

THE PENNSYLVANIA STATE UNIVERSITY  
SCHREYER HONORS COLLEGE

DEPARTMENT OF BIOMEDICAL ENGINEERING

MODELING THE BIDIRECTIONAL TRANSPORT OF INTRACELLULAR CARGOS BY  
OPPOSITELY-DIRECTED MOTOR PROTEINS

BRANDON MENTLEY  
FALL 2015

A thesis  
submitted in partial fulfillment  
of the requirements  
for a baccalaureate degree  
in Bioengineering  
with honors in Bioengineering

Reviewed and approved by the following:

Dr. William Hancock  
Professor of Biomedical Engineering  
Thesis Supervisor  
Honors Adviser

Dr. John Fricks  
Professor of Statistics  
Faculty Reader

\*Signatures are on file in the Schreyer Honors College

## Abstract

Transport of intracellular cargos is achieved through the combined efforts of kinesin and dynein motor proteins. This transport is characterized by periods of processive movement, as well as frequent pauses and switches in direction. Currently, the interactions between kinesins and dyneins bound to the same cargo are described as a direct mechanical competition between the two motor protein superfamilies. However, this “tug-of-war model” fails to explain experimental observations that show that inhibiting either kinesin or dynein results in reduced motility for both motor species. Three additional models have been proposed in an attempt to overcome the shortcomings of the tug-of-war model. The “microtubule tethering model” proposes a diffusive state in which motors are loosely associated with the microtubule, thus anchoring the cargo while providing minimal resistance to stepping by oppositely-directed motors. The “mechanical activation model” states that motor proteins tend to remain in an inhibited state unless a hindering force, such as the force exerted by an oppositely-directed motor, is applied. Lastly, the “steric disinhibition model” states that motor proteins tend to remain in an autoinhibited state unless they are bound at some site by oppositely-directed motors or regulatory proteins. This goal of this project was to produce computational simulations of the competing models describing bidirectional transport by kinesin and dynein. Analysis of these simulations has already revealed that the mechanical activation model displays a number of characteristics seen in experimental data, including pauses and, most importantly, the paradox of codependence, which is the phenomenon that inhibition of one motor species results in reduced motility in both directions. In the future, the simulations will be used to determine the direction of experiments that will hopefully validate one of the proposed models and lead to a greater understanding of bidirectional transport.

## TABLE OF CONTENTS

List of Figures .....	iii
Chapter 1: Introduction .....	1
Chapter 2: Methods .....	10
2.1: <i>Kinesin Parameters</i> .....	11
2.2: <i>Dynein Parameters</i> .....	12
2.3: <i>Tug-of-War Simulation</i> .....	17
2.4: <i>Microtubule Tethering Simulation</i> .....	18
2.5: <i>Mechanical Activation Simulation</i> .....	19
2.6: <i>Steric Disinhibition Simulation</i> .....	21
Chapter 3: Results/Discussion .....	24
3.1: <i>Initial testing</i> .....	24
3.2: <i>Comparison of Proposed Models</i> .....	28
3.2.1: Tug-of-War Model .....	28
3.2.2: Microtubule Tethering Model .....	30
3.2.3: Mechanical Activation Model .....	31
3.2.4: Steric Disinhibition Model .....	34
3.3: Analysis of Mechanical Activation Model .....	35
3.3.1: Kinesin-1 .....	35
3.3.2: Kinesin-2 .....	38
Chapter 4: Conclusion/Future Work .....	41
References .....	43
Appendix A: MATLAB Code for Mechanical Activation Model .....	46

## List of Figures

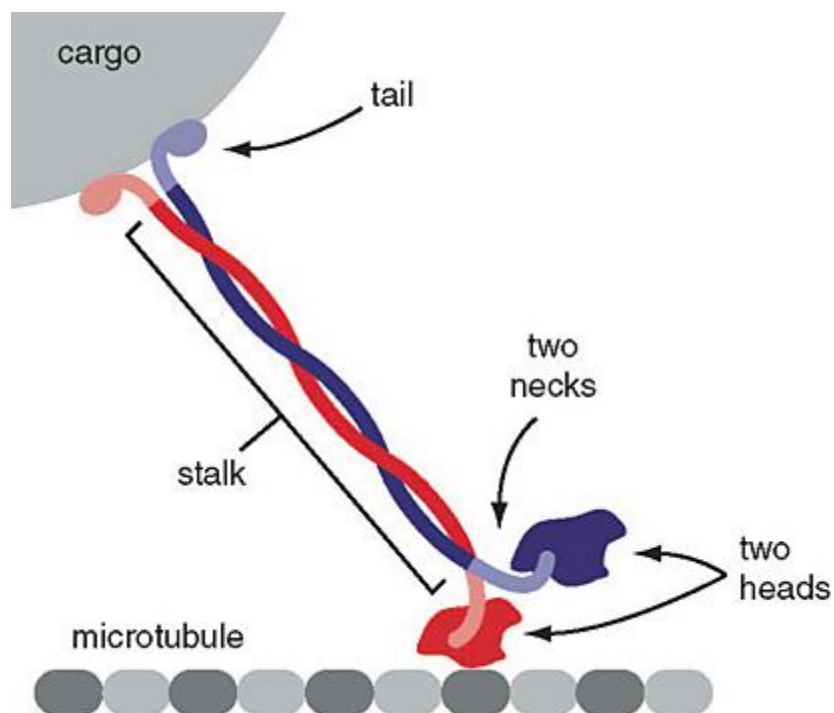
Figure 1. Simplified kinesin structure. Figure adapted from Asbury, et al. <sup>1</sup> .....	2
Figure 2. Computer-generated representation of dynein's structure. Figure adapted from Vale. <sup>3</sup> .....	3
Figure 3. Experimental data displaying the paradox of codependence. A: Inhibiting kinesin motors either by knocking out kinesin light chain 1 (KLC1-/-) or by knocking down light chain 2 (KLC2 shRNA) results in shorter run lengths and more frequent pauses during retrograde transport. B: Inhibiting dynein in mouse neurons results in more time spent in the paused state and less time spent in both retrograde and anterograde transport.....	5
Figure 4. Three models proposed to explain the codependence between oppositely-directed motor proteins. Figure adapted from Hancock. <sup>4</sup> .....	7
Figure 5. Force-velocity relationship for mammalian cytoplasmic dynein in 10 $\mu$ M and 1 mM ATP. For the purposes of this project, the curve at 1 mM ATP was used. Figure adapted from Toba, et al. <sup>27</sup> .....	13
Figure 6. Dwell time for the detachment of dynein vs. force. Figure adapted from Kunwar, et al. <sup>11</sup> .....	15
Figure 7. Visual Representation of Microtubule Tethering Model.....	19
Figure 8. Attachment Rate vs. Cargo Velocity for Mechanical Activation Simulation. Both motor species reach their minimum attachment rate at 0 velocity and reach their maximum at the average unloaded velocity of the opposing motor. ....	21
Figure 9. Kinesin Attachment Rate vs. Number of Dynein Bound to Cargo for Steric Disinhibition Model. Kinesin's attachment rate increases linearly with the number of dynein bound to the cargo before reaching a maximum when 5 dynein motors are bound.....	23
Figure 10. Kinesin-1 velocity vs. force for 10 $\mu$ M ATP (in red) and 1 mM (in blue). The data for 1mM ATP matches closely with simulation results. Figure adapted from Carter and Cross. <sup>25</sup>	24
Figure 11. Simulated force-velocity relationship for a single kinesin motor. The velocity at forces ranging from -10 to 10 pN was determined from the average of 3000 simulations. The graph reveals a velocity of approximately 800 nm/s for the unloaded case and for assisting loads.	25
Figure 12. Simulated force-velocity relationship for a single dynein motor using the dynein 1 model. The velocity at forces ranging from -10 to 10 pN was determined from the average of 3000 simulations. The graph reveals a velocity of approximately -200 nm/s for the unloaded case and for assisting loads.....	26
Figure 13. Simulated force-velocity relationship for a single dynein motor using the dynein 1 model. This graph reveals a larger sensitivity to load force than was seen in the dynein 1 model.....	27
Figure 14. Simulated position vs. time for tug-of-war model with a single kinesin motor pulling against six dynein motors. Each line represents a single run that was allowed to continue for 10 seconds or until all motors became detached from the microtubule. ....	29

- Figure 15. Simulated position vs. time for microtubule tethering model with a single kinesin motor pulling against six dynein motors. In the microtubule tethering model, motors pass through a non-stepping diffusive state when detaching from the microtubule. ....30
- Figure 16. Simulated position vs. time for mechanical activation model with a single kinesin motor pulling against six dynein motors. In the mechanical activation model, motor attachment rates depend on the cargo velocity in the opposite direction. ....32
- Figure 17. Simulated position vs. time for mechanical activation model including pauses with a single kinesin motor pulling against six dynein motors. Motors were allowed to reattach even if all motors became detached. This resulted in the appearance of clear pauses. ...33
- Figure 18. Simulated position vs. time for steric disinhibition model with a single kinesin motor pulling against six dynein motors. In the steric disinhibition model, motor attachment rates depend on the number of oppositely-directed motors bound to the cargo. ....34
- Figure 19. Effect of introducing dynein motors on cargo velocity in tug-of-war simulation. The introduction of additional dynein motors to the tug-of-war model results in a decreased average cargo velocity in the plus direction. ....36
- Figure 20. Effect of introducing dynein motors on cargo velocity in mechanical activation simulation. The introduction of a small number of dynein motors to the mechanical activation model results in an increased cargo velocity in the plus direction. ....37
- Figure 21. Average velocity with one kinesin bound to cargo vs. number of dynein motors bound to cargo using the mechanical activation model. The introduction of a small number of dynein motors results in an increase in anterograde velocity, demonstrating the paradox of codependence. ....38
- Figure 22. Effect of introducing kinesin-2 motors on cargo velocity in tug-of-war simulation. The introduction of additional kinesin-2 motors to the tug-of-war model results in a positive shift in velocity. ....39
- Figure 23. Effect of introducing kinesin-2 motors on cargo velocity in mechanical activation simulation. The introduction of additional kinesin-2 motors to the mechanical activation model results in a positive shift in velocity, contrasting with the results seen for kinesin-1. 40

## Chapter 1: Introduction

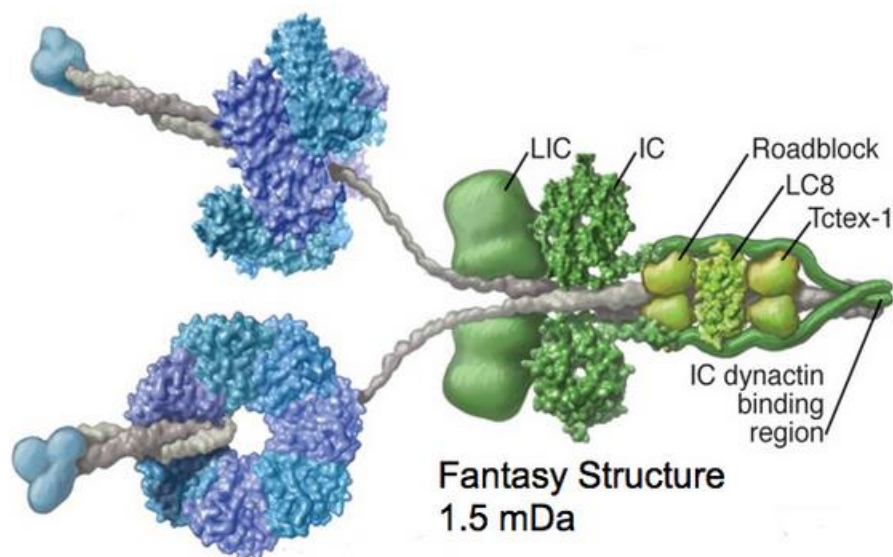
A motor protein is a protein that is capable of converting the chemical energy stored in ATP to mechanical work via a conformational change known as a “powerstroke.” A well-known example is the motor protein myosin, which binds to actin and is responsible for the contraction of muscles. The motor proteins discussed in this paper are microtubule-associated motor proteins. Microtubules are long, fibrous proteins formed by the polymerization of tubulin dimers, which consist of  $\alpha$  and  $\beta$  tubulin. The dimeric structure of microtubules results in a distinct polarity that influences the direction of transport by motor proteins. The opposite ends of a microtubule are referred to as the “plus end” and “minus end.”

In animals, microtubule-associated motor proteins are divided into two superfamilies: kinesin and dynein. While there are many proteins in the kinesin superfamily with differing structures, this project will be based on the structure of kinesin-1, which is the most studied and well-characterized member of the kinesin family. This protein contains two head or motor domains, which are capable of binding to microtubules and contain the protein's ATP hydrolysis sites. The head domains are connected via two neck linkers to a long, coiled coil stalk domain. At the other end of the stalk is the tail domain, which is capable of binding to a variety of intracellular cargos, including vesicles and mitochondria.<sup>1</sup> Figure 1 shows a simplified structure for a kinesin protein.



**Figure 1. Simplified kinesin structure.** Figure adapted from Asbury, et al.<sup>1</sup>

Members of the dynein superfamily of motor proteins fall into two main subcategories: axonemal dynein and cytoplasmic dynein. Axonemal dynein motor proteins work together in a coordinated fashion and are responsible for the back-and-forth movement of cilia and flagella. This project is focused on cytoplasmic dynein, which, along with kinesin, is responsible for the transport of intracellular cargos. Throughout the rest of this thesis, the term “dynein” will be used to refer specifically to cytoplasmic dynein. Compared to kinesin, dynein is a larger, more globular molecule. Like kinesin, dynein has two globular head domains, which contain the protein’s ATP hydrolysis sites. However, unlike kinesin’s head domains, dynein’s head domains do not directly bind to microtubules. Instead, each head binds to the microtubule via a coiled coil stalk domain. The head domains can bind intracellular cargos via two projections known as the tail domains.<sup>2</sup> Figure 2 shows a computer-generated representation of dynein’s structure.



**Figure 2. Computer-generated representation of dynein's structure. Figure adapted from Vale.<sup>3</sup>**

Intracellular transport of vesicles, organelles, and other cargo is accomplished via the movement of motor proteins along microtubules. Motor proteins are capable of binding to various cargos and “walking” toward either end of a microtubule. This walking is achieved by a hand-over-hand mechanism, in which the rear head of the protein detaches from the microtubule, swings forward via a powerstroke, and rebinds to the microtubule.<sup>1</sup>

With few exceptions, all kinesins are plus-end-directed, i.e. they move exclusively from the minus end of a microtubule toward the plus end. Conversely, all members of the dynein superfamily are minus-end-directed. In neurons, the result of this is that kinesin motors transport cargo toward the periphery of the neuron, while dynein motors transport cargo in the opposite direction, toward the cell body.

Until now, the majority of experiments dealing with microtubule-associated motor proteins have been relatively simple *in vitro* studies. While these experiments have certainly provided useful information, they are not sufficient for understanding the mechanisms underlying bidirectional transport by opposing

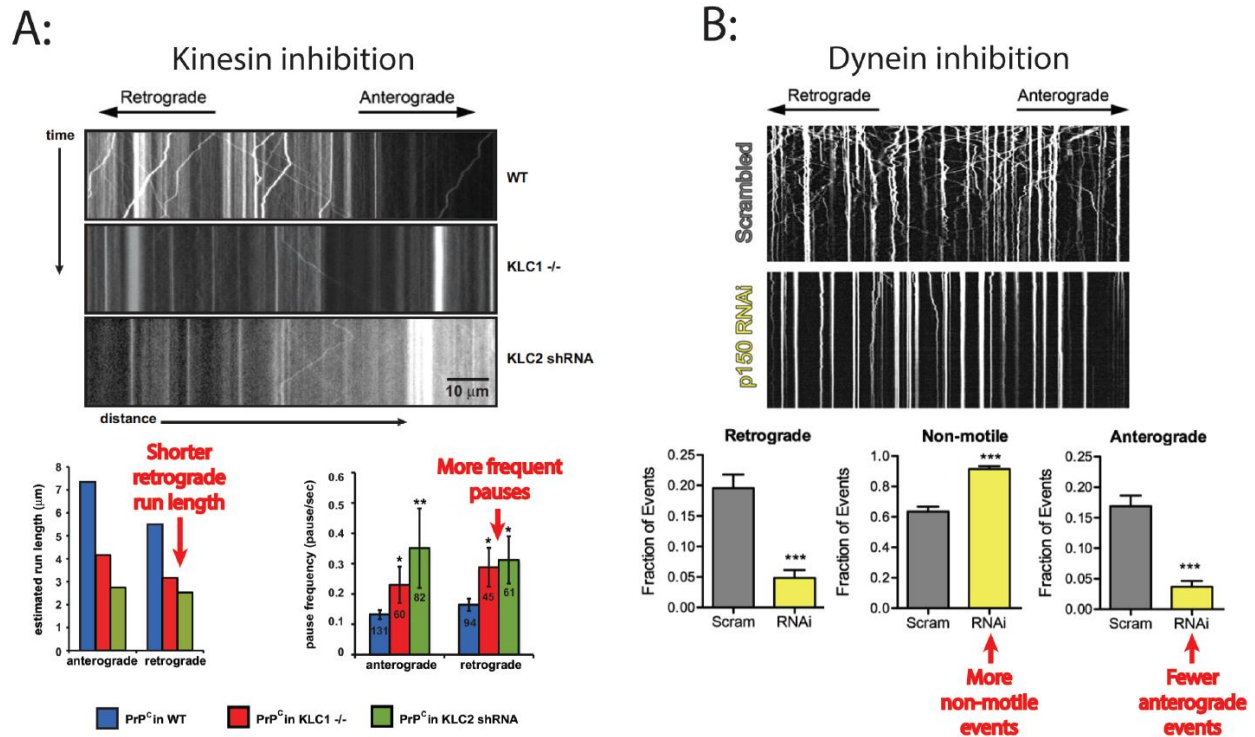


motor proteins. An understanding of these mechanisms is important because bidirectional transport in neurons is a crucial mechanism for the growth and maintenance of neurons. In addition, defects in intracellular transport in neurons are associated with a number of neurodegenerative diseases in humans. One example is Alzheimer's disease, which is characterized by tangles of the microtubule binding protein tau, which inhibit axonal transport. Other diseases that are believed to involve defects in axonal transport include Amyotrophic Lateral Sclerosis, Huntington's disease, and Parkinson's disease. Gaining a better understanding of the underlying mechanisms could eventually lead to the development of treatments for these diseases.<sup>4</sup>

Intracellular cargos are generally not transported by a single motor protein or even by members of a single motor protein superfamily. Instead, cargos are often bound to multiple kinesin and dynein motors. The movement of cargo transported by these motors is characterized by periods of smooth movement, rapid switches in direction, and frequent pauses. Currently, the leading model to explain these observations is a direct mechanical competition between kinesin and dynein, known as the "tug-of-war model."<sup>4</sup>

The use of the term "tug-of-war" to describe the mechanical competition between oppositely-directed motor proteins was first coined in 1998 by Welte, et al.<sup>5</sup> A direct implication of this mechanical competition is that the inhibition of one motor should theoretically result in enhanced motility in the opposite direction. However, a number of studies have shown that inhibiting either motor species results in a reduction in both anterograde and retrograde transport. For example, a study by Martin, et al. showed that knocking out kinesin and mutating dynein each resulted in randomly distributed axonal swellings caused by the accumulation of intracellular cargos.<sup>6</sup> If the tug-of-war model were accurate, one would expect these cargos to accumulate at either the cell body or the periphery of the axon, rather than at randomly distributed sites throughout the axon. Figure 3, adapted from Hancock<sup>4</sup>, shows additional

results for inhibiting either kinesin or dynein. Once again, these results show that inhibiting either motor species results in impaired motility in both directions.



