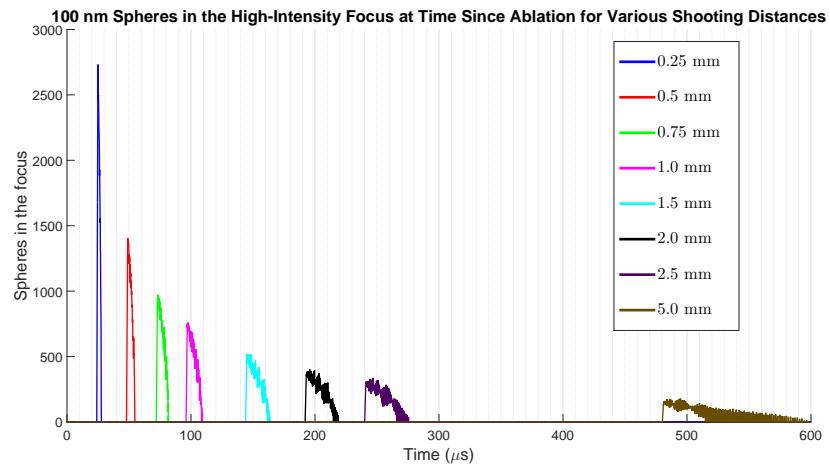


Figure 7.2.3: Same analysis of sphere targeting by shooting distance as in Figures 7.2.1 and 7.2.2, except for 100nm spheres.

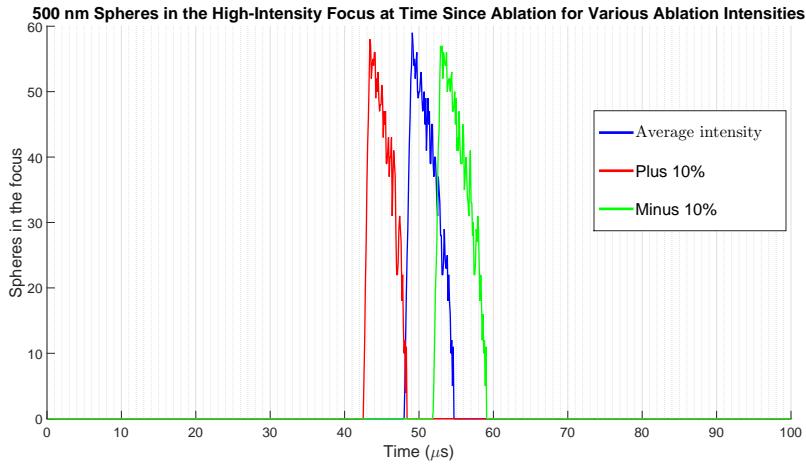


Figure 7.2.4: The number of spheres that would be in the high-intensity laser focus at a given time after the ablation pulse is plotted as a function of that time for a variety of laser intensities. The “average intensity” is the value we use for all other graphs in this thesis produced from our first-principles model, and is based on lab measurements of our ablation pulse’s intensity as quoted in Table 3.

times when spheres would be in the high-intensity laser focus. In reality, such large fluctuations are unlikely, and the expected magnitude would be more on the order of a percent or two at most, which would bring the three curves much closer in line. Hence, shooting for the average would, statistically speaking, be viable to hit a sphere on the vast majority of shots except extreme outliers in the intensity fluctuations of an ablation laser.

In Figure 7.2.5, we test the sensitivity of timing the high-intensity laser to potential misalignments, again using a fixed shooting distance of 0.5 mm. The misalignment we are incorporating for our test is between either the sample mount and the incident ablation laser or the ablation laser and the high-intensity pulse, as in Figure 3.1.1 with the scattering beam replaced as appropriate. Essentially, if the sample mount or ablation laser are off from parallel or perpendicular to the high-intensity laser path, then the ejected spheres will obtain some velocity along the x direction of the high-intensity beam and not plume straight outwards. The result of this is essentially to give the spheres an ejection angle. In other words, there will be a nontrivial θ component, unlike in Figure 6.2.1. From Figure 7.2.5, we see that any such misalignment is in fact a serious issue. The timing becomes quite unreliable beyond 1 to 1.5 degrees of misalignment, and beyond 3 degrees, there is no time in which a sphere would be in the high-intensity laser focus.⁴¹ Again, this analysis is for a 0.5 mm shooting distance and

⁴¹At least, not on the scale of 500 μ s. This is because spheres near the edge of the ablated region

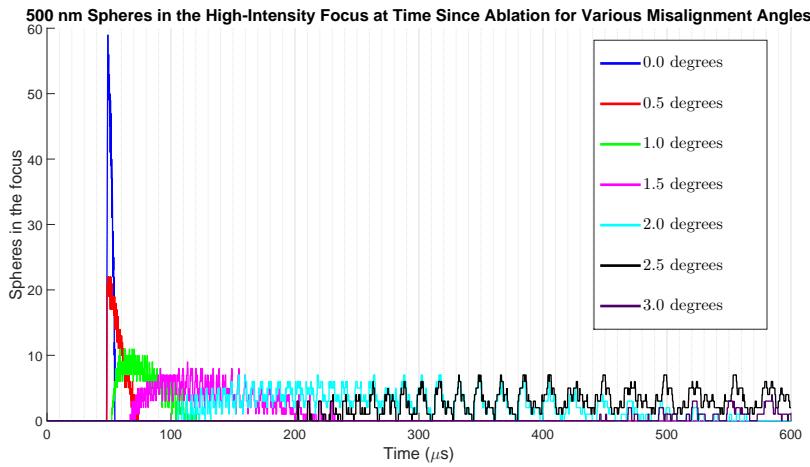


Figure 7.2.5: The number of spheres that would be in the high-intensity laser focus at a given time after the ablation pulse is plotted as a function of that time for a variety of misalignment angles. There are no spheres hit at any time beyond a 3.0 degree misalignment.

will only be more severe for larger distances. And of course from Figure 7.2.1, as the shooting distance is decreased, the time window in which to hit a sphere decreases. Further, unless a better sample mount is designed, we are limited to minimum shooting distance of around 0.4 mm anyways based on the geometry of a highly-focused laser pulse, as noted by Aaron [26].

This is of monumental importance. It implies that all alignment must be very precise in order for our laser ejection mechanism to have any chance at reliably delivering a spherical plasma target to the focus of a high-intensity laser. Even if it is, the timing will still be nontrivial and have slight variations due to ablation laser fluctuations. What is more, all of these models were run with ideal samples of perfect HCP monolayers, meaning the most spheres were ejected as possible without multilayers. So the samples used should be very good in order to get as close to this ideal as possible. Anything less will decrease the likelihood of successfully timing the high-intensity laser to hit a sphere in its focus after the ablation pulse. Ablating a larger area would not help, however, as spheres at the edge of the ablation disk would have zero probability of reaching the 10 micron high-intensity laser focus after being ablated more than 50 microns away in the $x - y$ plane with no or little ejection angle.⁴²

that would be more likely to pass through the high-intensity laser focus once an ejection angle is introduced move much slower than spheres towards the center of the ablation region.

⁴²Even if the system is misaligned such that there is some ejection angle, the spheres would be ablated with such a small velocity that they would not reach the focus in any reasonable time for timing purposes, in the extremely low probability case that they reach it at all.

8 Future work

In this section, we suggest some potential next steps for the lab group in coming years. It is essentially a compilation of the snippets mentioned throughout the previous sections.

8.1 Possibly useful experiments

There are a couple useful side experiments that could be done to test a couple things:

- (1) A temporal resolution of the ablation pulse to the nanosecond scale would be fantastic if possible with our equipment in order to determine the intensity distribution and standard deviation on the propagation axis of the ablation laser to get its temporal characteristics.
- (2) Experiments using the silica nanoparticles could be interesting. This was explored some by Amber in her thesis work, but could potentially be explored more as a substitute for polystyrene nanospheres [6].
- (3) A comprehensive set of time-of-flight data for un-bonded spheres of all sizes needs to be collected once the etching process is fine tuned, as discussed throughout this thesis and reiterated in Section 8.2.
- (4) Any experiments that could determine how deposition characteristics such as the sphere-silicon bonding strength changes with plasma etching. This experiments could potentially be carried out with electric fields or piezo oscillators, for example.

While it wouldn't be an in-lab experiment, it could also be useful trying out different distributions in the probabilistic and Monte Carlo models. Specifically with the Monte Carlo model, it could be worth trying other velocity distributions to select from, such as a log-normal, that may replicate the experimental conditions more accurately. Aaron fitted some data to a log normal and found it useful in finding velocities and peak signals [26].

8.2 Time-of-flight data with plasma etched spheres

One of the biggest challenges of the past year since Aaron's thesis, and our discovery of interspecies bonding in this past summer, has been trying to eliminate these bonds. They arise as a result of the current deposition process and make it impossible to run a stochastic heating experiment. Fortunately, we have been using oxygen plasma etching to eat away at bonds between spheres. However, as mentioned in Section 2, finding the ideal etching times as a function of sphere size is a work in progress. Further, once we consistently produce un-bonded

samples, a full set of time-of-flight data will need to be taken since the data in Aaron's thesis is suspect due to its high likelihood of being taken with bonded spheres. We need to be sure of the timing for the stochastic heating experiment such that we can reliably deliver a plasma target to the focus of a high-intensity laser. This scattering data should be the immediate focus of the group before heading to UT Austin this summer and after the plasma etching details have been sorted out.