**Figure 3. Experimental data displaying the paradox of codependence. A: Inhibiting kinesin motors either by knocking out kinesin light chain 1 (KLC1<sup>-/-</sup>) or by knocking down light chain 2 (KLC2 shRNA) results in shorter run lengths and more frequent pauses during retrograde transport. B: Inhibiting dynein in mouse neurons results in more time spent in the paused state and less time spent in both retrograde and anterograde transport.**

A computational model has been developed by Müller, Lipowsky, and colleagues that provides strong support for the tug-of-war model. This model utilizes parameters obtained from single-molecule experiments to estimate the velocity, run length, stall force, and detachment rates for kinesin and dynein motor proteins.<sup>7-9</sup> Support for this model has been shown in a study carried out by Hendricks, et al., in which cargo trajectories predicted by the Müller model were shown to match closely with experimentally observed bidirectional transport.<sup>10</sup> The authors of this study were able to estimate the number of kinesin and dynein motors bound to vesicles moving along microtubules *in vitro*. When the same numbers of kinesin and dynein motors were input into the Müller model, similar bidirectional transport was observed.

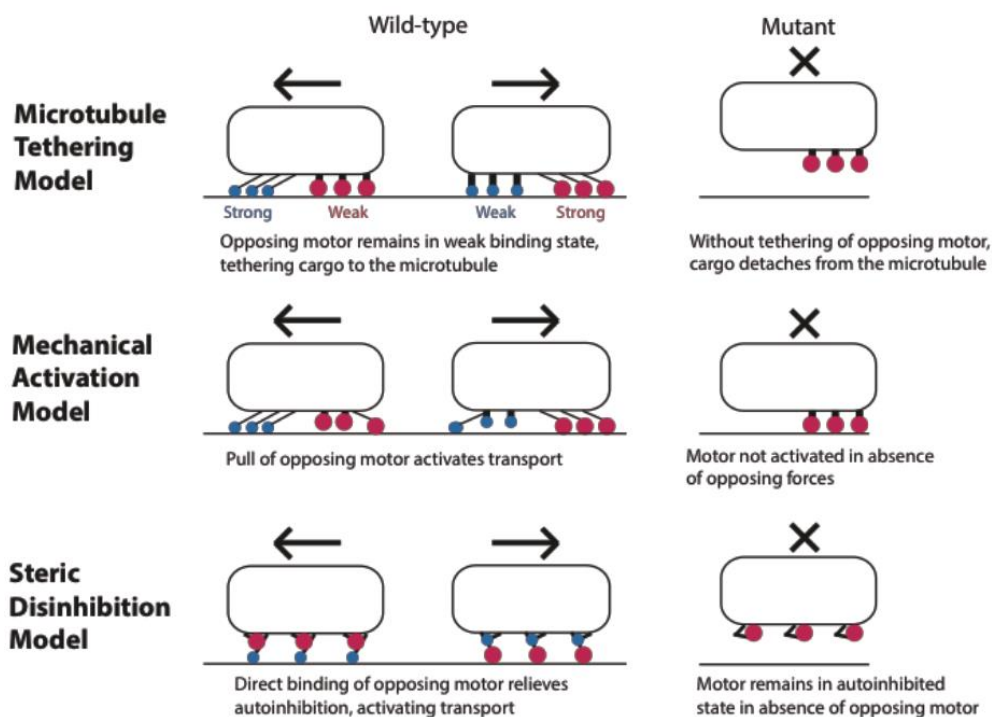
This model was also capable of simulating dynein inhibition studies by reducing the number of dynein motors bound to a given cargo. However, the model did not show the reduced anterograde transport observed in previous *in vitro* experiments.<sup>10</sup>

The Müller model fails to address several properties of real motor proteins, such as the stochastic nature of their stepping and the elastic tether connecting motors to their cargo. While the Müller model assumes that an equal load is applied to all motors of the same type, the stochastic nature of real motor proteins result in an unequal sharing of load between motors as some motors of the same type will move farther from the cargo than others. In an attempt to address the shortcomings of the Müller model, Kunwar, et al. developed a stochastic stepping model that accounted for the elastic tethers binding motors to their cargo. However, the authors concluded that this model was still unable to explain experimental results through a simple tug-of-war model.<sup>11</sup>

While these results show that the tug-of-war model cannot adequately explain the codependence between oppositely-directed motors, the question still remains as to whether this codependence relies on the presence of specific motor species or if it is simply the result of the forces exerted by these motors. A study conducted by Ally, et al. found that, in *Drosophila* S2 cells in which cargo transport had been arrested by kinesin-1 knockdown, motility could be restored by active kinesin-3 motors but not by inactive kinesin-1 motors. Likewise, when dynein was knocked down, motility could be restored using a minus-end-directed kinesin known as *ncd*.<sup>12</sup> These results show that, in this system, the codependence of oppositely-directed motors is due to the forces exerted by these motors and not due to the binding of specific motor species.

Clearly, the direct mechanical competition between oppositely-directed motor proteins proposed by the tug-of-war model is not sufficient to explain the codependence between antagonistic motors that is evident from these results. In order to overcome this shortcoming of the tug-of-war model, three

additional models have been proposed by Hancock to explain the observed behavior of bidirectional transport. It is important to note that these models are purely hypothetical and have been proposed as a means to direct the development of future experiments and computational models.



**Figure 4. Three models proposed to explain the codependence between oppositely-directed motor proteins. Figure adapted from Hancock.<sup>4</sup>**

The first mechanism proposed by Hancock is the “microtubule tethering model.” This model proposes that, in addition to an attached, processive state and a detached state, motors may exist in a diffusive state in which they are weakly associated with the microtubule. In this state, the motors would provide minimal resistance to processive motion in the opposite direction but would allow the cargo to remain associated with the microtubule in the case of the other motors detaching, resulting in longer run lengths.<sup>3</sup> Okada, et al. were the first to show that kinesin could enter a weakly-bound, diffusive state using kinesin-3. It was found that this weak binding was facilitated by the positively-charged “K-loop” of kinesin and the negatively-charged C-terminal tail of tubulin, the monomer that polymerizes to form microtubules.<sup>13</sup>

Since then, it has been shown that kinesin-5 and kinesin-1 are capable of entering a similar diffusive state.<sup>14,15</sup> In addition, Culver-Hanlon, et al. have shown that cytoplasmic dynein is also capable of diffusing along microtubules.<sup>16</sup>

The second proposed model is known as the “mechanical activation model.” This model states that motors tend to exist in an inhibited state unless activated by a hindering load, such as a force exerted by an oppositely-directed motor protein. This inhibition could be due to a number of mechanisms, including tight binding to the microtubules or an inability to properly interact with the microtubule in the absence of a hindering load.<sup>4</sup> At this time, evidence in favor of this model is somewhat lacking. Studies have shown that dynein tends to remain bound for a longer period of with for larger superstall forces<sup>11</sup> and that kinesin detaches more readily under assisting loads than hindering loads.<sup>17</sup> These results show that hindering loads can enhance attachment, but they do not necessarily imply any sort of mechanical activation. The aforementioned results from Ally, et al. that show that the binding of any functional, oppositely-directed motor can restore motility to a cargo that only has one type of motor bound are consistent with a mechanical activation mechanism.<sup>12</sup> However, these results could also be consistent with the microtubule tethering model.

The third and final hypothetical mechanism set forth by Hancock is the “steric disinhibition model.” This model proposes that kinesin and dynein tend to exist in an autoinhibited state until this inhibition is relieved by the binding of an oppositely-directed motor or a regulatory protein.<sup>3</sup> Studies have shown that kinesin motors can enter an autoinhibited state. This autoinhibition occurs due to the tail domain of kinesin folding back and interfering with the motor domains.<sup>18,19</sup> It has been hypothesized that this autoinhibition is relieved upon binding to a cargo or binding by regulatory proteins, but the steric disinhibition model holds that activation will not occur unless dynein is present. There is no defined mechanism by which dynein undergoes autoinhibition, but regulatory proteins have been shown to exist which are associated with the activation of dynein.<sup>20</sup> The steric disinhibition model is supported by

experiments that have shown that kinesin-1 is capable of directly binding to dynein<sup>21</sup> and that dynactin, a protein that links dynein to cargos, also links kinesin-2 to cargos.<sup>22</sup> In addition, an adaptor protein has been discovered that directly binds kinesin-1 and dynein and serves as a link to mitochondria.<sup>23</sup>

The goal of this project was to develop MATLAB scripts simulating the tug-of-war model, as well as the three proposed models. These simulations were created using parameters pulled from literature along with the Gillespie algorithm to determine rates for a number of events associated with the bidirectional stepping of motor proteins. While there are many results that have yet to be obtained from these simulations, the mechanical activation model has already been shown to be a viable candidate for explaining the paradox of codependence.

## Chapter 2: Methods

The objective of this project was to build a computational simulation capable of simulating the transport of a cargo bound to various numbers of kinesin-1 and cytoplasmic dynein motors. I began by developing a simulation according to the mechanism described by the tug-of-war model before extending the simulation to replicate the other three hypothetical models. Programming for these simulations was carried out in MATLAB.

These simulations make use of the first-reaction formulation of the Gillespie algorithm. This algorithm is a widely used method of simulating the progression of stochastic, highly-coupled chemical reactions with time. The Gillespie algorithm states that the time for a given event to occur is determined as

$$\tau_i = (1/k_i)\ln(1/r_i)$$

where  $\tau_i$  is the dwell time required before a given event  $i$  will occur,  $k_i$  is the rate constant for event  $i$ , and  $r_i$  is a pseudo-random number between 0 and 1.<sup>24</sup> In the basic tug-of-war model, the possible events include forward stepping of an attached motor, backward stepping of an attached motor, detachment of an attached motor, and attachment of a detached motor. After dwell times are calculated for all possible events, the event with the lowest dwell time is chosen as the next event that will occur. This process is then repeated until a set time limit is reached or all motors have detached from the microtubule. One might be tempted to select the second lowest time interval as the “second next” event to occur, but this would be invalid as it disregards the changes in rate constants cause by the previous event and precludes the possibility of the same event occurring twice sequentially.<sup>24</sup> A lower limit of 100  $\mu$ s was set on the dwell time in order to avoid unrealistically high instantaneous velocities.

### 2.1: Kinesin Parameters

The rate constants for various events were approximated based on values obtained from literature, typically as a function of force. For detached motors, the forward stepping rate, backward stepping rate, and detachment rate are set to 0 as none of these events can occur while the motor is detached. The attachment rate is approximated as  $5 \text{ s}^{-1}$ , which is the same value used in the Müller model.<sup>8</sup> If the motor is attached, the attachment rate is set to zero. The forward stepping rate for kinesin is determined as a function of force. The forward stepping rate for kinesin was determined from a force-velocity curve published by Carter and Cross. The forward stepping rate for the unloaded case and for assisting loads is held constant at  $100 \text{ s}^{-1}$ . For loads between 2.5 pN and 7.76 pN, the forward stepping rate is calculated as

$$k_f = 277.78(e^{-0.57F})$$

where  $k_f$  is the forward stepping rate in  $\text{s}^{-1}$  and  $F$  is the load force in pN. Note: for this and all subsequent equations, a hindering load is defined as a positive force. This equation was obtained by taking the inverse of Carter and Cross's equation for the forward dwell time of kinesin, which was determined by fitting experimental data.<sup>25</sup> For hindering loads below 2.5 pN, the forward stepping rate is calculated as

$$k_f = 100(e^{-0.162F})$$

This equation was obtained by fitting an exponential between the value of the forward rate in the unloaded case and the value at 2.5 pN. For hindering loads above 7.76 pN, the forward stepping rate was approximated as  $3.33 \text{ s}^{-1}$ . Based on the ratio of forward to backward steps described by Carter and Cross, the backward stepping rate for kinesin under assisting loads and hindering loads below 7.76 pN is calculated as

$$k_b = k_f/[802(e^{-0.95F})]$$



where  $k_b$  is the backward stepping rate in  $s^{-1}$ . From Andreasson<sup>17</sup>, kinesin's detachment rate under assisting loads is calculated as

$$k_d = 0.79 - 1.56F$$

where  $k_d$  is the detachment rate in  $s^{-1}$ . For hindering loads, the detachment rate is calculated as

$$k_d = 0.79(e^{F/6.1})$$

Later models included the introduction of kinesin-2 parameters. These parameters were estimate by scaling kinesin-1's forward and backward stepping rates to a maximum velocity half that of kinesin-1. Kinesin-2 also has a much more force-dependent detachment rate then kinesin-1. From Arpag, et al.<sup>26</sup>, the force-dependent detachment rate for kinesin-2 under a hindering load was determined to be

$$k_{off} = 8.37e^{-0.07F} - e^{0.43F}$$

and the detachment rate under an assisting load was determined to be

$$k_{off} = 15e^{F/2.0}$$

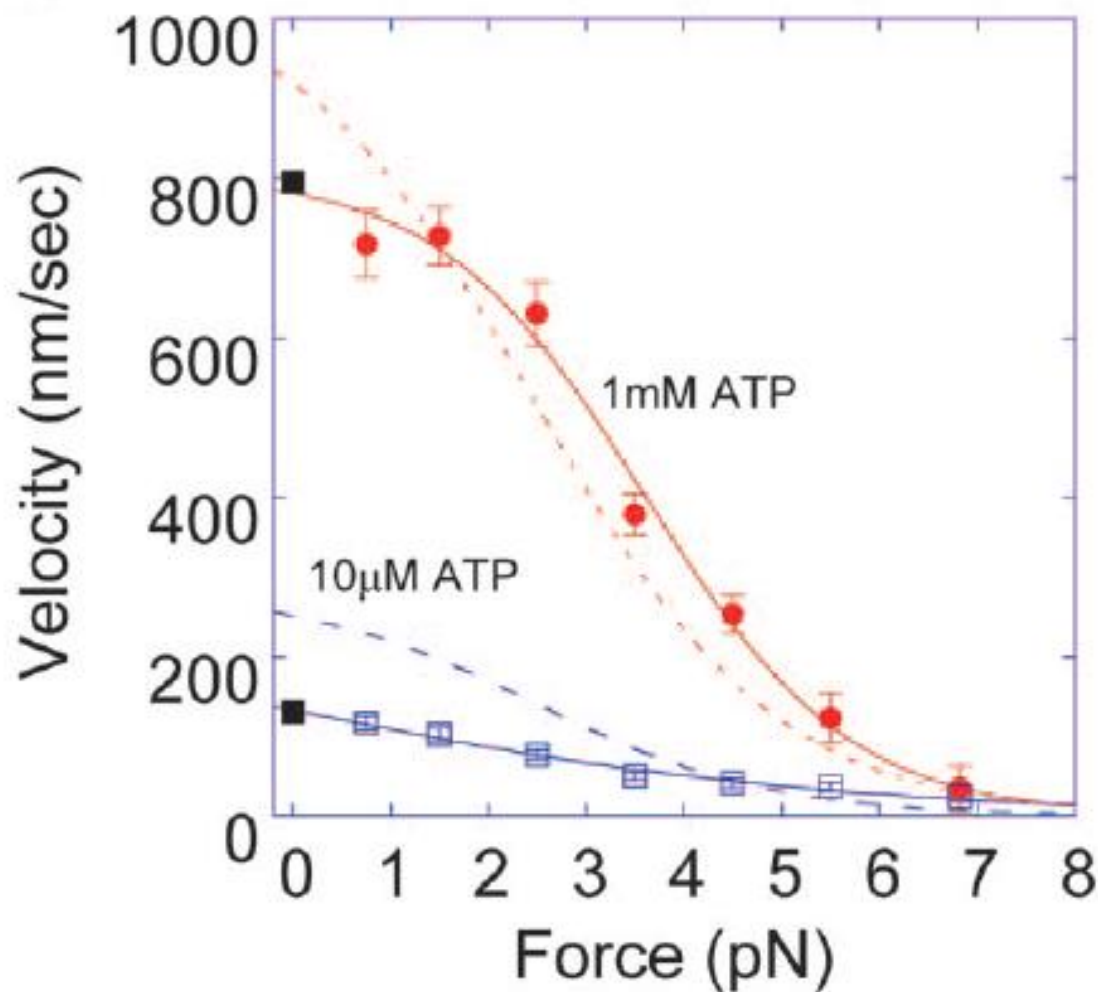
## 2.2: Dynein Parameters

It was somewhat difficult to define the rate constants for dynein as it is not as well characterized as kinesin, and the available literature contains a wide range of values. Due to the limited data available, it was often necessary to use results from experiments using yeast dynein. Compared to mammalian

dynein, yeast dynein exhibits a significantly lower velocity but a significantly higher stall force. As such, it was necessary to scale these results to values obtained from other studies for the velocity and stall force of mammalian dynein.

For detached dynein motors, the attachment rate is defined as  $5 \text{ s}^{-1}$ , the same as the attachment rate for kinesin. The only available source for cytoplasmic dynein's force-velocity relationship is Toba, et al.<sup>27</sup>

This force-velocity curve is shown in Figure 5.



**Figure 5. Force-velocity relationship for mammalian cytoplasmic dynein in 10  $\mu\text{M}$  and 1 mM ATP. For the purposes of this project, the curve at 1 mM ATP was used. Figure adapted from Toba, et al.<sup>27</sup>**

However, this study reports an unusually high unloaded velocity and stall force. In order to correct for this, the force-velocity relationship was scaled to an unloaded velocity of approximately 213 nm/s and a stall force of approximately 1 pN, consistent with experimental values reported by Mallik, et al.<sup>28</sup> The shape of the force-velocity curve shown by Toba, et al. appears to suggest a sigmoidal relationship. However, the data was instead fit by a constant value under assisting loads and low hindering loads and a decaying exponential under high hindering loads as this was expected to more accurately represent the underlying mechanisms. Also, for the sake of simplicity, dynein was assumed to exclusively take 8 nm steps, as opposed to the variable step size observed in experiments.<sup>28</sup> Thus, for assisting loads and hindering loads below 0.3 pN, dynein's forward stepping rate is held constant at approximately 34.1 s<sup>-1</sup>. For hindering loads exceeding 0.3 pN, the forward stepping rate is calculated as

$$k_f = 61.79(e^{-1.98F})$$

Reck-Peterson, et al. report that, for the unloaded case, approximately 20% of yeast dynein's steps are backward steps.<sup>29</sup> No data were available on the change in dynein's backward stepping rate or for mammalian dynein's backward stepping rate in general, so dynein's backward stepping rate was simply held constant at its unloaded rate of 8.52 s<sup>-1</sup>. Dynein's detachment rate was obtained from Kunwar, et al.<sup>11</sup> For hindering loads below 1.7 pN, the detachment rate is calculated as

$$k_d = e^{0.8155F}$$

For hindering loads exceeding 1.7 pN, the detachment rate is calculated as

$$k_d = 1/[0.254(1 - e^{-F/1.97})]$$

Figure 6 shows a graph published by Kunwar, et al. showing the dwell time for dynein's detachment as a function of force. Note that dwell time is the inverse of the rate.

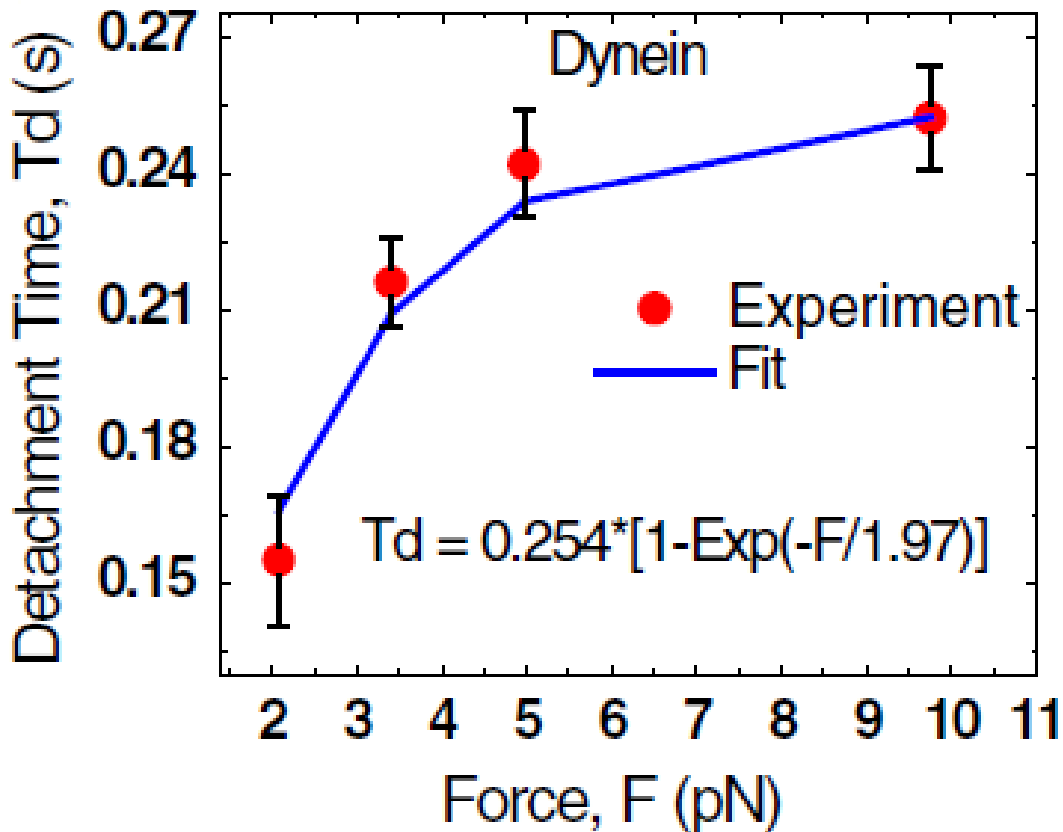


Figure 6. Dwell time for the detachment of dynein vs. force. Figure adapted from Kunwar, et al.<sup>11</sup>

There was no published data for the dynein detachment rate for assisting loads, so a simple exponential relationship was chosen, that has a somewhat stronger force dependence than the hindering direction

$$k_d = e^{-F/3}$$

A second model for dynein was developed in which dynein had a variable backward stepping rate. Recall that this rate was initially set to a constant value due to limited available data. At this point in the project, a source had been located containing the necessary data. Based on Gennerich, et al.'s study on the

bidirectional stepping of dynein<sup>30</sup>, the force-dependent ratio between the forward stepping rate and the backward stepping rate for yeast dynein was determined to be

$$SR = -0.1707F + 2.186$$

where SR is the ratio between dynein's forward stepping rate and its backward stepping rate and F is the force applied to the dynein motor in the plus direction. After scaling this equation to mammalian dynein's stall force of approximately 1 pN, the result was

$$SR = -1.2422F + 2.186$$

for  $0.13745 \text{ pN} < F < 1.3745 \text{ pN}$ . Outside the boundaries of this equation, the ratio is held constant. In order to simplify the calculations for the actual stepping rate, dynein's force-velocity curve was simplified to the following linear equation:

$$NSR = 31.606 - 33.095F$$

for  $0.3 \text{ pN} < F < 1 \text{ pN}$ , where NSR is the net stepping rate for dynein and F is the force applied to the dynein motor in the plus direction. Like the stepping ratio, the net stepping rate is held constant outside the boundaries of this equation. From these two equations, dynein's forward stepping rate is defined as

$$k_f = NSR / (1 - \frac{1}{SR})$$

From this point on, this model will be referred to as “dynein model 2,” while the model with a fixed backward stepping rate will be referred to as “dynein model 1.”

and the backward stepping rate is defined as

$$k_b = k_f/SR$$

### 2.3: Tug-of-War Simulation

I began building the simulation by creating a simple simulation of a single kinesin bound to a cargo to which a variable load force could be applied. The tether binding the kinesin to the cargo was modeled as an ideal Hookean spring with a spring constant of 0.3 pN/nm.<sup>31</sup> I then added complexity to the model by simulating two kinesin motors bound to a single cargo, followed by a simulation in which the cargo is bound by two normal kinesins and two kinesins that have had their polarity reversed to favor retrograde motion. This was initially used as a stepping stone for creating a simulation with dynein, but it ultimately proved useful for testing the sensitivity of certain parameters, such as the elasticity of the cargo-motor tether.

The next step was to input the parameters for dynein to create a simulation of two kinesin motors pulling against two dynein motors. The spring constant for dynein's tether was set as 0.065.<sup>2</sup> From there, the model was extended to be capable of handling any number of kinesin and dynein motors.

For the sake of simplicity, the tethers binding motors to the cargo were modeled as ideal Hookean springs. However, a Hookean spring model provides a poor approximation of the force-extension relationship observed in flexible polymers. A more accurate model is the worm-like-chain model. This model states that a flexible polymer under no load will exist in a randomly-coiled configuration due to thermal fluctuations. As a load is applied and the polymer extends, the number of available configurations is reduced, resulting in an entropic force that resists extension.<sup>32</sup>

The result of this is that the polymer provides very little resistance to extension until the polymer is stretched to a length approaching its contour length, which is defined as the maximum length to which the polymer can be stretched. At this point, the resistive force rises drastically, approaching an asymptote at the contour length. The relationship between force and extension is described mathematically as

$$F = \frac{k_B T}{L_p} \left[ \frac{1}{4} \left( 1 - \frac{x}{L_c} \right)^{-2} - \frac{1}{4} + \frac{x}{L_c} \right]$$

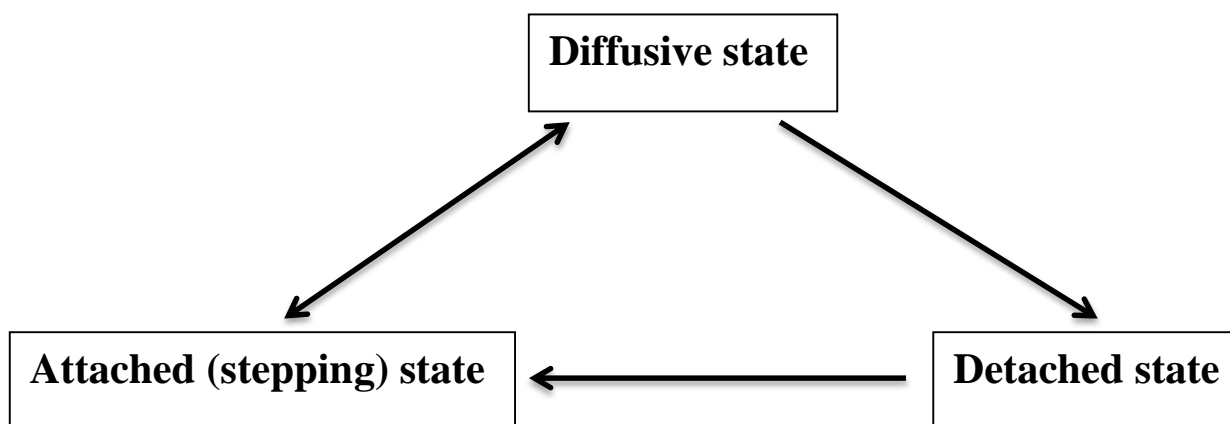
where  $F$  is the force resisting extension,  $k_B$  is Boltzmann's constant,  $T$  is the absolute temperature in K,  $L_p$  is the persistence length,  $L_c$  is the contour length, and  $x$  is the end-to-end distance of the polymer. The persistence length is defined as the minimum distance that must exist between two segments of the polymer for the orientation of the two segments to be completely uncorrelated.<sup>32</sup> For kinesin, a contour length of 84 nm was used, and for dynein, a contour length of 48 nm was used. Both motors used a persistence length of 0.7 nm.

While the worm-like chain model is believed to be more accurate, due to debugging difficulties and the inherent complexity of working with the worm-like chain model, the final comparison between the tug-of-war and mechanical activation models was carried out with the motor tethers approximated as Hookean springs. This comparison will be discussed in greater detail in subsequent sections.

#### *2.4: Microtubule Tethering Simulation*

After completing the basic tug-of-war model, the next step on the simulation was the addition of a diffusive state in order to simulate the mechanism proposed by the microtubule tethering model. In the diffusive state, the motor remains associated with the microtubule but will not take any steps forward or backward. In the initial version of this simulation, motors were only able to detach from the microtubule by going through the diffusive state. That is, the detachment rate for attached motors was set to 0.

The rate for motors to enter the diffusive state from the attached state was initially calculated using the same equations used for the detachment rate in the tug-of-war model. From the diffusive state, the motor can either detach completely or reattach to the microtubule. Figure 7 shows a visual representation of this model. In an effort to create a distinct paused state, it was decided that motors should be significantly more likely to reattach than detach from the diffusive state. Therefore, the attachment rate from the diffusive state was initially set to  $10 \text{ s}^{-1}$ , and the detachment rate was initially set to  $2 \text{ s}^{-1}$ .



**Figure 7. Visual Representation of Microtubule Tethering Model**

atch

model behavior to experimental behavior. From the detached state, motors were only able to reattach directly to the microtubule without passing through the diffusive state. The attachment rate was initially set to  $5 \text{ s}^{-1}$ .

### *2.5: Mechanical Activation Simulation*

The next model to be introduced was the mechanical activation model. This model is based on the hypothesis that motor proteins tend to exist in an inactive state unless activated by a hindering load. This was simulated by making the attachment rate for a particular motor protein dependent on the cargo velocity in the opposite direction.

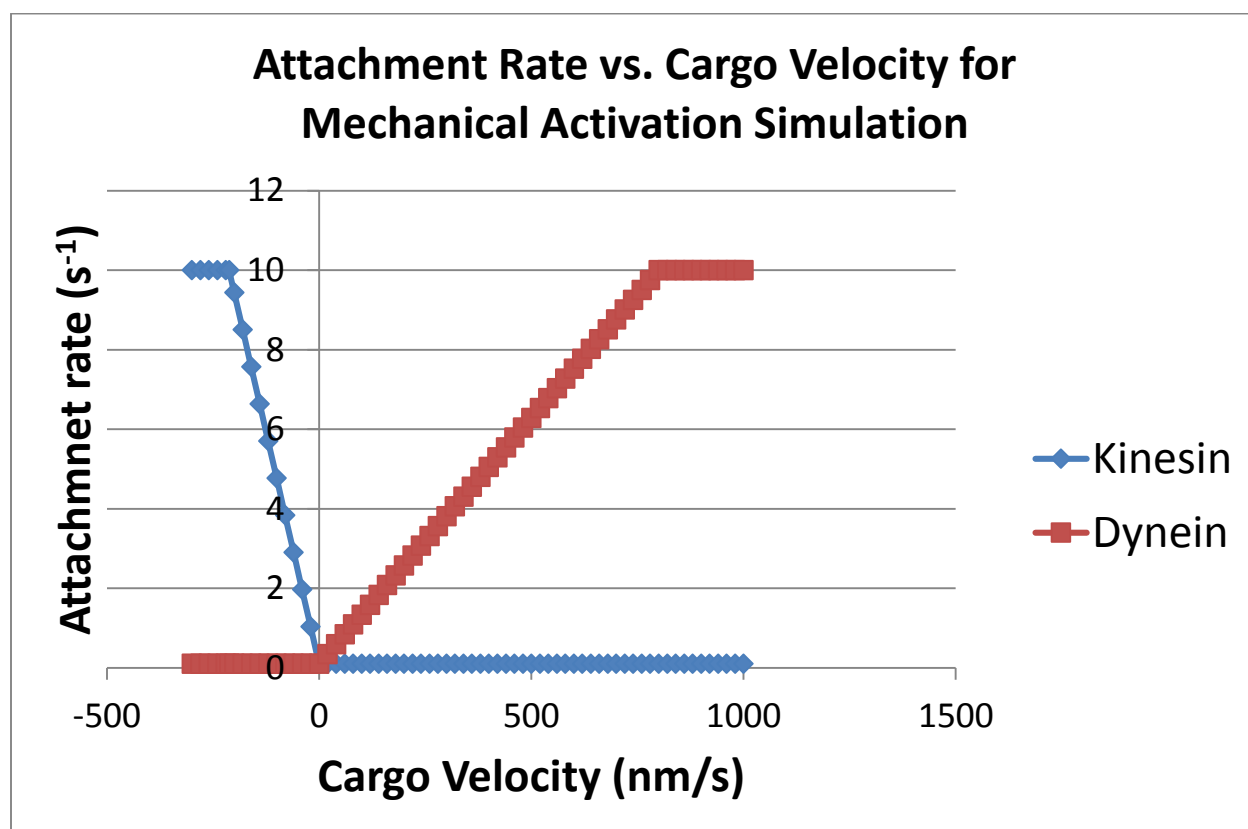


Because the simulation only recalculates the cargo position after an event has occurred and assumes that the cargo reaches equilibrium instantaneously, it is impossible to determine an instantaneous cargo velocity. The closest approximation of an instantaneous velocity would be determined by dividing the change in position between two subsequent time points by the change in time. However, the stochastic nature of the model can result in very short time steps, leading to very high velocities that cause immediate reattachment of any detached motor proteins that are directed opposite to the direction of cargo movement. As such, the attachment rate is determined using a velocity averaged over the last 100 time steps.