8.3 Further model improvements

As mentioned in Sections 5 and 6, there are a couple of ways in which the first-principles model can be improved. Two of these stand out as having significantly more potential.

The first would be to account for the dependencies of some “constants” used in the model, as per Table 3. These dependencies were ignored based on preliminary tests of the model which found such dependencies to not alter the result in any meaningful way. That said, many of the constants dependencies remain untested and a rigorous model would have all the dependencies in it. There are two methods we suggest as possible routes to accomplishing this. The first is to find analytical equations for the dependencies that can be incorporated in the big updating steps within the MATLAB scripts. This is ideal as there is minimal additional computation cost simply evaluating an analytical function at each timestep. The second, less desirable method to incorporate dependencies would be to load in data arrays for each constant and then search through them with some algorithm to find the appropriate value to insert for each timestep update. This would likely be a major blow to the runtime of the code, but by no means insurmountable given that the current runtime is on the order of five to ten minutes with reasonable partition numbers. That is, there is room to spare for a longer runtime in the name of accuracy so long as the user is willing to exercise patience, unlike us.

The second path to improve the first-principles model would be to trace back even further in the interactions and dive into the physics of depositions and the ablation pulse's interactions with any bonds between the spheres and silicon substrate. Thus far, all the group efforts on this front have been estimations, including in this thesis with Section 6.3. A full derivation of these forces and interactions could only improve the legitimacy of our first principles model. Further, it would help to understand the deposition process and how to produce ideal samples. This is an open research project in our group as with Amber's thesis [6]. If samples can be produced consistently with large, HCP regions of unbonded monolayers, then the statistical reliability of either time-of-flight or stochastic heating experiment shots will be greatly improved. That is, if we don't know for sure whether we are shooting monolayer or multilayer on any given shot, then we will not be able to accurately assess statistics on a group of such shots.

9 Conclusion

We have offered an explanation as to the shape and variability of previous time-of-flight data with spheres ejecting in sheets due to interspecies bonding as a result of the deposition method. Eliminating these bonds must be accomplished before the experiment can proceed. Once they are shown to be eliminated, a comprehensive set of time-of-flight data must be taken to determine the ejection characteristics of our polystyrene nanospheres from their silicon substrates. This data should then be compared to a first-principles model developed for this experiment to ascertain the robustness of the model.

This model, if it agrees with un-bonded time-of-flight data, could be immensely useful in assessing how the data from Mudd with a 200 micron scattering beam translates to a high-intensity laser with a 10 micron focus. Applying the model to the latter setup already, we find that there is a critical sensitivity to alignment that requires any experimental setup not have more than a 3 degree misalignment between the ablation pulse and the high-intensity laser. Otherwise there will essentially be no probability of successfully delivering a sphere to the focus. Running the model with appropriate laser parameters, we can predict the time-of-flight characteristics and determine the optimal time to shoot the high-intensity laser after our ablation pulse has fired in order to hit at least one sphere in the focus. Knowledge of such timing is vital to the success of a stochastic heating experiment.

10 References

- [1] JH Bechtel and W Lee Smith, *Pulse width fluctuations in mode-locked nd: Yag lasers*, Physics Letters A **55** (1975), no. 4, 203–204.
- [2] Lawrence A Booth, David A Freiwald, Thurman G Frank, and Francis T Finch, *Prospects of generating power with laser-driven fusion*, Proceedings of the IEEE **64** (1976), no. 10, 1460–1482.
- [3] Boris N Breizman and Alexey V Arefiev, *Ion acceleration by hot electrons in microclusters*, Physics of Plasmas (1994-present) **14** (2007), no. 7, 073105.
- [4] Boris N Breizman, Alexey V Arefiev, and Mykhailo V Fomytsâžkyi, *Non-linear physics of laser-irradiated microclusters*, Physics of Plasmas (1994-present) **12** (2005), no. 5, 056706.
- [5] John Charles Butcher, *The numerical analysis of ordinary differential equations: Runge-kutta and general linear methods*, Wiley-Interscience, 1987.
- [6] Xinyue Cai, *Creating isolated nanospherical targets for a stochastic heating experiment*.
- [7] John Y Campbell and Samuel B Thompson, *Predicting excess stock returns out of sample: Can anything beat the historical average?*, Review of Financial Studies **21** (2008), no. 4, 1509–1531.
- [8] Casey Cannon, *Computational modeling of optical traps: Fun with fortran*.
- [9] Steven Chu and Arun Majumdar, *Opportunities and challenges for a sustainable energy future*, nature **488** (2012), no. 7411, 294–303.
- [10] T. Donnelly and T. Ditmire, *Collaborative research: An experimental study of relativistic, stochastic heating of electrons during the irradiation of wavelength-scale plasmas by a strong-laser field*, 2012.
- [11] Jake Fish, *Computational modeling of optical traps*.
- [12] Brendan Folie, *Ka-boom! lasers!*
- [13] VM Glazov and Andrei Sergeevich Pashinkin, *The thermophysical properties (heat capacity and thermal expansion) of single-crystal silicon*, High Temperature **39** (2001), no. 3, 413–419.
- [14] D Gogel, M Weinl, JKN Lindner, and B Stritzker, *Plasma modification of nanosphere lithography masks made of polystyrene beads*, Journal of optoelectronics and advanced materials **12** (2010), no. 3, 740–744.

- [15] Martin A Green, *Self-consistent optical parameters of intrinsic silicon at 300k including temperature coefficients*, Solar Energy Materials and Solar Cells **92** (2008), no. 11, 1305–1310.
- [16] James Hansen, Makiko Sato, Pushker Kharecha, David Beerling, Robert Berner, Valerie Masson-Delmotte, Mark Pagani, Maureen Raymo, Dana L Royer, and James C Zachos, *Target atmospheric co₂: Where should humanity aim?*, arXiv preprint arXiv:0804.1126 (2008).
- [17] Andrew Higginbotham, *Devices and results from an investigation of intense laser-matter interaction in the mie regime*.
- [18] AP Higginbotham, O Semonin, S Bruce, C Chan, M Maindi, TD Donnelly, M Maurer, W Bang, I Churina, J Osterholz, et al., *Generation of mie size microdroplet aerosols with applications in laser-driven fusion experiments*, Review of Scientific Instruments **80** (2009), no. 6, 063503.
- [19] OA Hurricane, DA Callahan, DT Casey, PM Celliers, C Cerjan, EL Dewald, TR Dittrich, T Döppner, DE Hinkel, LF Berzak Hopkins, et al., *Fuel gain exceeding unity in an inertially confined fusion implosion*, Nature **506** (2014), no. 7488, 343–348.
- [20] Matthew J Lang and Steven M Block, *Resource letter: Lbot-1: Laser-based optical tweezers*, American Journal of Physics **71** (2003), no. 3, 201–215.
- [21] YF Lu, YW Zheng, and WD Song, *Characterization of ejected particles during laser cleaning*, Journal of Applied Physics **87** (2000), no. 1, 549–552.
- [22] George Mitev, P Colby, NR Roberson, HR Weller, and DR Tilley, *Radiative neutron capture by deuterium*, Physical Review C **34** (1986), no. 2, 389.
- [23] Nazim Muradov, *Liberating energy from carbon: introduction to decarbonization*, Springer, 2014.
- [24] S Nakai and K Mima, *Laser driven inertial fusion energy: present and prospective*, Reports on Progress in Physics **67** (2004), no. 3, 321.
- [25] J Ongena and Guido Van Oost, *Energy for future centuries: Will fusion be an inexhaustible, safe, and clean energy source?*, Fusion science and technology **45** (2004), no. 2T, 3–14.
- [26] Aaron M. Rosenthal, *Experimental examination of multipass stochastic heating as a mechanism for electron heating in a strong laser field*.
- [27] Michael P. Schubmehl, *Ultrasonic atomization and the droplet source project*.

- [28] Hong Sio, *Device development: Generation of sub-micron fuel targets for fusion experiments*.
- [29] Andrew C Tam, Wing P Leung, Werner Zapka, and Winfrid Ziemlich, *Laser-cleaning techniques for removal of surface particulates*, Journal of Applied Physics **71** (1992), no. 7, 3515–3523.
- [30] RE Tipton, *Generalized lawson criteria for inertial confinement fusion*, Tech. report, Lawrence Livermore National Laboratory (LLNL), Livermore, CA, 2015.
- [31] Anthony A Tovar, *Propagation of flat-topped multi-gaussian laser beams*, JOSA A **18** (2001), no. 8, 1897–1904.
- [32] Andrew J. Yandow, *Characterizing the laser ejection of polystyrene nanosphere targets from a silicon surface*.
- [33] Xin Zhang, *Computational simulations of multi-pass stochastic heating*.

11 Appendix A: Code

The purpose of this appendix is to provide the reader with the code used for all of the models and data analysis. Each will be presented separately with a comment on any nuances of how to run them. For the sake of clarity, they will be presented separately in different subsections.

Another thing to note is that we wrote many, many more scripts and functions than would be reasonable to put in this Appendix to do a number of things and play around with various concepts. An example of this is the code we have that tries to simulate the deposition process and indeed gets HCP regions with thick boundaries, so it could be a very interesting place to start if some future group member wishes to take on modeling that process.

We will be putting all of the files necessary to run each of these models and various other things in both the Sakai group resources and on the group's GitHub account shortly after this thesis is turned in. Turn to that for the randraw function needed for the Monte Carlo simulations as well as everything needed for the fitting data processor GUI. Also, significantly more detailed readme files will be found in those places, so if this Appendix is at all confusing in how to operate the code, turn to those resources.

11.1 Measuring the focus

Here we present the code for analyzing the focal spot size of the ablation laser, once imaged with the camera. Two pictures are taken, one of the focus and one of the focus with a wire of measured size in it. Then, the following two scripts are used to analyze it and a similar division determines the FWHM of the ablation laser focus. How exactly to image the focus is described in Aaron's thesis [26]. Note that when using both scripts, they require a fair amount of playing around with MATLAB's imtool function before running, as detailed in the code itself.

This first script is for finding the pixel width of the ablation laser's focal FWHM.

```
1 ci = imread('35unsat.jpg'); % Input the appropriate file name
2 c = improfile(ci,[1,744],[155,155],744); % The first vector is
   % horizontal numbers, the second is vertical
3 % These are found by opening the image in the MATLAB
   % command line with
4 % test = imread('filename.extension'); and imshow(test);
   % then use
5 % imtool(test) and find the pixel numbers appropriate.
   % These should be
6 % along a line that passes through the center of the
   % brightest spot.
7 pixvals = linspace(1,744,744);
```

```

8  yvals = linspace(1,744,744);
9  relc = c(:,:,3);
10 FWHM = 0;
11 max = 5;
12 starti = 0;
13 endi = 0;
14 peaki = 0;
15 for i=1:744
16     yvals(i) = relc(i);
17     if (yvals(i) > max)
18         max = yvals(i);
19         peaki = i;
20     end
21 end
22 base = yvals(1);
23 HM = (max+base)/2.0;
24 for i=1:368 % The second number is based on the output of
    % peaki once this is run the first time
25     if (yvals(i) < HM)
26         starti = i;
27     end
28 end
29 for i=368:744
30     if (yvals(i) > HM)
31         endi = i;
32     end
33 end
34 FWHM = endi - starti;
35 FWHM % return the fwhm in pixels
36 max % return the maximum pixel brightness
37 peaki % return the pixel coordinate of the maximum
38 plot(pixvals,yvals) % plot the intensity profile

```

This second script is for finding the pixel width of a wire whose size (placed in the focus) whose physical size is known by a priori measurement.

```

1 ci = imread('argonionWireShot.jpg'); % Input the appropriate
   file name
2 c = improfile(ci,[1,320],[480,320],480); % The first vector is
   horizontal numbers, the second is vertical
3 % These are found by opening the image in the MATLAB
   command line with
4 % test = imread('filename.extension'); and imshow(test);
   then use

```

```

5      % imtool(test) and find the pixel numbers appropriate.
       These should be
6      % along a line that passes through the center of the
       brightest spot.
7  pixvals = linspace(1,480,480);
8  yvals = linspace(1,480,480);
9  relc = c(:,:,3);
10 width = 0;
11 startsmall = 0;
12 endsmall = 0;
13 for i=1:480 % The second number is based on the output of
       peaki once this is run the first time
14     yvals(i) = relc(i);
15     if (yvals(i) < 14)
16       if (i < 300) % Note that all the random numbers are
           found by analyzing
           % the picture initially with imtool
17       width = width + 1;
18     end
19   end
20   if (yvals(i) == 10)
21     if (i < 190)
22       startsmall = i;
23     end
24     if (i < 250)
25       endsmall = i;
26     end
27   end
28 end
29 end
30 width % output the pixel width of the wire
31 smallerwidth = endsmall - startsmall;
32 smallerwidth
33 plot(pixvals,yvals)

```

11.2 Time-of-flight data processor

Here is the script used for processing time-of-flight data from the .DAT files out of the PMT after using a waveform converter as detailed by Aaron [26].

```

1 %% This script takes in the first and last .DAT file numbers
       that are
2 % to be plotted and outputs the normalized, negative bias PMT
       signal as

```

```

3 % a function of time since ablation. Note that if some data
4 % manipulation
5 % is desired, the vectors 'actualTimes' and 'actualSumSig' can
6 % be called
7 % in the command window after this script finishes executing
8 %% The two inputs: the last two digits in the .DAT file names
9 % for both
10 % the first and the last file.
11 startFile = 15;
12 endFile = 17;
13 %% Working with the data, summing it for each time, then
14 % normalizing
15 % to the peak and accounting for any strange minimum behavior.
16 % The normalized PMT signal is slapped with a negative bias to
17 % replicate
18 % time-of-flight traces (which are the inputs here).
19 fileNum = endFile-startFile+1;
20 timeNum = 500; % Based on the number of data points in each .
21 % DAT file from
22 % our oscilloscope
23 A = zeros(timeNum, fileNum);
24 times = zeros(timeNum);
25 sumSig = zeros(timeNum);
26 for i=startFile:endFile
27     if i<10
28         M = csvread(strcat('TEK0000',int2str(i),'.DAT'));
29     end
30     if i>=10
31         M = csvread(strcat('TEK000',int2str(i),'.DAT'));
32     end
33     for j=1:timeNum
34         A(j,i+1) = M(j,2);
35         sumSig(j) = sumSig(j) + M(j,2);
36     end
37 end
38 % Accounting for when the signal actually begins based on time
39 % t=0 when
40 % our photodiode trigger goes off from an ablation pulse
41 for j=1:timeNum
42     times(j) = M(j,1);
43     if times(j)<0
44         starti = j;

```

```

38     end
39 end
40 starti = starti + 1;
41 totalNums = timeNum - starti + 1; % Running yields totalNums =
42     450
42 actualSumSig = linspace(1,totalNums,totalNums)*0.0;
43 actualTimes = linspace(1,totalNums,totalNums)*0.0;
44 minVal = 0;
45 maxVal = 0;
46 for k=1:totalNums
47     actualSumSig(k) = sumSig(timeNum - totalNums + k);
48     if actualSumSig(k) < maxVal
49         maxVal = actualSumSig(k);
50     end
51     if actualSumSig(k) > minVal
52         minVal = actualSumSig(k);
53     end
54     actualTimes(k) = times(timeNum - totalNums + k);
55 end
56 % Normalizing
57 maxVal = maxVal - minVal;
58 for k=1:totalNums
59     actualSumSig(k) = -1.0*(actualSumSig(k) - minVal)/maxVal;
60 end
61 %% Plotting the results
62 % hold on % For use when overlaying with other figures from
63 % the other
64 % models for example
64 actualTimes = actualTimes*10^6; % Scaling the times so that
65 % the graph is in microseconds
65 plot(actualTimes,actualSumSig,'LineWidth',2,'Color','Blue') %
66 % Adjust color as appropriate
66 xlabel('Time (\mu s)', 'FontSize',25);
67 ylabel('PMT signal (normalized, negative bias)', 'FontSize',25)
68 ;%
68 title('Comparing Data and First-Principles Model for 5 micron
69 % Spheres at a 0.8 mm Shooting Distance', 'FontSize',25); %
70 % Adjust title as appropriate
70 set(gca,'FontSize',20,'YTick',[-1.0:0.2:0.0], 'XGrid','on',
71 XMinorGrid','on');

```

11.3 Probabilistic model

Here, we provide the code used for the probabilistic model described in Section 4.4. Note that all three functions and the plotting script require inputs from the user. These are noted in the code.

This function calculates and returns the “hit efficiency” ζ for a given distance from the slide D , sphere velocity v and time t .

```
1 function [zeta] = zetaFactor(D, v, t)
2 % This function calculates and returns a corrective scaling
3 % factor
4 % to adjust the linear one-dimensional "spheres in the box"
5 % calculation in order to account for the three-dimensional
6 % spherical
7 % cone wavefront of the ejected spheres. Specifically, it
8 % returns
9 % the percent of spheres in the wavefront of velocity v that
10 % will be
11 % hit by the laser a distance D from the slide at time t.
12 % One of two parameters that can be changed (though it is
13 % likely a
14 % good value by default here): the maximum angle of ejection
15 % for the
16 % spherical cone. Currently, it is 20 degrees.
17 phi = pi/9.0;
18 % The second parameter that may be adjusted: the radius of
19 % the beam
20 % width
21 R = 100.0*10.0^-6.0;
22 % The distance the spheres in the wavefront of velocity v
23 % have
24 % travelled
25 d = v*t;
26 % The total surface area of the spherical wavefront that the
27 % spheres
28 % are covering
29 area = 2.0*pi*d^2.0*(1-cos(phi));
30 % The distance to the center of the beam
31 B = D + R;
32 % A value used in calculating ymin, the smallest value of y
33 % in the
34 % intersection
35 A = d^2.0+2*R*D+D^2.0;
```

```

26 %% A check on ensuring that the spherical wavefront is
27 % actually
28 % in the beam for at least some part (as this function is
29 % only called
30 % if this is true)
31     if D > d
32         d = D;
33     end
34 %% The angle formed with the outermost edge of the
35 % instersection
36     theta = acos(D/d);
37 %% A check on the angle to ensure it is less than the 20
38 % degree
39 % specified ejection angle
40     if theta > pi/9
41         theta = pi/9;
42     end
43 %% The minimum and maximum x values contained in the surface
44 % of
45 % intersection between the cylindrical beam and the spherical
46 % wavefront section
47     minx = -d*sin(theta);
48     maxx = d*sin(theta);
49 %% A numerical integration to calculate the total surface area
50 % of
51 % the intersection surface through a double integral that has
52 % been
53 % simplified to a single integral by performing one fo the
54 % integrals
55 % already
56 %% The step size for the numerical integral
57     dx = (maxx - minx) / 100.0;
58 %% The updating value of the total area of the surface of
59 % intersection
60     hit = 0.0;
61 %% for x=minx:dx:maxx
62     %% Calculating the minimum and maximum y-values at a
63     % given
64     % x value in the surface of intersection in order to
65     % make
66     % the calculation into a one-dimensional integral
67     miny = (A - x^2.0) / (2*B);