The attachment rate for kinesin has a minimum value when the cargo velocity is either 0 or positive, i.e. directed toward the plus end of the microtubule. The attachment rate reaches its maximum value when the cargo velocity is equal to or less than the approximate unloaded velocity for dynein. Note that “less than” is used because dynein’s velocity is negative. Dynein’s unloaded velocity is approximately -212 nm/s. Between these values, the attachment rate for kinesin increases linearly, resulting in the following equation:

$$k_{kon} = -((k_{kon\_max} - k_{kon\_min})/212) * vel + k_{kon\_min}$$

where  $k_{kon}$  is the attachment rate for kinesin,  $k_{kon\_max}$  is the maximum attachment rate for kinesin,  $k_{kon\_min}$  is the minimum attachment rate for kinesin, and  $vel$  is the cargo velocity averaged over the most recent 100 time steps. The negative sign is used to account for the fact that dynein has a negative velocity. A similar equation is used to calculate the attachment rate for dynein with minor changes to account for the fact that kinesin’s unloaded velocity is approximately 800 nm/s. For both kinesin and dynein motors, the minimum attachment rate was set to  $0.1 \text{ s}^{-1}$ , and the maximum attachment rate was set to  $10 \text{ s}^{-1}$ . Figure 8 shows how the attachment rate for each motor species varies with the cargo velocity.



**Figure 8. Attachment Rate vs. Cargo Velocity for Mechanical Activation Simulation.** Both motor species reach their minimum attachment rate at 0 velocity and reach their maximum at the average unloaded velocity of the opposing motor.

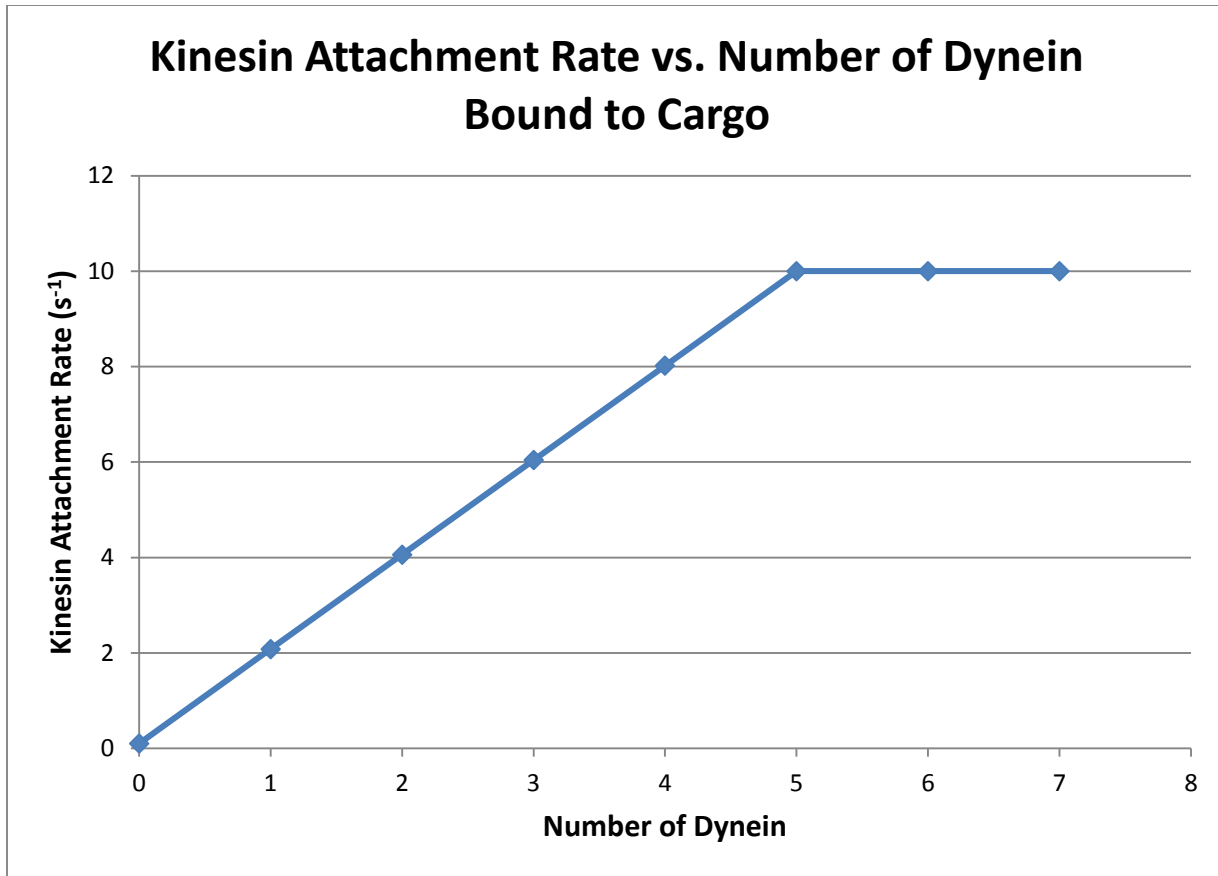
## 2.6: Steric Disinhibition Simulation

The final model that was introduced was the steric disinhibition model. This model states that motor proteins tend to exist in an autoinhibited state unless they are bound by either an oppositely-directed motor protein or some other associated protein. For the sake of simplicity, the simulation assumes that this binding is achieved directly by an oppositely-directed motor protein. If this project is continued beyond the culmination of my thesis, microtubule-associated proteins could possibly be introduced.

The steric disinhibition simulation functions by basing the attachment rate for kinesin and dynein on the number of oppositely-directed motors bound to the cargo. Note that while the motors frequently detach from the microtubule, this simulation assumes that the motors are not capable of detaching from the cargo. Therefore, for any given run of the simulation, the attachment rates will remain constant throughout the run. For kinesin motors, the attachment rate increases linearly from a minimum value when there are no dynein motors bound to the cargo to a maximum value when there are five dynein motors bound to the cargo, resulting in the following equation

$$k_{kon} = ((k_{kon\_max} - k_{kon\_min})/5) * dyns + k_{kon\_min}$$

where  $k_{kon}$  is the attachment rate for kinesin,  $k_{kon\_max}$  is the maximum attachment rate for kinesin,  $k_{kon\_min}$  is the minimum attachment rate for kinesin, and  $dyns$  is the number of dynein motors bound to the cargo. Figure 9 shows how kinesin's attachment rate is affected by an increasing number of dynein motors bound to the same cargo.



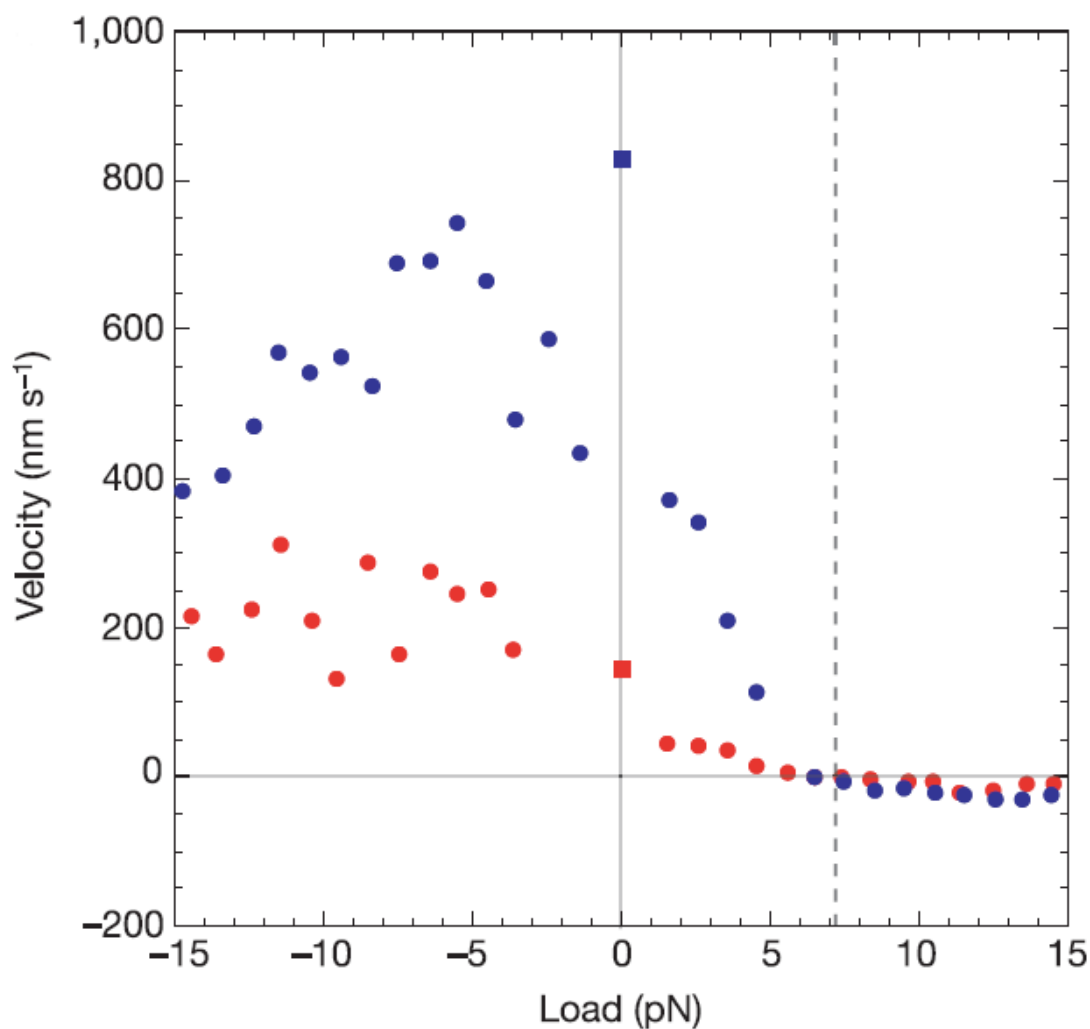
**Figure 9. Kinesin Attachment Rate vs. Number of Dynein Bound to Cargo for Steric Disinhibition Model.** Kinesin's attachment rate increases linearly with the number of dynein bound to the cargo before reaching a maximum when 5 dynein motors are bound.

Because a cargo will typically have a high ratio of bound dynein motors to bound kinesin motors<sup>10</sup>, dynein's attachment rate simply depends on whether there are any kinesin motors bound to the cargo. That is, dynein's attachment rate is either set to its minimum value when there are no kinesin motors bound to the cargo or its maximum value when there is any number of kinesin motors bound to the cargo. As a result, this model has no effect on the behavior of dynein during bidirectional transport. Like the mechanical activation model, the minimum attachment rate for both kinesin and dynein was set to  $0.1 s^{-1}$ , and the maximum attachment rate was set to  $10 s^{-1}$ .

## Chapter 3: Results/Discussion

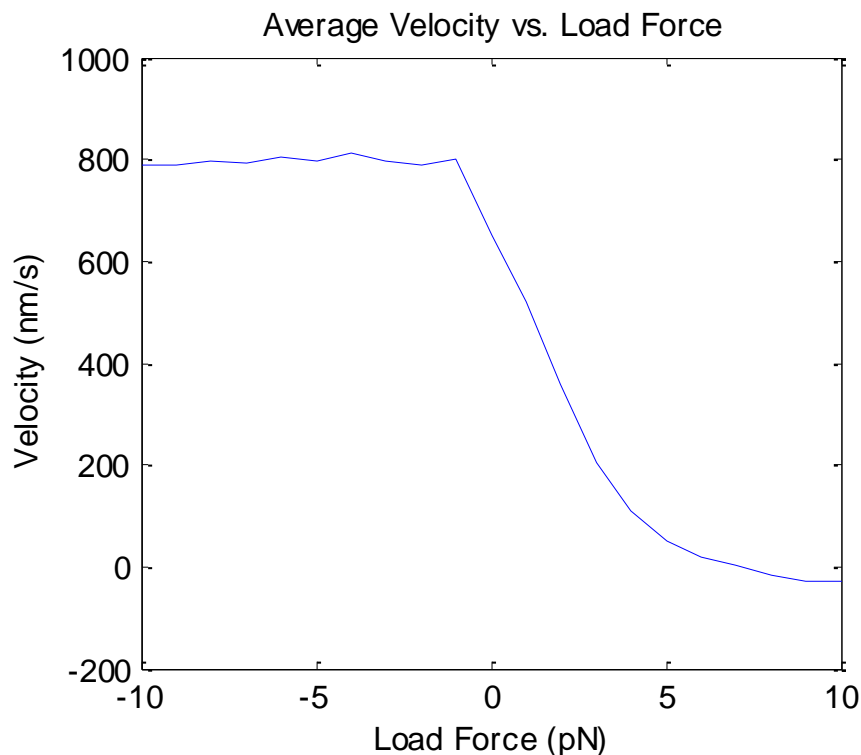
### 3.1: Initial testing

In order to confirm the validity of the underlying techniques used in the simulation, the single-kinesin simulation utilizing the worm-like-chain model was run using load forces ranging from -10 to 10 pN. Based on experimental results, such as those obtained by Carter Cross, we expect to see an average velocity in the vicinity of 800 nm/s under assisting loads and an exponentially decreasing velocity with increased hindering load.<sup>25</sup> These experimental results are shown in Figure 10.



**Figure 10. Kinesin-1 velocity vs. force for 10 μM ATP (in red) and 1 mM (in blue). The data for 1mM ATP matches closely with simulation results. Figure adapted from Carter and Cross.<sup>25</sup>**

Figure 11 shows the velocity as a function of load force averaged over 3000 runs of the simulation. Each run was allowed to continue for 10 seconds of simulation time or until the motor detached from the microtubule.

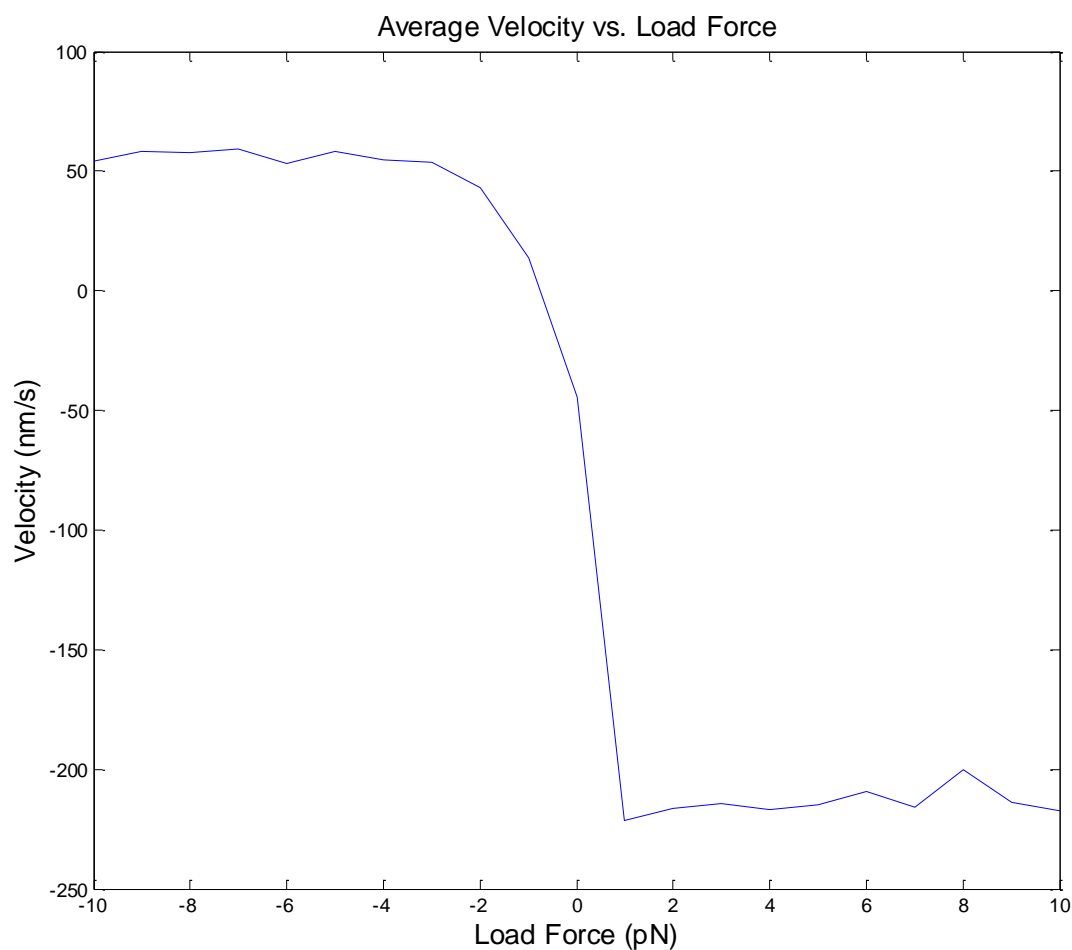


**Figure 11. Simulated force-velocity relationship for a single kinesin motor. The velocity at forces ranging from -10 to 10 pN was determined from the average of 3000 simulations. The graph reveals a velocity of approximately 800 nm/s for the unloaded case and for assisting loads.**

The simulated force-velocity curve shows a relatively constant velocity of approximately 800 nm/s for the unloaded case and for assisting loads. For hindering loads, the velocity appears to decrease roughly exponentially as load increases. This is precisely what we expect to see, so this result confirms the validity of the most basic aspects of the simulation.