```

```

58      z = sqrt(d^2.0 - miny^2.0 - x^2.0);
59      beta = asin(z/d);
60      maxy = miny + d*(1-cos(beta));
61      %% Checks on the maximum y value, which should not go
       beyond
62      % the furthest edge of the beam nor should the total
       y
63      % distance covered between the start and end of the
64      % intersecting surface exceed the diameter of the
       beam
65      if maxy > (D + 2.0*R)
66          maxy = D + 2.0*R;
67      end
68      if maxy > (miny + 2.0*R)
69          maxy = miny + 2.0*R;
70      end
71      %% Updating the total surface area of the surface of
72      %% intersection
73      dhit1 = d*atan(maxy/sqrt(d^2.0-x^2.0-maxy^2.0));
74      dhit2 = d*atan(miny/sqrt(d^2.0-x^2.0-miny^2.0));
75      dhit = (dhit1 - dhit2)*dx;
76      hit = hit + abs(dhit);
77  end
78 %% Calculating the ratio of the area of the intersection
       surface to
79 % the total surface area of the section of the spherical cone
80 % wavefront that we are considering
81 zeta = abs(hit/area);
82 %% A check on the ratio. It should not exceed one, which may
       be
83 % possible due to the numerical integration
84 if zeta > 1.0
85     zeta = 1.0;
86 end
87 end

```

This function is a simple numerical integrator for the simplified Maxwell-Boltzmann distribution given in Equation 4.4.3. It calculates and returns the probability of finding a sphere between velocities v_1 and v_2 , and utilizes the most probable velocity in simplifying $f(v)$.

```

1 function [prob] = spheresPerInt(v1, v2)
2 %% This function returns the total probability density of a
       sphere

```

```

3 % having a velocity between v1 and v2 for a Boltzmann
   distribution
4 % with an expected value of V for the velocity
5 %% The one parameter that can be changed, the expected or mean
   value
6 % for the velocity of the spheres
7     V = 8.3;
8 %% The two constants for the Maxwell-Boltzmann distribution
9     A = sqrt((1.0/(pi*V^2.0))^3.0)*4.0*pi;
10    B = -1.0/(V^2.0);
11 %% A numerical integrator, once again following a simple
   Riemann
12 % summation technique with a left-hand sum approach
13 prob = 0.0;
14 delta = (v2 - v1)/100.0;
15 while v1 < v2
16     prob = prob + A*v1^2.0*exp(v1^2.0*B)*delta;
17     v1 = v1 + delta;
18 end
19 end

```

This function calculates and returns the percentage of the total number of spheres in the beam at any time during the laser shot if the laser is shot at time t a distance D from the sample. Essentially, this function implements Equation 4.4.25, setting $N = 100$ in order to directly calculate percentages.

```

1 function [num] = spheresIndrGap(D, t)
2 %% This function returns the number of spheres in the width of
3 % the laser beam a distance D from the slide fired at a time
   t
4 % for a duration of about 1 femto-second
5 %% Parameters of the laser: duration of the pulse dt and width
   of
6 % the beam w
7     dt = 1.0*10.0^-7.0;
8     w = 200.0*10.0^-6.0;
9 %% The minimum velocity necessary to reach the front edge of
   the
10 % beam between time t when the laser is fired and t + dt when
   the
11 % pulse ends
12     minv = D/(t + dt);
13 %% The maximum velocity allowed for a sphere to be at the back
   edge

```

```

14 % of the beam width at the start of the laser pulse at time t
15     maxv = (D + w) / (t * cos(pi/9));
16 %% Numerical integration using a lower summation method to
17 % determine
18 % the total number of spheres that will be hit by the laser
19     delta = (maxv - minv) / 100.0;
20     num = 0.0;
21     while minv < maxv
22         %% The adjustment scaling factor (0 to 1) necessary
23         % due to the
24         % spherical cone wavefront (calling the zetaFactor.m
25         % function)
26         zeta = zetaFactor(D, minv, t);
27         %% The number of spheres that would be in the one-
28         % dimensional
29         %% "box" if the spheres were ejected in a straight line
30         % . Note
31         % that spheresPerIntGaussian may be called here
32         % instead of
33         % spheresPerInt depending on the desired distribution
34         % . The
35         % current is a Maxwell-Boltzmann distribution, but
36         % the
37         % mentioned substitute follows a Gaussian
38         % distribution.
39         spheres = spheresPerInt(minv, minv + delta);
40         %% Updating the number of spheres, accounting for the
41         % change
42         %% from one-dimension to three
43         num = num + zeta * spheres;
44         minv = minv + delta;
45     end
46     num = num * 100.0;
47 end

```

This script uses the previous three functions to actually generate plots of the percentage of spheres that would be hit if the laser were shot at a given time as a function of that time, or the scattering light for a PMT signal.

```

1 %% This script plots the percent of spheres that will be hit
2 % by a
3 % laser a distance D from the slide as a function of time.
4 % It also
5 % returns the greatest percent of the spheres that can be hit

```

```

        and the
4 % time at which this would occur. That is , it returns the
5 % optimal
5 % firing time for the laser.
6 %% Parameters: D, the distance from the front edge of the beam
7 % (nearest
7 % to the slide) to the front edge of the slide (nearest to
8 % the beam)
8 D = 800.0*10^-6.0;
9 % The number of times to discretely sample with. A larger N
10 % increases
10 % the run-time significantly , but also allows for a more
11 % detailed and
11 % further time window to view it in.
12 N = 4500;
13 % The timescale on which the bulk of the "significant" action
13 % is
14 % expected to occur. Really , just change the values of N and
14 % the
15 % timescale until a good graph is obtained in which the peak
15 % and its
16 % decay are clearly visible.
16 timescale = 10.0^7.0;
18 %% The actual plotting , which calls upon the various other
18 % functions .
19 time = linspace(1 , N, N);
20 times = time/timescale;
21 percents = linspace(1 , N, N);
22 maxpercent = 0.0;
23 optimalt = 0.0;
24 for i=1:N
25     percents(i) = spheresIndrGap(D, times(i));
26     if percents(i) > maxpercent
27         maxpercent = percents(i);
28         optimalt = times(i);
29     end
30 end
31 for i=1:N
32     percents(i) = -1.0*percents(i)/maxpercent;
33 end
34 %% Creating the plot and printing the maximum percent of
34 % spheres

```

```

35 % that can be hit and the optimal time at which the laser
36 % should be
37 % fired in order to realize this maximum percentage. Note
38 % that
39 % the plot has on the y-axis the percent of spheres (0 to
40 % 100) that
41 % would be hit as a function of time (of the laser firing) on
42 % the
43 % x-axis.
44 maxpercent
45 optimalt
46 times = times*10^6; % Setting the timescale for plotting to
47 % microseconds
48 hold on % For use with overlaying on data or other model
49 % results
50 plot(times, percents, 'LineWidth', 2, 'Color', 'Black')
51 % xlabel('Time (\mu s)', 'FontSize', 25);
52 % ylabel('PMT signal (normalized, negative bias)', 'FontSize'
53 % , 25);
54 % title('Probabilistic Model for 500 nm Spheres at a 0.8 mm
55 % Shooting Distance', 'FontSize', 25);
56 % set(gca, 'FontSize', 20, 'YTick', [-1.0:0.2:0.0], 'XGrid', 'on',
57 % 'XMinorGrid', 'on');

```

11.4 Monte Carlo fitting

Here, we present the code used for the Monte Carlo simulations discussed in Section 4.5. The function requires inputs from the user and the script must be changed only if the timescale or timestep is changed and/or a different number of simulations are desired. Note crucially that this code does not run at all unless the randraw function is also in the same folder as it.

This function generates the simulated PMT voltage (normalized with a negative bias) as a function of time since ablation. It takes in all the relevant parameters before doing so.

```

1 function [ voltages ] = SimulationOneRun()
2 %SimulationOneRun Use with MultipleSimulationRuns to plot
3 %several Monte
4 %Carlo runs.
5 %% Parameters for the beam: radius, distance from the edge of
6 %the
5 % slide (before the spheres) to the center of the beam
6 R = 0.1*10^-3;

```

```

7      D = 0.9*10^-3;
8 % The radius of the sphere
9      r = 500*10^-9;
10 % Maximum ejection angle (20 degrees):
11      thetamax = pi/9;
12 %% More parameters
13 % The number of spheres (100 is convenient to make everything
14      a
15 % percentage:
16      N = 3251;
17 % The most probable velocity V and the appropriate scaling
18      factor
19 % for the Maxwell-Boltzmann distribution (change V as
20      necessary):
21      V = 8.3;
22      a = V/sqrt(2);
23 % Parameters for a Gaussian distribution of the ejection
24      angles, the
25 % mean and standard deviation (taking the maximum ejection
26      angle to
27 % be at three standard deviations from the mean, which is
28 % statistically viable):
29      mangle = 0;
30      stdangle = thetamax/3;
31 %% Setting up all the necessary vectors:
32      vels = linspace(1, N, N);
33      xpos = linspace(1, N, N);
34      ypos = linspace(1, N, N);
35      zpos = linspace(1, N, N);
36      phiangles = linspace(1, N, N);
37      thetaangles = linspace(1, N, N);
38 %% Determining the xpos and ypos of spheres from HCP packing
39      inside
40 % the HCP structure
41      RA = 30.0*10^-6; %Ablating radius
42      d = 2*RA;
43      endi = ceil(d/(2*r));
44      endj = ceil(d/(sqrt(3)*r));
45      counter = 1;
46      for j=1:endj
47          currentheight = RA - sqrt(3)*r*(j - 1);
48          for i=1:endi

```

```

43     if (mod(j , 2) == 1)
44         currentwidth = -1*RA + 2*r*i;
45     else
46         currentwidth = -1*RA + 2*r*i + r;
47     end
48     if (currentwidth^2 + currentheight^2 < RA^2)
49         xpos(counter) = currentwidth;
50         ypos(counter) = currentheight;
51         counter = counter + 1;
52     end
53 end
54 % Appropriately re-defining all of the vectors to fit the HCP
55 % (hexagonal close packing) structure assuming (reasonably to
56 % a degree)
57 % that the laser strikes a sphere straight on:
58 boltzmannvels = randraw('maxwell', a, N);
59 for i=1:counter
60     vels(i) = boltzmannvels(i);
61     zpos(i) = r;
62     thetaangles(i) = abs(normrnd(mangle, stdangle));
63     phiangles(i) = rand(1)*2*pi;
64 end
65 %% Doing the actual simulation
66 % The step size for the simulation
67 dt = 10^-7;
68 % The number of steps in the simulation
69 M = 4500;
70 % Setting up the times vector
71 times = linspace(1, M, M);
72 time = times/10^7;
73 % The PMT voltage that will be read out, which is a vector as
74 % it will be time dependent
75 voltages = linspace(1, M, M);
76 % A scale factor for the PMT voltage
77 scale = -1.0/3.0*10^-6;
78 % The cross-sectional area of the sphere
79 A = pi*r^2;
80 % The density of the spheres (in g/cm^3)
81 rho = 1.05;
82 % The mass of the sphere
83 m = pi*4/3*(r*100)^3*rho/1000;

```

```

84 % The parameter for the Gaussian beam width (at full-width
85 % half-max
86 stdR = R*sqrt(1/(2*log(2)));
87 % The y-direction acceleration due to gravity (in m/s/s)
88 a = 9.81;
89 % Running the simulation through some final time
90 maxvoltage = 0.0;
91 for i=1:M
92 voltages(i) = 0;
93 for j=1:counter
94 xpos(j) = xpos(j) + vels(j)*dt*sin(thetaangles(j))
95 *cos(phiangles(j));
96 ypos(j) = ypos(j) + vels(j)*dt*sin(thetaangles(j))
97 *sin(phiangles(j));
98 tempypos = ypos(j) - 1/2*a*(i*dt)^2;
99 zpos(j) = zpos(j) + vels(j)*dt*cos(thetaangles(j))
100 ;
101 dFC = sqrt(tempypos^2 + (zpos(j) - D)^2);
102 if (dFC <= R)
103 add = scale/(stdR*sqrt(2*pi))*exp(-dFC^2/(2*
104 stdR^2));
105 voltages(i) = voltages(i) + add;
106 end
107 end
108 if voltages(i)<maxvoltage
109 maxvoltage = voltages(i);
110 end
111 end
112 voltages = -1.0*voltages/maxvoltage;
113 end

```

This script uses the single Monte Carlo simulation function and plots three curves on the same axes for comparison of how the traces vary trial by trial.

```

1 %% Plots three Monte Carlo simulation traces
2 % Adjust the parameters for the simulation in the
3 % SimulationOneRun function
4 DataOne = SimulationOneRun();
5 DataTwo = SimulationOneRun();
6 DataThree = SimulationOneRun();
7 M = 4500; % Number of timesteps
8 times = linspace(1, M, M);
9 time = times/10^7; % Dividing by the timescale

```

```

9 time = time*10^6; % Setting the timescale for plots to
10 % microseconds
11 hold on % For use with overlays with data or other models
12 plot(time, DataOne, 'LineWidth', 2, 'Color', 'Magenta')
13 hold on
14 plot(time, DataTwo, 'LineWidth', 2, 'Color', 'Red')
15 hold on
16 xlabel('Time (\mu s)', 'FontSize', 25);
17 ylabel('PMT signal (normalized, negative bias)', 'FontSize', 25)
18 ;
19 title('Monte Carlo Simulations for 500 nm Spheres at a 0.8 mm
Shooting Distance', 'FontSize', 25);
20 set(gca, 'FontSize', 20, 'YTick', [-1.0:0.2:0.0], 'XGrid', 'on',
'XMinorGrid', 'on');

```

11.5 Zero free parameter model

Here, we present the code used for the first-principles model discussed in Section 5.

This script is the bulk of the model and calculates everything from the multitude of input values that it takes in, as described in Table 3. The most important outputs are Z, the silicon thickness array, and Zpmax, the ejection velocities array. This function has several useful plotting features outlined in detail at the bottom of the code that can be uncommented, copy and pasted into the command line of MATLAB and ran there to avoid the time delay of re-running the script for each different plot.

```

1 %% Constants throughout the computation. The absorption
2 % coefficient
3 % and index of refraction depend on the wavelength of the
4 % incident
5 % light (in vacuum). Here, we use the 532 nm wavelength of
6 % the green
7 % laser.
8 N = 500; % Number of partitions for the z axis.
9 M = 1000; % Number of partitions for the r axis.
10 T = 4000; % Total number of timesteps.
11 dt = 1.0*10^(-11.0); % Timestep used in the numerical
% simulation.
12 R = 50.0*10^(-6.0); % Total radial distance we will consider
% in the simulation.
13 FWHM = 24.0*10^(-6.0); % Beam waist FWHM of the green laser at

```

```

12 % its focus.
13 dr = R/M; % The radial width of each box.
14 Tamb = 294.5; % Room temperature at 70 degrees F
15 c = 299792458.0; % Speed of light in vacuum.
16 po = 2330.0; % Initial density of the silicon .
17 Nd = 1000.0; % Decimal number of partitions .
18 Zo = 500.0*10^(-6.0); % Initial thickness of the silicon .
19 n = 3.42; % Index of refraction .
20 vlight = c/n; % Speed of light in silicon .
21 a = 7.85*10^(5.0); % alpha , or the absorption coefficient .
22 Ap = 1.77*10^(-8.0); % The area hit by the green ablating
laser .
23 Ep = 3.25*10^(-4.0); % The total energy in the green
% ablating laser pulse .
24 s = 5.67*10^(-8.0); % Stefan-Boltzmann constant
% (really should be a sigma) .
25 sigz = 0.1; % Standard z deviation for green laser
% (if normalized to be from -0.5 to 0.5) .
26 sigr = FWHM/(2.0*sqrt(2.0*log(2.0))); % Standard r deviation
% for green laser
% (no normalization)
27 tp = 20.0*10^(-9.0); % The time duration of the green
% ablating laser pulse .
28 tdelay = 5.0*10^(-10); % The time delay before the laser hits .
29 npstart = ceil(tdelay/dt); % Starting time index for the pulse
.
30 npend = ceil((tp+tdelay+Zo/vlight)/dt); % End index for the
pulse .
31 Zdiff = vlight*tp; % Magnitude of the range of Z values for
the
% pulse .
32 Zmin = 0.0; % Used in the laser heat part: lowest Z value of
the
% laser .
33 Zmax = 0.0; % Used in the laser heat part: biggest Z value of
the
% laser .
34 Zavg = 0.0; % Used in the laser heat part: average Z value of
the
% laser .
35 %% "Constants" that vary with the temperature. Eventually
will be

```

```

46 % loaded in from a graph or data file into a vector.
47 aL = 4.86*10^(-6.0); % The coefficient for linear thermal
    expansion.
48 k = 131.0; % Thermal conductivity.
49 e = 0.5; % Emissivity.
50 cp = 700.0; % Specific heat capacity of the silicon.
51 %% Setting up the various matrices that will be used in the
52 % simulation.
53 u = zeros(N,M); % Temperature at a given z and r value.
54 uo = zeros(N,M); % "Old" temperature to keep track of the temp
    of
55 % each box before updating in the given
    timestep.
56 L = zeros(N,M); % Length at a given z and r value.
57 Z = zeros(T,M); % Thickness of the silicon at a given time, as
    a
        % function of radius.
58 Zp = zeros(T,M); % Rate of change of the silicon thickness at
    a given
        % time.
59 Zh = zeros(N,M); % For the laser heat part: accounts for
    relative
        % depth in the silicon wafer.
60 Zatt = zeros(N,M); % For the laser heat part: accounts for the
    % max thickness of the silicon as a
        function
        % of the radial distance of the box.
61 RADS = zeros(N,M); % Radius values. This is a constant matrix
    .
62 DTZ = zeros(N,M); % Used in the heat equation part. Second
    % derivative of the temperature with respect
    to z
    % at a given z and r.
63 DOR = zeros(N,M); % Used in the heat equation part. First
    % derivative of the temperature with respect
    to
    % r at a given z and r.
64 DTR = zeros(N,M); % Used in the heat equation part. Second
    % derivative of the temperature with respect
    to r
    % at a given z and r.
65 AMB = zeros(N,M); % Ambient temperature matrix. This will