After adding dynein parameters to the simulation, the same test was performed for dynein. Again, the simulation was run using load forces ranging from -10 to 10 pN, and the velocity for each force was averaged over 3000 runs, which were allowed to continue for 10 seconds of simulation or until the motor

detached from the microtubule. The results of this test performed on the dynein 1 model can be seen in Figure 12. Unfortunately, these results cannot be directly compared to an experimental force-velocity curve as experimental data could be found showing a clear force-velocity curve for mammalian dynein.

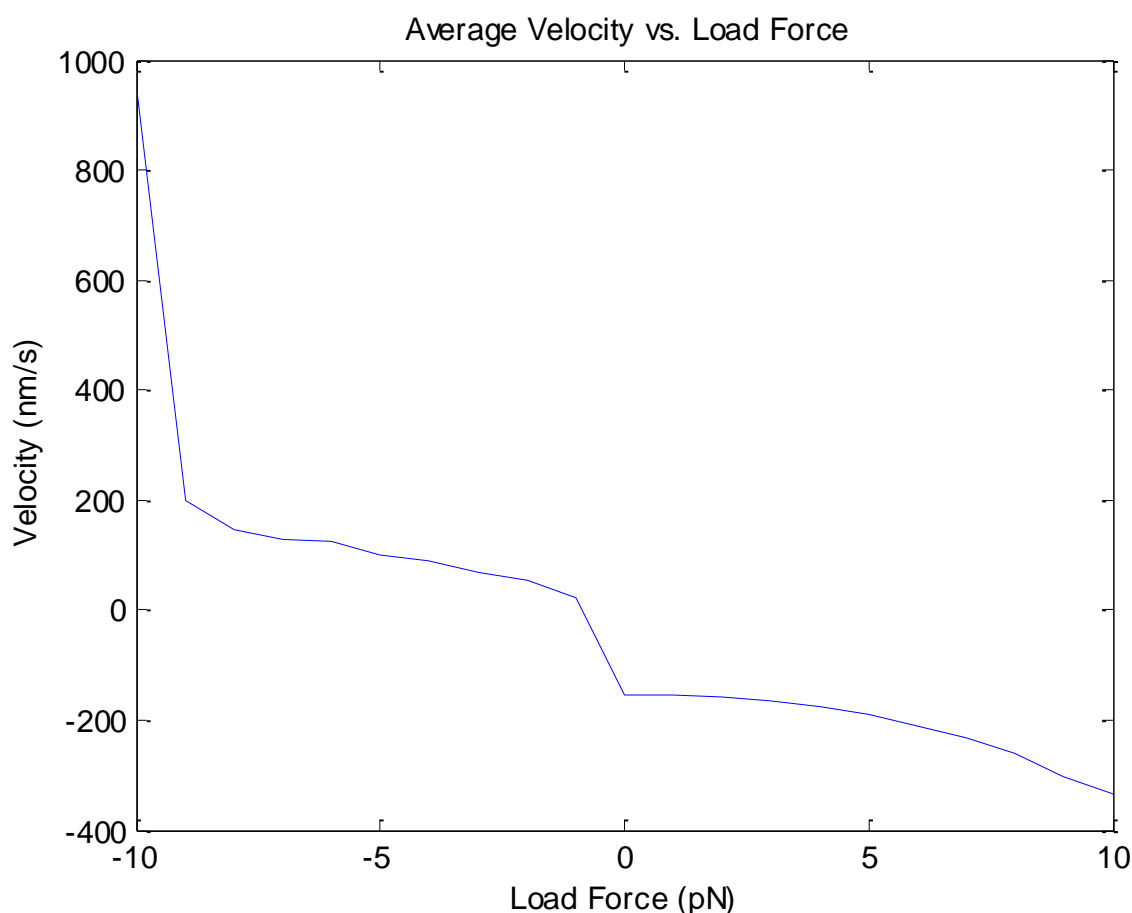


**Figure 12. Simulated force-velocity relationship for a single dynein motor using the dynein 1 model. The velocity at forces ranging from -10 to 10 pN was determined from the average of 3000 simulations. The graph reveals a velocity of approximately -200 nm/s for the unloaded case and for assisting loads.**

Recall that the sign conventions used in this simulation result in dynein having a negative forward velocity. Additionally, while a positive load force is a hindering load when applied to a kinesin motor, it is an assisting load when applied to a dynein motor. Keeping this in mind, Figure 12 shows that dynein

has a fairly constant maximum forward velocity slightly over 200 nm/s for an assisting load, which is what we expect to see based on the dynein parameters that were entered into the model. This velocity then decreases exponentially to a backward velocity of approximately 50 nm/s for large hindering loads. The maximum backward velocity for dynein is much greater than that of kinesin due to dynein's more frequent backward stepping.

Figure 13 shows the results of the same simulation performed for the dynein 2 model, which includes variable backward stepping. Due to time constraints, the velocities in Figure 1 were averaged over 1000 runs.



**Figure 13. Simulated force-velocity relationship for a single dynein motor using the dynein 1 model. This graph reveals a larger sensitivity to load force than was seen in the dynein 1 model.**



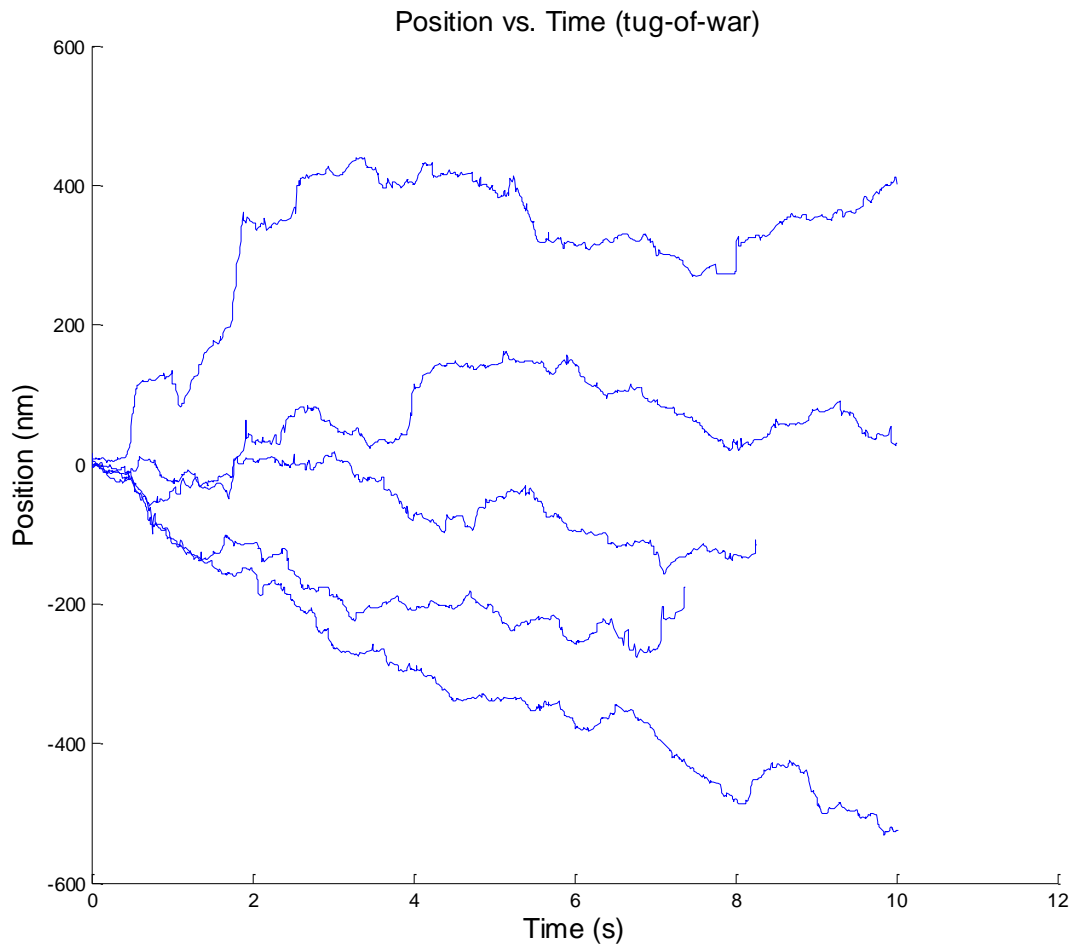
As Figure 13 shows, the dynein 2 model is much more sensitive to the load force. The average velocity decreases to a lower value than dynein 1 under hindering loads, and large assisting loads result in a large increase in the average velocity.

### *3.2: Comparison of Proposed Models*

After writing scripts for each of the proposed models, the first step in testing them was to carry out a simple qualitative comparison between the distance vs. time plots produced by each simulation. Note that these plots were produced using dynein model 1 and are included as an early proof of concept.

#### **3.2.1: Tug-of-War Model**

Simulations of a single kinesin pulling against multiple dyneins reveal that approximately six dynein motors are required to balance out a single kinesin motor. This is consistent with values reported by Hendricks, et al.<sup>10</sup> As an example, Figure 14 shows the position as a function of time for five runs with a single kinesin motor pulling against six dynein motors using the basic tug-of-war simulation. The simulation was allowed to continue for 10 seconds or until all motors came detached from the microtubule.



**Figure 14. Simulated position vs. time for tug-of-war model with a single kinesin motor pulling against six dynein motors. Each line represents a single run that was allowed to continue for 10 seconds or until all motors became detached from the microtubule.**

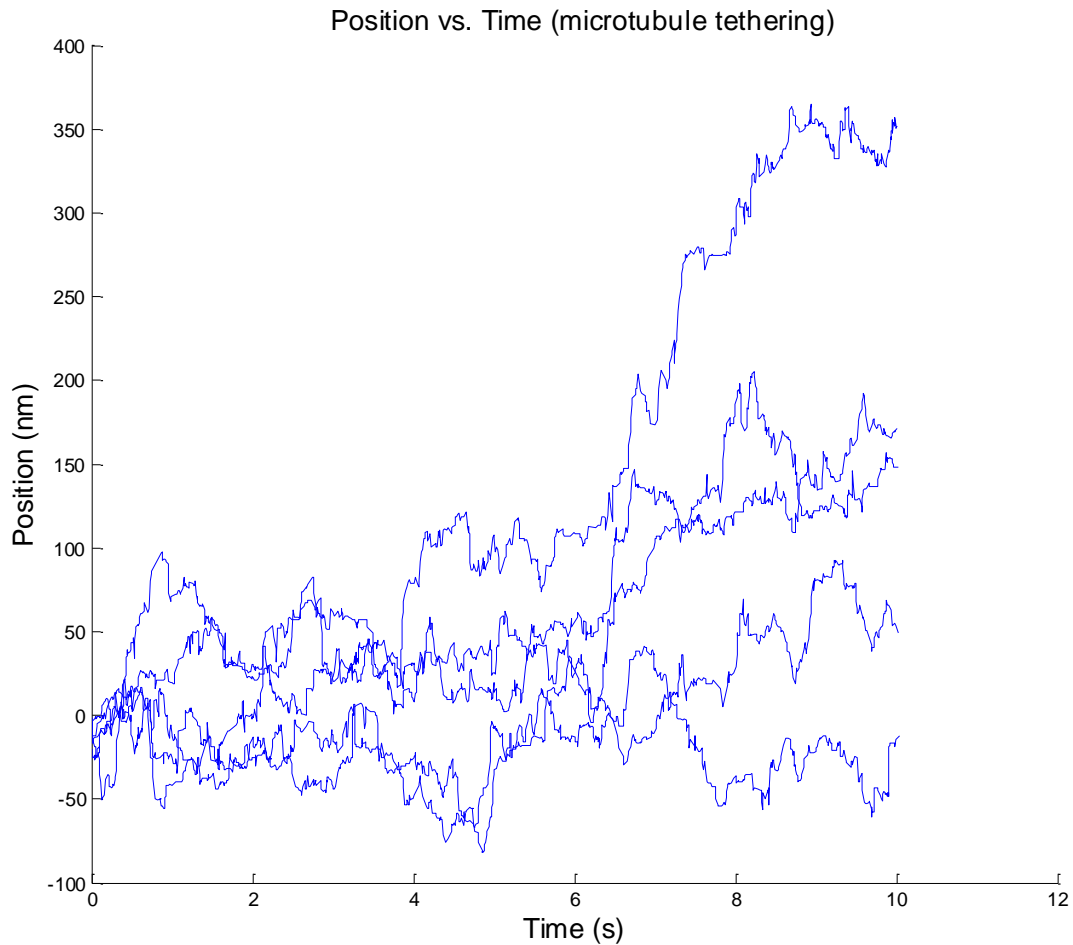
Figure 14 shows the processive, unidirectional runs and frequent switches in direction that are associated with bidirectional transport. It does not, however, show the pauses that are associated with bidirectional transport. Figure 14 shows that a single kinesin pulling against 6 dynein motors results in a mean velocity near 0.

Simulations of varying numbers of motors reveal that, while run lengths and attachment time do depend on the number of motors bound to the cargo, it is not necessary to have antagonistic motors. That is, the

addition of a motor protein does not enhance motility in the opposite direction. This is inconsistent with experimental results and demonstrates the limitations of the tug-of-war model.

### 3.2.2: Microtubule Tethering Model

Figure 15 shows the results of the same test (five runs with one kinesin motor and six dynein motors) performed for the microtubule tethering model.

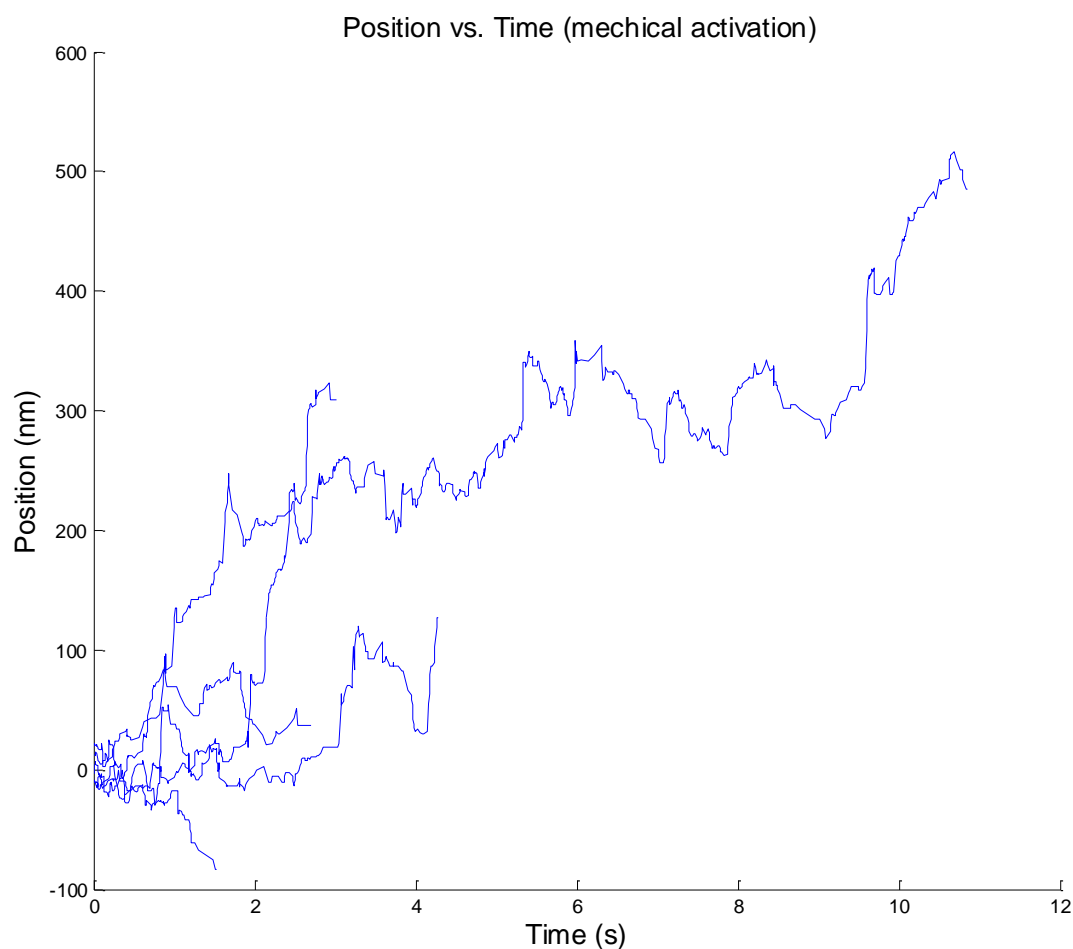


**Figure 15. Simulated position vs. time for microtubule tethering model with a single kinesin motor pulling against six dynein motors. In the microtubule tethering model, motors pass through a non-stepping diffusive state when detaching from the microtubule.**

As Figure 15 shows, the results of the microtubule tethering model are similar to those of the tug-of-war model. The most immediately apparent difference is that the addition of a diffusive state makes it very unlikely for all of the motors to come detached from the microtubule. In addition, multiple motors of the same type are less capable of working together cooperatively, allowing a single kinesin motor to overpower six dynein motors. Even when the number of dyneins is increased beyond six, a single kinesin is still frequently able to overpower them, which is not the case for the tug-of-war model. The exact reason for this behavior is not known at this time.