```

have the

```
77 % ambient temperature in the top and bottom
    rows
78 % for the surfaces of the silicon and will
    be set
79 % equal to uo matrix elsewhere.
80 GammaMult = zeros(N,M); % For lensing
81 repeatDs = 0.0:5.0*10^(-6.0):50.0*10^(-6.0); % For lensing
82 %% Initializing the radius matrix.
83 RADS(:,1) = RADS(:,1) + dr/2.0;
84 for j=2:M
85     RADS(:,j) = RADS(:,j-1) + dr;
86 end
87 for i=1:N
88     %Use the following two for lensing
89     % GammaMult(i,:) = 75.0*pulstran(RADS(1,:),repeatDs,
    % @rectpuls,0.5*10^(-6.0));
90     % GammaMult(i,1:(50)) = GammaMult(i,1:50)*2.0;
91     %Use the following for non-lensing
92     GammaMult(i,:) = 1.0;
93 end
94 %% Initializing the lengths of the boxes.
95 L = L + 1.0;
96 L = L*Zo/N;
97 %% Initializing the temperature matrices to the appropriate
    values
98 % for the first iteration of the simulation.
99 u = u + Tamb;
100 uo = uo + u;
101 %% Running the simulation through all the timesteps. Note
    this will
102 % involve a lot of matrix operations to (hopefully) speed up
    what
103 % will otherwise take a very, very long time to run if many
104 % z, r and t partitions are used.
105 %% Running up until the laser begins to hit the silicon
    surface.
106 for t=1:(npstart-1)
107     %% Setting the values for Z and Zp as a function of radius
        , which
108     % are both constant as no heat is lost or gained before
        the
```

```

109      % laser interaction. Note Zp just stays zero so there is
110          no
110      % need to update it.
111      Z(t,:) = Z(t,:)+Zo;
112  end
113 %% Running through the time in which the laser is interacting
114      with
114      % the silicon.
115  for t=npstart:npend
116      %% Updating the temperature.
117      % Setting up the ambient temperature matrix.
118      AMB = uo;
119      AMB(1,:) = AMB(1,:).*0.0 + Tamb;
120      AMB(N,:) = AMB(N,:).*0.0 + Tamb;
121      % Calculating the partial derivative with respect to z for
122          each
122          % box.
123      DTZ(1,:) = 8.0*((uo(2,:)-uo(1,:))/(L(2,:)+L(1,:))-(uo
124          (1,:)-AMB(1,:))/L(1,:))/(L(1,:)+2.0*L(2,:));
124      DTZ(N,:) = 8.0*((AMB(N,:)-uo(N,:))/L(N,:)-(uo(N,:)-uo(N
125          -1,:))/(L(N,:)+L(N-1,:)))./(L(N,:)+2.0*L(N-1,:));
125  for i=2:(N-1)
126      DTZ(i,:) = 4.0*((uo(i+1,:)-uo(i,:))/(L(i+1,:)+L(i,:))
127          -(uo(i,:)-uo(i-1,:))/(L(i,:)+L(i-1,:)))./(L(i+1,:)
128          +L(i-1,:));
128 end
129      % Calculating the partial derivative with respect to r for
130          each
130          % box. We calculate both the first and second derivatives
130          .
130      DOR(:,1) = (u(:,2)-u(:,1))/(dr*2.0);
131      DOR(:,M) = (u(:,M)-u(:,M-1))/(dr*2.0);
132      for j=2:(M-1)
133          DOR(:,j) = (u(:,j+1)-u(:,j-1))/(dr*2.0);
134      end
135      DTR(:,1) = 2.0*DOR(:,1)/dr;
136      DTR(:,M) = -2.0*DOR(:,M)/dr;
137      for j=2:(M-1)
138          DTR(:,j) = (u(:,j+1)-2.0*u(:,j)+u(:,j-1))/dr^(2.0);
139      end
140      % Updating the temperature from the heat diffusion effects
141      u = u + dt*k*Zo*(DTZ+DOR./RADS+DTR)/(cp*N*po)./L;

```

```

142 % Accounting for thermal radiation .
143 u = u - Zo*e*s*dt*(uo-AMB).^(4.0)/(cp*po*N)./(L.^(2.0));
144 % Setting up the Zh matrix .
145 Zh(N,:) = L(N,:);
146 for i=1:(N-1)
147     Zh(N-i,:)=Zh(N-i+1,:)+L(N-i,:);
148 end
149 % Accounting for the heat input from the green laser .
150 for i=1:N
151     Zatt(i,:)=Z(t-1,:);
152 end
153 Zmin = Zo - vlight*(t*dt-tdelay);
154 Zmax = Zmin + vlight*tp;
155 Zavg = (Zmin + Zmax)/2.0;
156 % Have a factor of 0.97 for 5 micron and 0.997 for other
157 % sizes (that
158 % comes from 500 nm, but is similar for others)
159 u = u + 0.997*0.9066*GammaMult.*L.^(-2.0)*2.0*n*Ep*Zo
160    /((1.0+2.0*n+n^2.0)*tp*Ap*cp*N*po*(sigr/R)*sigz*pi)*dt
161    .*exp(-1.0*((Zh-Zavg)/Zdiff).^2.0/(2.0*sigz^2.0)).*exp
162    (-1.0*RADS.^2.0/(2.0*sigr^2.0)).*exp(-a*(Zatt-Zh))
163    .*(-1.0*exp(-a*L)+1.0);
164 %% Updating the lengths .
165 L = L + aL*(u-uo).*L;
166 %% Calculating and storing the silicon thickness as a
167 % function
168 %% of radius .
169 Z(t,:)=sum(L);
170 %% Calculating and storing the rate of expansion or
171 % compression
172 %% of the silicon surface as a function of radius .
173 Zp(t,:)=(Z(t,:)-Z(t-1,:))/dt;
174 %% Updating the old temperature for the next time
175 % iteration .
176 uo = u;
177 end
178 %% Running through the time the laser has ceased interaction
179 % with
180 %% the surface until the end of the simulation .
181 for t=(npnd+1):T
182     %% Updating the temperature .
183     %% Setting up the ambient temperature matrix .

```

```

175    AMB = uo;
176    AMB(1,:) = AMB(1,:).*0.0 + Tamb;
177    AMB(N,:) = AMB(N,:).*0.0 + Tamb;
178    % Calculating the partial derivative with respect to z for
        each
179    % box.
180    DTZ(1,:) = 8.0*((uo(2,:)-uo(1,:))/(L(2,:)+L(1,:))-(uo
        (1,:)-AMB(1,:))/L(1,:))/(L(1,:)+2.0*L(2,:));
181    DTZ(N,:) = 8.0*((AMB(N,:)-uo(N,:))/L(N,:)-(uo(N,:)-uo(N
        -1,:))/(L(N,:)+L(N-1,:)))/(L(N,:)+2.0*L(N-1,:));
182    for i=2:(N-1)
183        DTZ(i,:) = 4.0*((uo(i+1,:)-uo(i,:))/(L(i+1,:)+L(i,:))
            -(uo(i,:)-uo(i-1,:))/(L(i,:)+L(i-1,:)))/(L(i+1,:)
            +L(i-1,:));
184    end
185    % Calculating the partial derivative with respect to r for
        each
186    % box. We calculate both the first and second derivatives
        .
187    DOR(:,1) = (u(:,2)-u(:,1))/(dr*2.0);
188    DOR(:,M) = (u(:,M)-u(:,M-1))/(dr*2.0);
189    for j=2:(M-1)
190        DOR(:,j) = (u(:,j+1)-u(:,j-1))/(dr*2.0);
191    end
192    DTR(:,1) = 2.0*DOR(:,1)/dr;
193    DTR(:,M) = -2.0*DOR(:,M)/dr;
194    for j=2:(M-1)
195        DTR(:,j) = (u(:,j+1)-2.0*u(:,j)+u(:,j-1))/dr^(2.0);
196    end
197    % Updating the temperature from the heat diffusion effects
        .
198    u = u + dt*k*Zo*(DTZ+DOR./RADS+DTR)/(cp*N*po)./L;
199    % Accounting for thermal radiation .
200    u = u - Zo*e*s*dt*(uo-AMB).^(4.0)/(cp*po*N)./(L.^2.0));
201    %% Updating the lengths .
202    L = L + aL*(u-uo).*L;
203    %% Calculating and storing the silicon thickness as a
        function
204    % of radius .
205    Z(t,:)= sum(L);
206    %% Calculating and storing the rate of expansion or
        compression