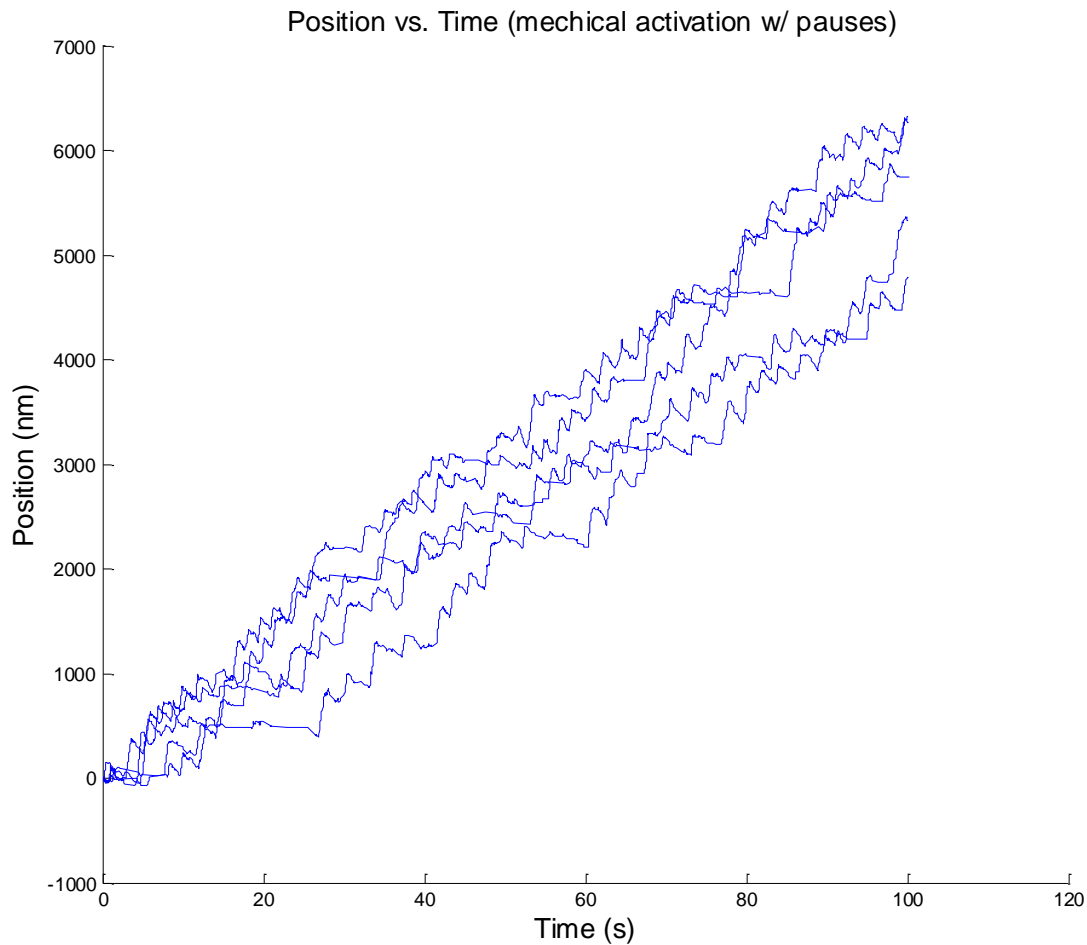
### 3.2.3: Mechanical Activation Model

Next, the qualitative features of the mechanical activation model were examined. Figure 16 shows five runs with a single kinesin and six dyneins for the mechanical activation model. Unlike the previous tests, the simulation was allowed to run until all motors had detached from the microtubule, regardless of how much time had passed.



**Figure 16. Simulated position vs. time for mechanical activation model with a single kinesin motor pulling against six dynein motors. In the mechanical activation model, motor attachment rates depend on the cargo velocity in the opposite direction.**

As this graph shows, it is far more likely for all of the motors in the mechanical activation model to detach from the microtubule than in the other simulations. Based on these results, the mechanical activation model was changed to allow for motors to reattach to the microtubule even after all motors had come detached. This was done in the hope that this would result in pauses similar to those seen in *in vivo* experiments. The results of this alteration to the simulation are shown in Figure 17. In order to allow enough time for multiple pauses to occur, each simulation was allowed to continue for 100 seconds.

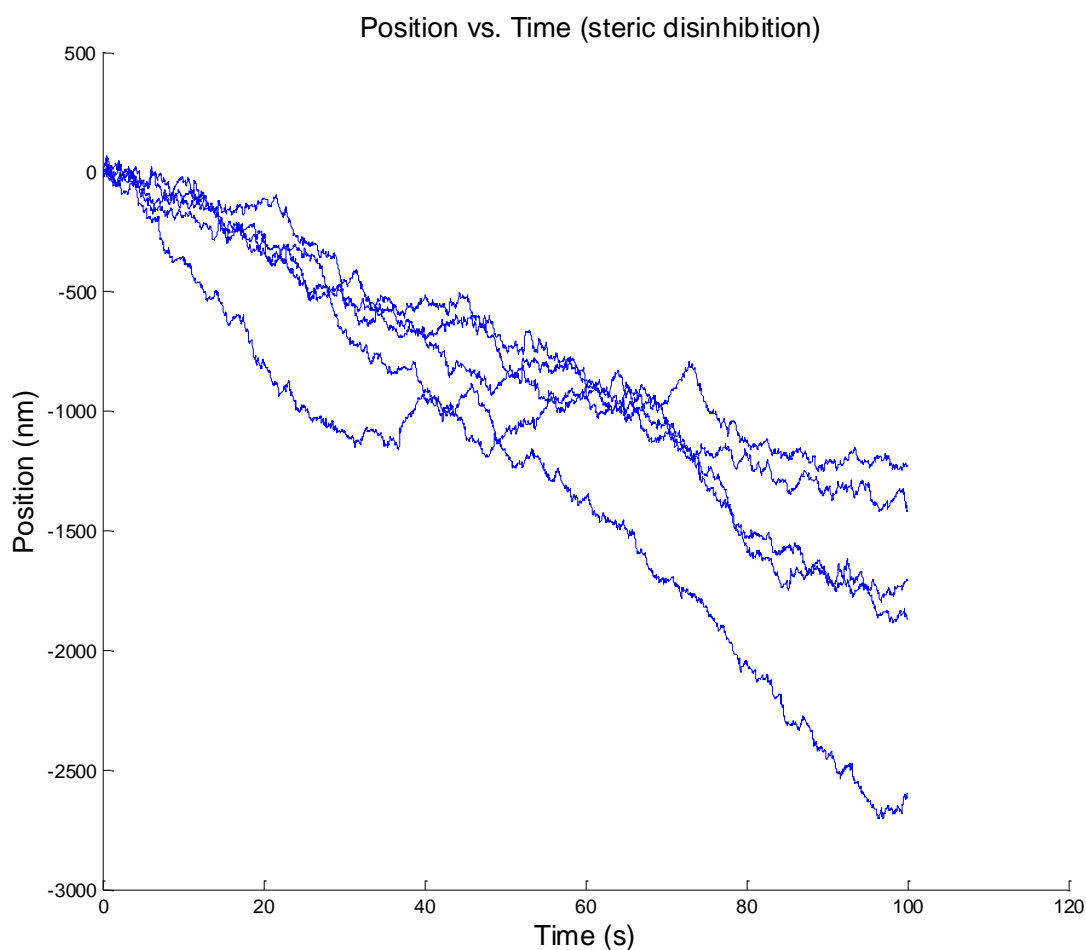


**Figure 17. Simulated position vs. time for mechanical activation model including pauses with a single kinesin motor pulling against six dynein motors. Motors were allowed to reattach even if all motors became detached. This resulted in the appearance of clear pauses.**

All of these runs show multiple pauses. Most of the pauses are brief, but some continue for several seconds. These results show a qualitative resemblance to experimental results, so we can conclude that the mechanical activation model is promising and warrants more in-depth investigation. In the future, attachment and detachment rates can be changed in order to match the frequency and duration of pauses to experimental results.

### 3.2.4: Steric Disinhibition Model

Finally, the same five runs were carried out for the steric disinhibition model. As was the case for the mechanical activation model, motors were allowed to rebind to the microtubule even if all motors were detached. However, testing showed that this was unlikely to ever occur. The results of the five runs are shown in Figure 18.



**Figure 18. Simulated position vs. time for steric disinhibition model with a single kinesin motor pulling against six dynein motors. In the steric disinhibition model, motor attachment rates depend on the number of oppositely-directed motors bound to the cargo.**

Unlike the other models, the changes made in the steric disinhibition model decrease kinesin's ability to overpower multiple dyneins, resulting in a mean velocity below 0. This is likely due to the fact that dynein's attachment rate is always at its maximum, which is greater than the attachment rate in the tug-of-

war model. This is also the case for kinesin, but it seems that this change favors the larger number of dynein motors. While the steric disinhibition model does not show the pauses that we hope to see, tests with no kinesin motors (and with dissociation of the cargo permitted) show that the cargo is rarely able to remain associated with the microtubule for even a full second. These results are consistent with the hypothesis that the paradox of codependence results from a mechanical activation mechanism.

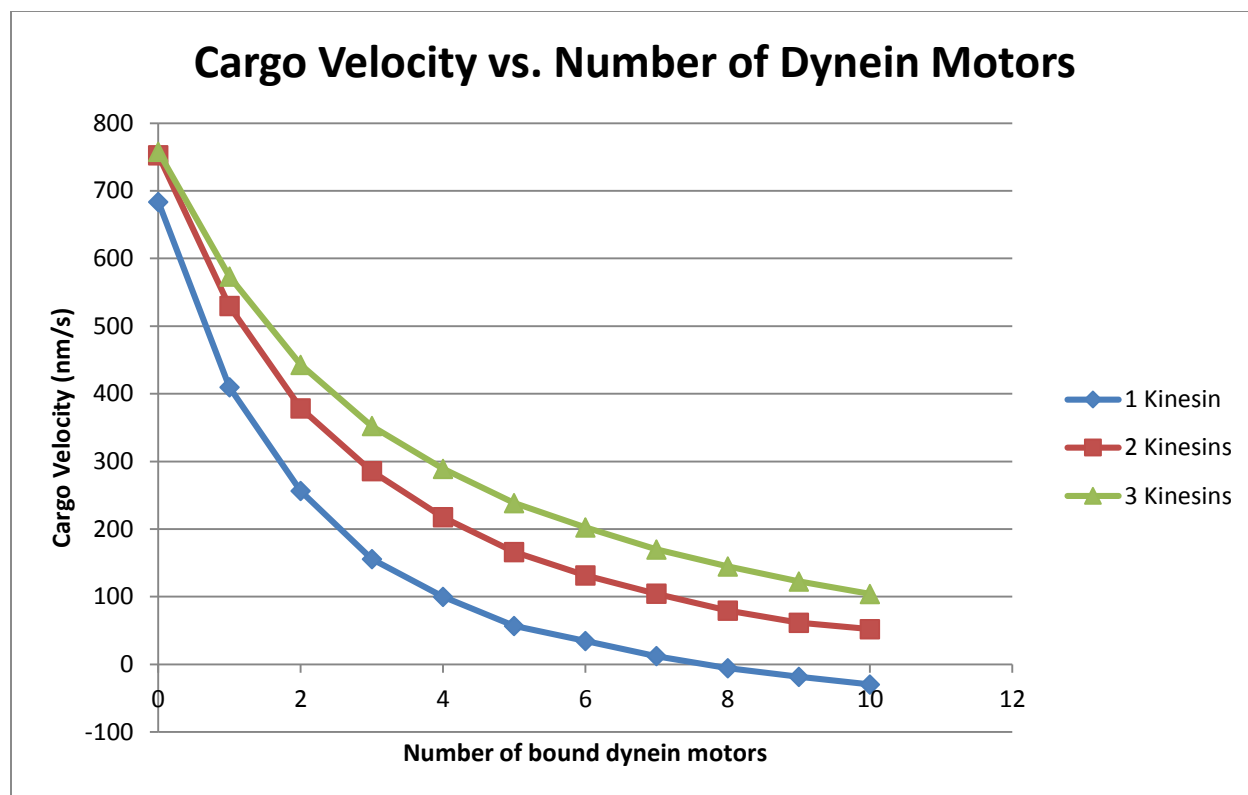
### 3.3: Analysis of Mechanical Activation Model

After performing a simple qualitative comparison, a more in-depth quantitative comparison was carried out. This comparison involved plotting the velocity for each model versus the number of dynein motors for a given number of kinesin motors. This allowed us to see whether either model exhibited the paradox of codependence. Due to time constraints, this comparison was limited to the tug-of-war and mechanical activation models. In addition, for the sake of simplicity, the motor tethers were modeled as Hookean springs rather than worm-like-chains. Note that these results were obtained using the updated dynein model 2. This analysis was performed using parameters for kinesin-1, as well as parameters for kinesin-2.

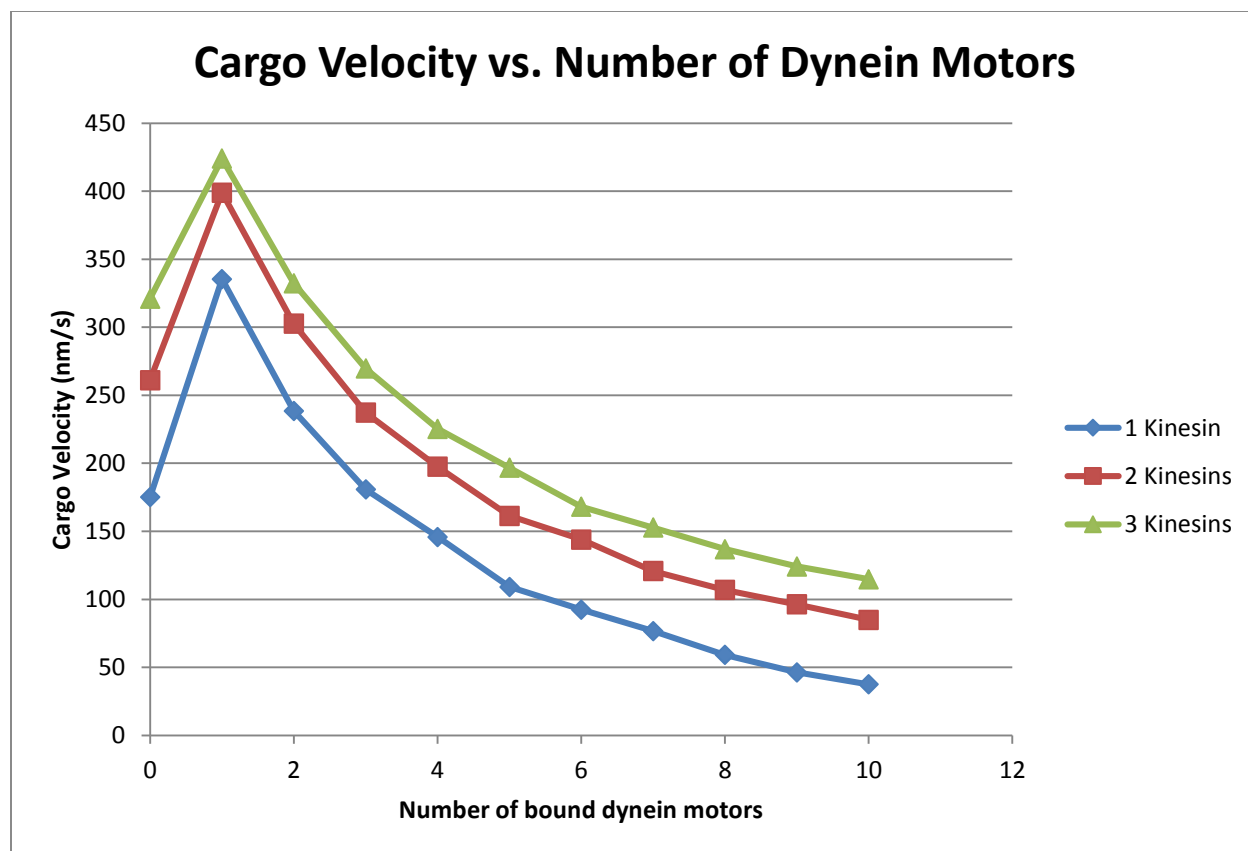
#### 3.3.1: Kinesin-1

Figure 19 shows the effect of an increasing number of dyneins on the cargo velocity for one, two, or three kinesin motors. Figure 20 shows the results of the same analysis carried out on the mechanical activation simulation.



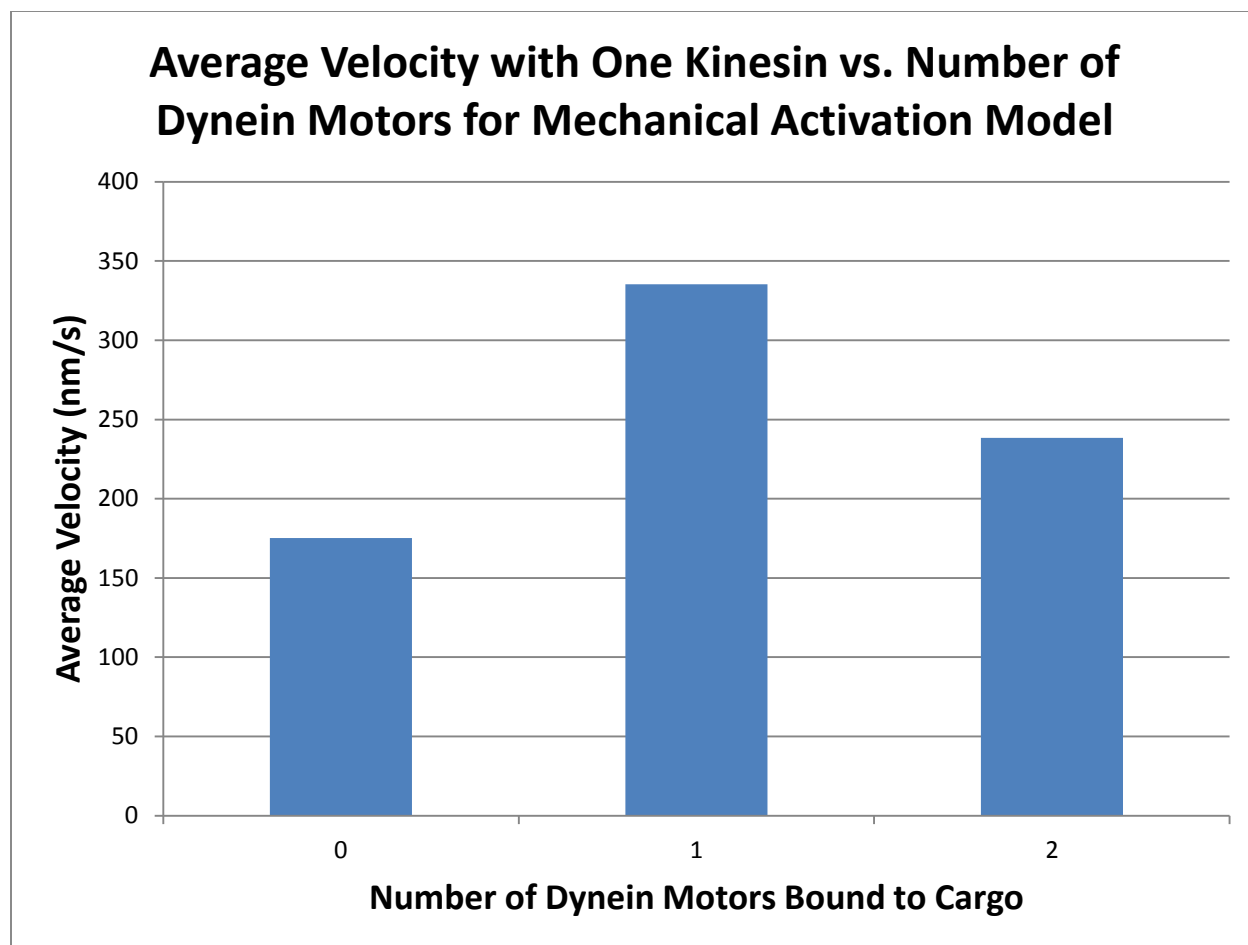


**Figure 19.** Effect of introducing dynein motors on cargo velocity in tug-of-war simulation. The introduction of additional dyneins motors to the tug-of-war model results in a decreased average cargo velocity in the plus direction.



**Figure 20.** Effect of introducing dynein motors on cargo velocity in mechanical activation simulation. The introduction of a small number of dynein motors to the mechanical activation model results in an increased cargo velocity in the plus direction.

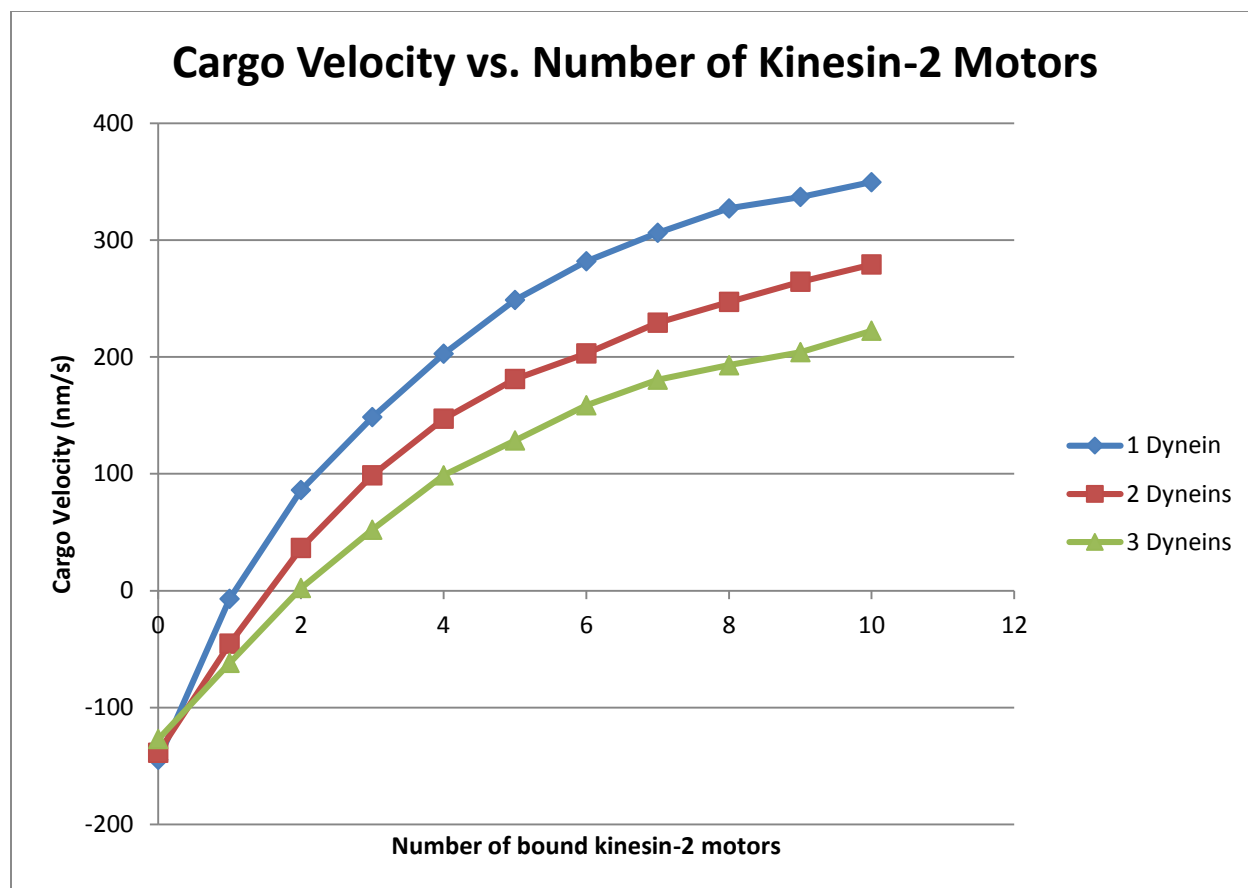
As these figures show, introduction of dynein motors to the tug-of-war simulation always results in reduced motility in the plus direction. However, the introduction of one or even two dynein motors to the mechanical activation simulation actually enhances motility in the plus direction. The relationship between anterograde velocity and the number of dynein motors bound to the cargo is illustrated in Figure 21. In addition, while introducing a second or third kinesin motor enhanced motility in the plus direction, the introduction of a dynein motor results in a greater enhancement in plus-directed motility.



**Figure 21.** Average velocity with one kinesin bound to cargo vs. number of dynein motors bound to cargo using the mechanical activation model. The introduction of a small number of dynein motors results in an increase in anterograde velocity, demonstrating the paradox of codependence.

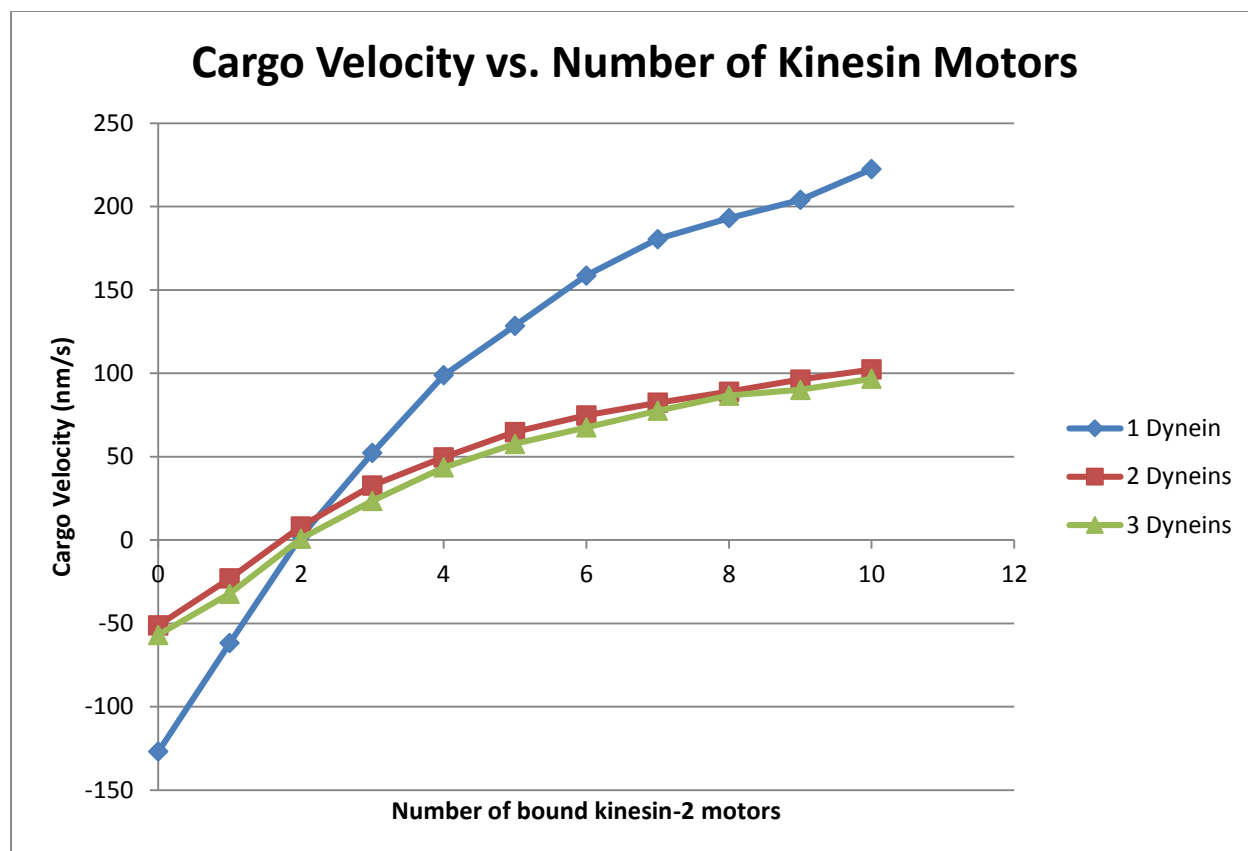
### 3.3.2: Kinesin-2

The tug-of-war and mechanical activation simulations were also compared for dynein pulling against kinesin-2. Kinesin-2 has a detachment rate that is much more dependent on force than that of kinesin-1. Due to kinesin-2's different parameters, we predicted that it might exhibit different behaviors in both the tug-of-war and mechanical activation models. It was found that one dynein could overpower one kinesin-2. As such, the velocity was plotted as a function of an increasing number of kinesin-2 motors pulling against a fixed number of dynein motors. The results of this analysis for the tug-of-war model are shown in Figure 22.



**Figure 22. Effect of introducing kinesin-2 motors on cargo velocity in tug-of-war simulation. The introduction of additional kinesin-2 motors to the tug-of-war model results in a positive shift in velocity.**

Figure 22 shows that the introduction of an increasing number of kinesin-2 motors pulling against a fixed number of dynein motors results in reduced retrograde velocity and eventually increased anterograde velocity before ultimately reaching an asymptote. These results mirror the results seen for the tug-of-war model using kinesin-1. Figure 23 shows the results of the same analysis performed using the mechanical activation model.



**Figure 23. Effect of introducing kinesin-2 motors on cargo velocity in mechanical activation simulation. The introduction of additional kinesin-2 motors to the mechanical activation model results in a positive shift in velocity, contrasting with the results seen for kinesin-1.**

In contrast to the results seen when using kinesin-1 parameters, these results show a more positive velocity, rather than a more negative velocity, upon introducing a small number of kinesin-2 motors.

These results do not display the paradox of codependence. This may be explained by the fact the kinesin-2 and dynein are much more closely matched than are kinesin-1 and dynein. Given that the motors are close in strength, any benefits gains from the effects of mechanical activation may be overcome by the force applied by the opposing motor.

## Chapter 4: Conclusion/Future Work

The goal of this thesis was to develop MATLAB simulations that demonstrate the shortcomings of the tug-of-war model and that demonstrate the potential for alternative models to address these shortcomings. The tug-of-war simulation showed none of the pauses expected based on experimental results. In addition, the introduction of additional motors to the tug-of-war model always results in diminished motility in the opposite direction. Thus, the tug-of-war simulation does not display the paradox of codependence. However, the mechanical activation simulations displayed both pauses and the paradox of codependence. As such, we can conclude that the mechanical activation model is viable and warrants further investigation.

Following my graduation, this project will be continued by another undergraduate student working in Dr. Hancock's laboratory. There are a number of steps that could be taken in the future in order to make the simulations more realistic. One such step would be to introduce diffusion. This would serve to make the simulations more realistic and could allow the velocities of individual detached motors to be determined, rather than relying on the cargo velocity. This would involve very short, fixed time steps, rather than the variable time steps currently used. This would greatly increase the time required to carry out a simulation, so I would only recommend introducing diffusion if it is believed that it would make a significant difference. Another change that could be made would be adding code to simulate the activity of dynactin or other microtubule-associated proteins. It would also be ideal to allow for dynein to have a variable step size. Recall that, while dynein's step size was set to a constant value in these simulations for the sake of simplicity, this does not accurately portray the behavior of actual dynein motors.<sup>28</sup>

In addition to these and any other changes that are made to the existing models, future work will also comprise a large portion of the data collection. Similar analyses to those performed during this project will need to be carried out for the steric disinhibition and microtubule tethering simulations. Thus far,

only the mechanical activation model has shown results that are characteristic of the paradox of codependence, specifically increased plus-directed velocities associated with the introduction of dynein motors. However, other models may also be found to display this behavior once testing is complete. In addition, more in-depth analyses will need to be carried on all of the models, including the mechanical activation model. One such analysis could involve varying motor attachment and detachment rates in order to produce simulated results that more closely replicate the pattern of runs and pauses seen in experimental data. Parameters such as pause frequency and pause duration would need to be recorded and compared to corresponding parameters for experimental data.

Ultimately, there is still a great deal of work to be done on these simulations, so it is difficult to predict exactly what the results of this project will be. It bears repeating that the models being tested through these simulations are all hypothetical at this time. However, these simulations will serve to show which of these hypothetical models have the potential to be accurate representations of the mechanisms underlying bidirectional transport. In fact, the results already obtained have shown that the mechanical activation model holds great promise. The ultimate goal of this project is that we will be able to identify key mechanisms and rate constants that result in the paradox of codependence. Once these have been identified, we will be able to make predictions using the simulations that can be verified against experimental results. This will hopefully lead to the understanding and treatment of any number of neurodegenerative diseases.

## References

1. C.L. Asbury, A.N. Fehr, and S.M. Block (2003). *Science*. Dec 19 2003: 2130-2134.
2. Oiwa, K. & Sakakibara, H. Recent progress in dynein structure and mechanism. *Cell Biol* **17**, 98-103 (2005)
3. Vale, R.D. (2003). The molecular motor toolbox for intracellular transport. *Cell* **112**, 467-480.
4. Hancock, W.O. Bidirectional cargo transport: moving beyond tug-of-war. *Nat Cell Biol*, in-press
5. Welte, M.A., Gross, S.P., Postner, M., Block, S.M. & Wieschaus, E.F. Developmental regulation of vesicle transport in *Drosophila* embryos: forces and kinetics. *Cell* **92**, 547-57 (1998).
6. Martin, M. et al. Cytoplasmic dynein, the dynactin complex, and kinesin are interdependent and essential for fast axonal transport. *Mol Biol Cell* **18**, 2081-9 (2007).
7. Klumpp, S. & Lipowsky, R. Cooperative cargo transport by several molecular motors. *Proc Natl Acad Sci U S A* **102**, 17284-9 (2005).
8. Müller, M.J., Klumpp, S. & Lipowsky, R. Tug-of-war as a cooperative mechanism for bidirectional cargo transport by molecular motors. *Proc Natl Acad Sci U S A* **105**, 4609-14 (2008)
9. Müller, M.J., Klumpp, S. & Lipowsky, R. Bidirectional transport by molecular motors: enhanced processivity and response to external forces. *Biophys J* **98**, 2610-8 (2010).
10. Hendricks, A.G. et al. Motor coordination via a tug-of-war mechanism drives bidirectional vesicle transport. *Curr Biol* (2010).
11. Kunwar, A. et al. Mechanical stochastic tug-of-war models cannot explain the bidirectional lipid-droplet transport. *Proc Natl Acad Sci U S A* **108**, 18960-5 (2011).
12. Ally, S., Larson, A.G., Barlan, K., Rice, S.E. & Gelfand, V.I. Opposite-polarity motors activate one another to trigger cargo transport in live cells. *J Cell Biol* **187**, 1071-82 (2009).
13. Okada, Y. & Hirokawa, N. Mechanism of the single-headed processivity: diffusional anchoring between the K-loop of kinesin and the C terminus of tubulin. *Proc Natl Acad Sci U S A* **97**, 640-5 (2000).