```

```

207      % of the silicon surface as a function of radius.
208      Zp(t,:) = (Z(t,:)-Z(t-1,:))/dt;
209      %% Updating the old temperature for the next time
210      % iteration .
211      uo = u;
212  end
213 %% Note in the following two sections that only one of the
214 %% portions should be uncommented (comment out one of the
215 %% section's
216 %% plot commands to view whichever data you want to look at).
217
218 %% Plotting the silicon thickness as a function of radius and
219 %% time.
220 %% To do this, we first create a couple more matrices.
221
222 timespace = zeros(T,M);
223 radspace = zeros(T,M);
224 for t=1:T
225     radspace(t,:) = RADS(1,:);
226 end
227 for j=1:M
228     timespace(:,j) = linspace(1,T,T)*dt;
229 end
230
231 %% Heat map
232 % fig = figure;
233 % colormap('default');
234 % imagesc(RADS(1,:)*10^6,linspace(1,T,T)*dt*10^9,Z*10^6);
235 % colorbar;
236 % ylabel(colorbar,'Silicon Thickness (\mu m)','FontSize',25);
237 % xlabel('Radius (\mu m)', 'FontSize',25);
238 % ylabel('Time (ns)', 'FontSize',25);
239 % title('Silicon Surface Expansion: Heat Map','FontSize',25);
240 % set(gca,'FontSize',20);
241
242 %% Mesh grid view
243 % mesh(timespace*10^9,radspace*10^6,Z*10^6);
244 % xlabel('Time (ns)', 'FontSize',25);
245 % ylabel('Radius (\mu m)', 'FontSize',25);
246 % zlabel('Silicon Thickness (\mu m)', 'FontSize',25);
247 % title('Silicon Surface Expansion with Ball Lensing','

```

```

        FontSize',25);
245 % set(gca,'FontSize',20);
246 % view(-115,25) % for the overall view
247 % view(0,0) % for the time view
248 % view(90,0) % for the radial view
249
250 %% Finding and plotting the ejection velocities of the spheres
251 % as a function of radius, as well as ejection angles
252 [Zpmax, maxind] = max(Zp);
253
254 thetadist = linspace(1,M,M)*0.0;
255 for i=2:(M-1)
256     thetadist(i) = atan((Z(maxind(i),i-1)-Z(maxind(i),i+1))
257                         /(2.0*dr));
258 end
259
260 %% Ejection velocity plot
261 % plot(RADS(1,:)*10^6,Zpmax,'LineWidth',3)
262 % xlabel('Radius (\mu m)', 'FontSize',25);
263 % ylabel('Ejection Velocity (m/s)', 'FontSize',25);
264 % title('Silicon Surface Expansion with Ball Lensing: Ejection
265 % Velocities', 'FontSize',25);
266 % set(gca,'FontSize',20);
267
268 %% Ejection angle plot
269 % plot(RADS(1,:)*10^6,thetadist*10^3,'LineWidth',3)
270 % xlabel('Radius (\mu m)', 'FontSize',25);
271 % ylabel('Ejection Angle \theta (milliradians)', 'FontSize',25)
272 % ;
273 % title('Silicon Surface Expansion After Ablation: Ejection
274 % Angles', 'FontSize',25);
275 % set(gca,'FontSize',20);

```

To convert the model into either a simulated PMT signal or to analyze what it means for sphere targeting with the high-intensity laser, use the following function after running the model script. Call this function in the command line with SurfaceExpansionToPMT(RADS(1,:),Zpmax) and hit enter. It will then spit back the appropriate plot as selected in the bottom of the function itself.

```

1 function [ ] = SurfaceExpansionToPMT( radii , ejectvels )
2 %% To run this function, first run ExpansionMatrix3D script
3 % and then run
4 % this function using the inputs of RADS(1,:) for radii and
5 % Zpmax for

```

```

4 % ejectvels. This function returns the simulated PMT signal (
5   normalized,
6 % negative bias) as a function of time since ablation OR the
7   number of
8 % spheres in the high-intensity laser focus as a function of
9   time since
10 % ablation, depending on the input parameters. There are
11   various things
12 % that must be commented/uncommented depending on if using the
13 % high-intensity laser or the scattering beam
14 %% Input values
15 Nspheres = 36254; % Set this equal to the value of counter+1
16   after running once
17 rspheres = 250.0*10^(-9.0); % The radius of the deposition
18   spheres
19 Rscatter = 5*10^(-6.0); % Radius of the scattering beam or
20   THOR focal sphere
21   % 5E-6 is THOR and 100E-6 is
22   % scattering
23 Dscatter = 505*10^(-6.0); % Distance from the sample surface
24   to the center of
25   % the scattering beam or high-
26   % intensity focus
27   % Use 600E-6 for scattering at 0.5
28   % nm shooting
29   % distance or 505E-6 for THOR
30 Ntheta = 0.0; % The misalignment angle to test sensitivity to
31   alignment
32 timenum = 6000; % The number of timesteps to consider. Use
33   4500 for replicating
34   % time-of-flight traces
35 timestep = 10^(-7.0); % The timestep. Use 10E-7 for
36   replicating time-of-flight
37   % traces with the timenum of 4500
38 %% Running everything
39 scattersigR = Rscatter/(2.0*sqrt(2.0*log(2.0)));
40
41 vels = zeros(1,Nspheres);
42 xpos = zeros(1,Nspheres);
43 ypos = zeros(1,Nspheres);
44 zpos = zeros(1,Nspheres);
45
46

```

```

32 rablate = 50.0*10^(-6.0);
33 dablate = 2.0*rablate;
34 endi = ceil(dablate/(2.0*rspheres));
35 endj = ceil(dablate/(sqrt(3.0)*rspheres));
36 counter = 1;
37 for j=1:endj
38     currentheight = rablate - sqrt(3.0)*rspheres*(j-1);
39     for i=1:endi
40         currentwidth = 0.0;
41         if (mod(j,2)==1)
42             currentwidth = -1.0*rablate+2.0*rspheres*i;
43         else
44             currentwidth = -1.0*rablate+2.0*rspheres*i+
45                 rspheres;
46         end
47         currentrad = sqrt(currentwidth^2.0+currentheight^2.0);
48         if (currentrad<rablate)
49             xpos(counter) = currentwidth;
50             ypos(counter) = currentheight;
51             tempMatrix = abs(radial-currentrad);
52             [tempval, tempind] = min(tempMatrix);
53             % Use the following velocity in general
54             vels(counter) = ejectvels(tempind);
55             % Use this for 5 micron spheres with the
56             % coefficient of the
57             % exponential determined from the peak of the
58             % lensing ejection
59             % velocity curve from ExpansionMatrix3D
60             vels(counter) = 270*exp(-currentrad
61             ^2/(1.5*10^(-5))^2);
62             counter = counter + 1;
63         end
64     end
65 end
66 counter = counter-1;
67 counter
68 timevals = linspace(1,timenum,timenum)*timestep;
69 signalout = zeros(1,timenum);
70 theta=Ntheta*pi/180;
71 for t=1:timenum
72     zpos = zpos + vels*cos(theta)*timestep;
73     xpos = xpos + vels*sin(theta)*timestep;

```

```

70      for j=1:counter
71          dFC = sqrt((ypos(j) - 0.5*9.81*(t*timestep)^2.0)^2.0 +
72              xpos(j)^2.0+(zpos(j)-Dscatter)^2.0); % For the high
73              -intensity laser spherical spot
72  %      dFC = sqrt((ypos(j) - 0.5*9.81*(t*timestep)^2.0)^2.0 +
73              zpos(j)-Dscatter)^2.0); % For the CW scattering beam solid
74              cylinder
75      if dFC <= Rscatter
76          signalout(t) = signalout(t) + 1.0; % For the
77              high-intensity laser
78      signalout(t) = signalout(t) -
79      exp(-1.0*dFC^2.0/(2.0*scattersigR^2.0)); % For
80      the CW laser
81      end
82      end
83  end
84  %% For the scattering time-of-flight
85  % signalout = -1.0*signalout/min(signalout);
86  % edgetime = 0;
87  % found = 0;
88  % for t=1:timenum
89  %     if signalout(t)<-0.995&&found==0
90  %         edgetime = t*timestep;
91  %         found=1;
92  %     end
93  % end
94  % edgetime % Spitting out the peak time
95  %% For the high-intensity laser
96  started = 0;
97  ended = 0;
98  for t=1:timenum
99      if signalout(t)>1 && started==0
100         started = t;
101     end
102     if signalout(t)>0
103         ended = t;
104     end
105 end
106 (ended-started)*timestep % Spitting out the total time window
107 a sphere could be hit in
108 %% Plotting (only use one setup)
109
110
```

```

105 % For high-intensity
106 hold on % For use with other graphs or multiple runs of this
    function
107 plot(timevals*10^6, signalout, 'LineWidth',2,'Color','Blue') %
    Change the color as needed
108 xlabel('Time (\mu s)', 'FontSize',25);
109 title('500 nm Spheres in the High-Intensity Focus at Time
    Since Ablation for Various Ablation Intensities', 'FontSize'
    ,25); % Change title as needed
110 ylabel('Spheres in the focus', 'FontSize',25);
111 set(gca,'FontSize',20,'XGrid','on','XMinorGrid','on');
112
113 % For scattering
114 % hold on % For use with other graphs or multiple runs of this
    function
115 % plot(timevals*10^6, signalout, 'LineWidth',2,'Color','Blue')
    % Change the color as needed
116 % xlabel('Time (\mu s)', 'FontSize',25);
117 % title('Simulated Time-of-Flight Data from First-Principles
    Model', 'FontSize',25);
118 % ylabel('PMT signal (normalized, negative bias)', 'FontSize
    ',25);
119 % set(gca,'FontSize',20,'YTick',[-1.0:0.2:0.0], 'XGrid','on',
    'XMinorGrid','on');
120 end

```

11.6 Ball lensing

Here, we present the code used for the ball lensing model. None of these codes take in parameters other than the number of partitions to use.

This function produces the data for the mapped light intensity on our silicon surface OR for the total power absorbed as a function of normalized radius.

```

1 function [datavec] = SphereLensingDataProducer(N,M)
2 %UNTITLED2 Summary of this function goes here
3 % Detailed explanation goes here
4 R = 1;
5 deltar = R/M;
6 na = 1.0; % index of refraction of air
7 ns = 1.615; % index of refraction of polystyrene
8 nui = 1.0/pi; % for normalizing to a total input power of 1
9 thetavals = linspace(1,N,N);
10 nuivals = linspace(1,N,N);

```

```

11 nufvals = linspace(1,N,N)*0.0;
12 xivals = linspace(1,N,N);
13 xfvals = linspace(1,N,N);
14 rvals = linspace(1,N,N);
15 energies = linspace(1,N,N)*0.0;
16 sumenergies = linspace(1,N,N)*0.0;
17 for i=1:N
18     rvals(i) = ((i-1.0)/M)*R;
19     nuivals(i) = nui;
20     thetavals(i) = ((i-1.0)/M)*(pi/2.0);
21     ctheta = thetavals(i);
22     xivals(i) = -1.0*R*sin(ctheta);
23     alpha = asin(na*sin(ctheta)/ns);
24     L = 2.0*R*cos(alpha);
25     deltaxc = L*sin(ctheta-alpha);
26     H = R*(1.0-cos(2.0*alpha-ctheta));
27     deltaxa = H*tan(2.0*alpha);
28     if ctheta >= 2.0*alpha
29         deltaxa = H*tan(2.0*ctheta-3.0*alpha);
30     end
31     xfvals(i) = xivals(i) + deltaxc + deltaxa;
32     if (2.0*ctheta-3.0*alpha) >= (pi/2.0)
33         xfvals(i) = 2.0*R;
34     end
35     if i>1
36         energies(i) = nui*pi*(xivals(i)^2-xivals(i-1)^2);
37     end
38 end
39 for i=1:N
40     xfvals(i) = abs(xfvals(i));
41 end
42 for i=1:N
43     for j=1:N
44         if xfvals(i) >= rvals(j) && xfvals(i) < (rvals(j) +
45             deltar)
46             nufvals(j) = nufvals(j) + energies(i);
47         end
48     end
49 end
50 for i=1:N
51     for j=1:i
52         sumenergies(i) = sumenergies(i) + nufvals(j);

```

```

52     end
53 end
54 nufvals(1) = nufvals(1) / (pi*rvals(2)^2);
55 for j=2:N-1
56     nufvals(j) = nufvals(j) / (pi*(rvals(j+1)^2-rvals(j)^2));
57 end
58 nufvals = nufvals/nuivals(1);
59 % In the second slot of datavec, use nufvals for the mapped
    intensity and
60 % sumenergies for the total power absorbed as a function of
    radius
61 datavec = [rvals; nufvals];
62 end

```

This script uses the data producing function to display how the results change based on the number of partitions used.

```

1 %% Takes in the output of the data from the
    SphereLensingDataProducer
2 % and produces pretty pictures
3 makenums = [100 1000 10000]; % The partition numbers to try
4 makenumsdec = [99.0 999.0 9999.0]; % Partition numbers minus
    one
5 endnums = makenums*2/10; % for what fraction of the normalized
    radius (0 to 1) to view
6 hold on
7 for i=1:length(makenums)
8     cdata = SphereLensingDataProducer(makenums(i),makenumsdec(
        i));
9     plot(cdata(1,1:endnums(i)),cdata(2,1:endnums(i)),'
        LineWidth',1.5)
10 end
11 % Plotting the output of SphereLensingDataProducer for the
    different
12 % partition numbers
13 xlabel('Radial distance (normalized)', 'FontSize', 25);
14 ylabel('Intensity (normalized)', 'FontSize', 25);
15 set(gca, 'FontSize', 20);
16 set(gcf, 'Units', 'Normalized', 'OuterPosition', [0 0 1 1]);
17 axis([0 0.2 0 500]);
18 [hleg1,hobj1] = legend('N=100','N=1000','N=10000');
19 textobj = findobj(hobj1, 'type', 'text');
20 set(textobj, 'Interpreter', 'latex', 'fontsize', 30);
21 set(hleg1, 'position',[0.7 0.7 0.2 0.2]);

```

```
22 set(hobj1, 'linewidth',4);
```

This script produces the mapped intensity data similar to the first ball lensing function, except that it is set to output a heat map of the log of the normalized intensity.

```
1 N = 1000;
2 M = 999.0;
3 R = 1;
4 deltar = R/M;
5 na = 1.0;
6 ns = 1.615;
7 nui = 1.0;
8 thetavals = linspace(1,N,N);
9 nuivals = linspace(1,N,N);
10 nufvals = linspace(1,N,N);
11 xivals = linspace(1,N,N);
12 xfvals = linspace(1,N,N);
13 rvals = linspace(1,N,N);
14 energies = linspace(1,N,N);
15 Z = zeros(N+1,N+1);
16 for i=1:N
17     rvals(i) = ((i-1.0)/M)*R;
18     nuivals(i) = nui;
19     nufvals(i) = 0.0;
20     thetavals(i) = ((i-1.0)/M)*(pi/2.0);
21     ctheta = thetavals(i);
22     xivals(i) = -1.0*R*sin(ctheta);
23     alpha = asin(na*sin(ctheta)/ns);
24     L = 2.0*R*cos(alpha);
25     deltaxc = L*sin(ctheta-alpha);
26     H = R*(1.0-cos(2.0*alpha-ctheta));
27     deltaxa = H*tan(2.0*alpha);
28     if ctheta >= 2.0*alpha
29         deltaxa = H*tan(2.0*ctheta-3.0*alpha);
30     end
31     xfvals(i) = xivals(i) + deltaxc + deltaxa;
32     if (2.0*ctheta-3.0*alpha) >= (pi/2.0)
33         xfvals(i) = 2.0*R;
34     end
35     if i > 1
36         energies(i-1) = nui*pi*(xivals(i)^2-xivals(i-1)^2);
37     end
38 end
```

```

39  for i=1:N
40      xfvals(i) = abs(xfvals(i));
41  end
42 energies(N) = 0;
43 for i=1:N
44     for j=1:N
45         if xfvals(i) >= rvals(j) && xfvals(i) < (rvals(j) +
46             deltar)
47             nufvals(j) = nufvals(j) + energies(i);
48         end
49     end
50     for j=1:N
51         nufvals(j) = nufvals(j) / (pi*(2.0*rvals(j)*deltar+deltar^2)
52             );
53         rvals(j) = rvals(j)/R;
54     end
55     for i=1:N+1
56         for j=1:N+1
57             xnorm = ((i-1.0)-N/2.0)/(N/2.0);
58             ynorm = ((j-1.0)-N/2.0)/(N/2.0);
59             cr = sqrt(xnorm^2 + ynorm^2);
60             z(i,j) = 1.0;
61             for m=1:N-1
62                 if cr >= rvals(m) && cr < rvals(m+1)
63                     z(i,j) = z(i,j) + nufvals(m);
64                 end
65             end
66             z(i,j) = log(z(i,j));
67         end
68     end
69 fig = figure;
70 colormap('default');
71 imagesc(z);
72 colorbar;

```

11.7 Fitting graphical-user-interface

We will not even bother attempting to put any code in this thesis for the MATLAB fitting GUI that we created. It requires a significant amount of helper functions, both that we wrote and that can be found easily for GUIs. These will all be in both the group's Sakai resources folder and on the group's GitHub account shortly af-

ter this thesis is turned in.

Once everything necessary to run the fitting GUI has been downloaded, the user needs to do only one thing to run it. That is to open the script (NOT the figure) titled “FirstGUI.m” and run it. Then, a figure should pop up that can be interfaced with as detailed in Section 4.6. a sample of this window is seen in Figure 4.6.1.

It is important to note that in fact every file in the zipped folder for this GUI is required to run it. MATLAB GUIs are not exactly intuitive, so if the user hopes to edit its functionality, we suggest watching a couple online help videos first. Nonetheless, here is a very brief rundown of what to change if the user wishes to modify it:

- (i) The “FirstGUI.fig” file: if the user wishes to modify the interface itself, this is what to open to do it. It does not run the GUI, but changes made here will stay when the user opens and runs the corresponding script. So here is where the user could delete plots, add more ones, change inputs, change outputs, change the layout, et cetera.
- (ii) The “FirstGUI.m” file: if the user wishes to modify what actually happens in the GUI once the user has opened it and is interacting with it, this script file is where to change things. It is very nonintuitive but can be learned in part by looking through what we already have there.
- (iii) Any of the helper functions called in the plotting portions of “FirstGUI.m”: these functions are where the actual fitting mechanism and data processing can be adjusted to suit the user’s fancy. However, as a word of caution, the user should probably stick to only modifying functions we wrote. We have not tested changing any of the other functions that come with the usage of the GUI, and the user should probably steer clear of this as well, although there will always be the failsafe on the web provided no one erases it. So the functions that can be experimented with at the user’s delight are: “getDataFromFiles.m”, “calcFit.m”, “avgSim.m”, and “simFunction.m”. The primary one to edit would probably be the latest one.