14. Kapitein, L.C. et al. Microtubule cross-linking triggers the directional motility of kinesin-5. *J Cell Biol* **182**, 421-8 (2008).
15. Lu, H., Ali, M.Y., Bookwalter, C.S., Warshaw, D.M. & Trybus, K.M. Diffusive movement of processive kinesin-1 on microtubules. *Traffic* **10**, 149-38 (2009).
16. Culver-Hanlon, T.L., Lex, S.A., Stephens, A.D., Quintyne, N.J. & King, S.J. A microtubule-binding domain in dynactin increases dynein processivity by skating along microtubules. *Nat Cell Biol* **8**, 264-70 (2006).
17. Andreasson, J.O., Shastry, S., Hancock, W.O., Block, S.M. The Mechanochemical Cycle of Mammalian Kinesin-2 KIF3A/B under Load. *Curr Biol* (2015).
18. Coy, D.L., Hancock, W.O., Wagenbach, M. & Howard, J. Kinesin's tail domain is an inhibitory regulator of the motor domain. *Nat Cell Biol* **1**, 288-92 (1999).
19. Hackney, D.D., & Stock, M.F. Kinesin's IAK tail domain inhibits initial microtubule-stimulated ADP release. *Nat Cell Biol* **2**, 257-60 (2000).
20. Egan, M.J., Tan, K. & Reck-Peterson, S.L. Lis1 acts as a "clutch" between the ATPase and microtubule-binding domains of the dynein motor. *Cell Biol* **22**, 2221-30 (2012).
21. Ligon, L.A., Tokito, M., Finklestein, J.M., Grossman, F.E. & Holzbaur, E.L. A direct interaction between cytoplasmic dynein and kinesin-1 may coordinate motor activity. *J Biol Chem* **279**, 19201-8 (2004).
22. Deacon, S.W. et al. Dynactin is required for bidirectional organelle transport. *J Cell Biol* **160**, 297-301 (2003)
23. van Spronsen, M. et al. TRAK/Milton motor-adaptor proteins steer mitochondrial trafficking to axons and dendrites. *Neuron* **77**, 485-502 (2013).
24. Gillespie, D.T. A General Method for Numerically Simulating the Stochastic Time Evolution of Coupled Chemical Reactions. *J Comp Phys* **22**, 403-34 (1976).
25. Carter, N.J. & Cross, R.A. Mechanics of the kinesin step. *Nature* **435**, 308-12 (2005).

26. Arpag, G., Shastry, S., Hancock, W.O. & Tuzel, E. Transport by populations of fast and slow kinesins uncovers novel family-dependent motor characteristics important for in vivo function. *Biophys Journal* **107**, 1896-1904 (2014).
27. Toba, S., Watanabe, T.M., Yamaguchi-Okimoto, L., Toyoshima, Y.Y. & Higuchi, H. Overlapping hand-over-hand mechanism of single molecular motility of cytoplasmic dynein. *Proc Natl Acad Sci U S A* **103** 5741-5 (2006).
28. Mallik, R., Petrov, D., Lex, S.A., King, S.J. & Gross, S.P. Building complexity: an in vitro study of cytoplasmic dynein with in vivo implications. *Curr Biol* **15**, 2075-2085 (2005)
29. Reck-Peterson, S.L., Yildiz, A., Carter, A.P., Gennerich, A., Zhang, N. & Vale, R.D. Single-molecule analysis of dynein processivity and stepping behavior. *Cell* **126**, 335-48 (2006)
30. Gennerich, A., Carter, A.P., Reck-Peterson, S. & Vale, R.D. Force-induced bidirectional stepping of cytoplasmic dynein. *Cell* **131**, 952-965 (2007).
31. Coppin, C.M., Finer, J.T., Spudich, J.A. & Vale, R.D. Detection of sub-8-nm movement of kinesin by high-resolution optical-trap microscopy. *Proc Natl Acad Sci U S A* **93** 1913-7 (1996)
32. Kutys, M.L., Fricks, J. & Hancock, W.O. Monte Carlo analysis of neck linker extension in kinesin molecular motors. *PLoS Comput Biol* **6** doi:10.1371/journal.pcbi.1000980

## Appendix A: MATLAB Code for Mechanical Activation Model

```

function [Simu] = cargo_bsm_wlc_mech_back_uncom(F,kins,dyns)

pfs = 8;

motor_type_ss = [pfs, -pfs];

plk = 0.7;
clk = 84;
pld = 0.7;
cld = 48;

valid = 0;

while valid == 0

    valid = 1;

    motX = pfs*round(3*randn(1,kins + dyns));
    motA = ones(1,kins + dyns);
    motT = [1*ones(1,kins),2*ones(1,dyns)];
    motS = [motor_type_ss(1)*ones(1,kins),motor_type_ss(2)*ones(1,dyns)];
    cX = sum(motA.*motX)/sum(motA);
    motF = zeros(1,length(motT));

    for mm = 1:length(motT)
        switch motT(mm)
            case 1
                motF(mm) = motA(mm)*(4.11433/plk)*(0.25*(1-abs(motX(mm)-cX)/clk)^(-2)-0.25+abs(motX(mm)-
cX)/clk);
                if (motX(mm)-cX) < 0
                    motF(mm) = -motF(mm);
                end
            case 2
                motF(mm) = motA(mm)*(4.11433/pld)*(0.25*(1-abs(motX(mm)-cX)/cld)^(-2)-0.25+abs(motX(mm)-
cX)/cld);
                if (motX(mm)-cX) < 0
                    motF(mm) = -motF(mm);
                end
            end
        end

    cF = sum(motF)-F;

    motF_plus = zeros(1,length(motT));
    for mm = 1:length(motT)
        switch motT(mm)
            case 1
                motF_plus(mm) = motA(mm)*(4.11433/plk).*(0.25*(1-abs(motX(mm)-(cX+0.1))/clk).^(-2)-
0.25+abs(motX(mm)-(cX+0.1))/clk);
                if (motX(mm)-(cX+0.1)) < 0
                    motF_plus(mm) = -motF_plus(mm);

```

```

        end
    case 2
        motF_plus(mm) = motA(mm)*(4.11433/pld).*(0.25*(1-abs(motX(mm)-(cX+0.1))/cld).^(-2)-
0.25+abs(motX(mm)-(cX+0.1))/cld);
        if (motX(mm)-(cX+0.1)) < 0
            motF_plus(mm) = -motF_plus(mm);
        end
    end
end

cF_plus = sum(motF_plus)-F;

motF_minus = zeros(1,length(motT));
for mm = 1:length(motT)
    switch motT(mm)
        case 1
            motF_minus(mm) = motA(mm)*(4.11433/plk).*(0.25*(1-abs(motX(mm)-(cX-0.1))/clk).^(-2)-
0.25+abs(motX(mm)-(cX-0.1))/clk);
            if (motX(mm)-(cX-0.1)) < 0
                motF_minus(mm) = -motF_minus(mm);
            end
        case 2
            motF_minus(mm) = motA(mm)*(4.11433/pld).*(0.25*(1-abs(motX(mm)-(cX-0.1))/cld).^(-2)-
0.25+abs(motX(mm)-(cX-0.1))/cld);
            if (motX(mm)-(cX-0.1)) < 0
                motF_minus(mm) = -motF_minus(mm);
            end
        end
    end
end

cF_minus = sum(motF_minus)-F;

if abs(cF_plus) < abs(cF)
    while abs(cF) > 1
        cX = cX + 0.1;
        for mm = 1:length(motT)
            switch motT(mm)
                case 1
                    if abs(motX(mm)-cX) >= clk
                        valid = 0;
                        disp('This should not happen often')
                        break
                    end
                case 2
                    if abs(motX(mm)-cX) >= cld
                        valid = 0;
                        disp('This should not happen often')
                        break
                    end
                end
            end
        end
    end

    if valid == 0
        break
    end
end

```

```

motF = zeros(1,length(motT));
for mm = 1:length(motT)
    switch motT(mm)
        case 1
            motF(mm) = motA(mm)*(4.11433/plk)*(0.25*(1-abs(motX(mm)-cX)/clk)^(-2)-
0.25+abs(motX(mm)-cX)/clk);
            if (motX(mm)-cX) < 0
                motF(mm) = -motF(mm);
            end
        case 2
            motF(mm) = motA(mm)*(4.11433/pld)*(0.25*(1-abs(motX(mm)-cX)/cld)^(-2)-
0.25+abs(motX(mm)-cX)/cld);
            if (motX(mm)-cX) < 0
                motF(mm) = -motF(mm);
            end
        end
    end
end

cF = sum(motF)-F;
end
elseif abs(cF_minus) < abs(cF)
while abs(cF) > 1
    cX = cX - 0.1;
    for mm = 1:length(motT)
        switch motT(mm)
            case 1
                if abs(motX(mm)-cX) >= clk
                    valid = 0;
                    disp('This should not happen often')
                    break
                end
            case 2
                if abs(motX(mm)-cX) >= cld
                    valid = 0;
                    disp('This should not happen often')
                    break
                end
            end
        end
    end

    if valid == 0
        break
    end

    motF = zeros(1,length(motT));
    for mm = 1:length(motT)
        switch motT(mm)
            case 1
                motF(mm) = motA(mm)*(4.11433/plk)*(0.25*(1-abs(motX(mm)-cX)/clk)^(-2)-
0.25+abs(motX(mm)-cX)/clk);
                if (motX(mm)-cX) < 0
                    motF(mm) = -motF(mm);
                end
            case 2

```

```

        motF(mm) = motA(mm)*(4.11433/pld)*(0.25*(1-abs(motX(mm)-cX)/cld)^(-2)-
0.25+abs(motX(mm)-cX)/cld);
        if (motX(mm)-cX) < 0
            motF(mm) = -motF(mm);
        end
    end
end

    cF = sum(motF)-F;
end
end
end

motF = zeros(1,length(motT));
for mm = 1:length(motT)
    switch motT(mm)
        case 1
            motF(mm) = motA(mm)*(4.11433/plk)*(0.25*(1-abs(motX(mm)-cX)/clk)^(-2)-0.25+abs(motX(mm)-
cX)/clk);
            if (motX(mm)-cX) < 0
                motF(mm) = -motF(mm);
            end
        case 2
            motF(mm) = motA(mm)*(4.11433/pld)*(0.25*(1-abs(motX(mm)-cX)/cld)^(-2)-0.25+abs(motX(mm)-
cX)/cld);
            if (motX(mm)-cX) < 0
                motF(mm) = -motF(mm);
            end
        end
    end
end

Ks = zeros(kins + dyns,4);

Simu = struct;
Simu(1).motX = motX;
Simu(1).motA = motA;
Simu(1).motF = motF;
Simu(1).motT = motT;
Simu(1).cX = cX;
Simu(1).t = 0;
cc = 1;
uu = 0;
uuu = 0;
sim_time = 0;
while Simu(cc).t < 10

    if length(sim_time) > 30
        if sim_time(length(sim_time)) == sim_time(length(sim_time)-30)
            Simu(cc)
            error('infinite loop')
        end
    end
end

```

```

motX1 = motX;
motA1 = motA;
motF1 = motF;
cX2 = cX;

for mm = 1:length(motT)
    if cc == 1
        vel = 0;
    else
        vel = (Simu(cc).motX(mm) - Simu(cc-1).motX(mm))/(Simu(cc).t - Simu(cc-1).t);
    end
    Ks = setRates(mm,motT,motA,motF,vel,Ks);

    if motA(mm) == 0
        motX(mm) = cX;
    end
end

Dw = (1./Ks).*log(1./rand(size(Ks)));
if size(Ks,1) == 1
    motnum = 1;
    ratenum = find(Dw == min(Dw));
else
    [motnum,ratenum] = minarr(Dw);
end

if Dw(motnum,ratenum) < 0.0001
    Dw(motnum,ratenum) = 0.0001;
end

[motX,motA,valid] = simEvent(motnum,ratenum,motX,motA,motT,motS,cX,pfs,clk,cld);

if valid == 0
    uu = uu + 1;
end

motF = zeros(1,length(motT));
for mm = 1:length(motT)
    switch motT(mm)
        case 1
            motF(mm) = motA(mm)*(4.11433/plk)*(0.25*(1-abs(motX(mm)-cX)/clk)^(-2)-0.25+abs(motX(mm)-
cX)/clk);
            if (motX(mm)-cX) < 0
                motF(mm) = -motF(mm);
            end
        case 2
            motF(mm) = motA(mm)*(4.11433/pld)*(0.25*(1-abs(motX(mm)-cX)/cld)^(-2)-0.25+abs(motX(mm)-
cX)/cld);
            if (motX(mm)-cX) < 0
                motF(mm) = -motF(mm);
            end
    end
end
end

```

```

cF = sum(motF)-F;

motF_plus = zeros(1,length(motT));
for mm = 1:length(motT)
    switch motT(mm)
        case 1
            motF_plus(mm) = motA(mm)*(4.11433/plk).*(0.25*(1-abs(motX(mm)-(cX+0.1))/clk).^(-2)-
0.25+abs(motX(mm)-(cX+0.1))/clk);
            if (motX(mm)-(cX+0.1)) < 0
                motF_plus(mm) = -motF_plus(mm);
            end
        case 2
            motF_plus(mm) = motA(mm)*(4.11433/pld).*(0.25*(1-abs(motX(mm)-(cX+0.1))/cld).^(-2)-
0.25+abs(motX(mm)-(cX+0.1))/cld);
            if (motX(mm)-(cX+0.1)) < 0
                motF_plus(mm) = -motF_plus(mm);
            end
        end
    end
end

cF_plus = sum(motF_plus)-F;

motF_minus = zeros(1,length(motT));
for mm = 1:length(motT)
    switch motT(mm)
        case 1
            motF_minus(mm) = motA(mm)*(4.11433/plk).*(0.25*(1-abs(motX(mm)-(cX-0.1))/clk).^(-2)-
0.25+abs(motX(mm)-(cX-0.1))/clk);
            if (motX(mm)-(cX-0.1)) < 0
                motF_minus(mm) = -motF_minus(mm);
            end
        case 2
            motF_minus(mm) = motA(mm)*(4.11433/pld).*(0.25*(1-abs(motX(mm)-(cX-0.1))/cld).^(-2)-
0.25+abs(motX(mm)-(cX-0.1))/cld);
            if (motX(mm)-(cX-0.1)) < 0
                motF_minus(mm) = -motF_minus(mm);
            end
        end
    end
end

cF_minus = sum(motF_minus)-F;

if abs(cF_plus) < abs(cF)
    while abs(cF) > 1
        cX = cX + 0.1;
        for mm = 1:length(motT)
            switch motT(mm)
                case 1
                    if abs(motX(mm)-cX) >= clk
                        valid = 0;
                        uuu = uuu + 1;
                        break
                    end
                case 2
                    if abs(motX(mm)-cX) >= cld

```



```

        valid = 0;
        uuu = uuu + 1;
        break
    end
end
end

if valid == 0
    break
end

motF = zeros(1,length(motT));
for mm = 1:length(motT)
    switch motT(mm)
        case 1
            motF(mm) = motA(mm)*(4.11433/plk)*(0.25*(1-abs(motX(mm)-cX)/clk)^(-2)-
0.25+abs(motX(mm)-cX)/clk);
            if (motX(mm)-cX) < 0
                motF(mm) = -motF(mm);
            end
        case 2
            motF(mm) = motA(mm)*(4.11433/pld)*(0.25*(1-abs(motX(mm)-cX)/cld)^(-2)-
0.25+abs(motX(mm)-cX)/cld);
            if (motX(mm)-cX) < 0
                motF(mm) = -motF(mm);
            end
        end
    end
end

cF = sum(motF)-F;
end
elseif abs(cF_minus) < abs(cF)
    while abs(cF) > 1
        cX = cX - 0.1;
        for mm = 1:length(motT)
            switch motT(mm)
                case 1
                    if abs(motX(mm)-cX) >= clk
                        valid = 0;
                        uuu = uuu + 1;
                        break
                    end
                case 2
                    if abs(motX(mm)-cX) >= cld
                        valid = 0;
                        uuu = uuu + 1;
                        break
                    end
                end
            end
        end
    end

    if valid == 0
        break
    end
end

```

```

    motF = zeros(1,length(motT));
    for mm = 1:length(motT)
        switch motT(mm)
            case 1
                motF(mm) = motA(mm)*(4.11433/plk)*(0.25*(1-abs(motX(mm)-cX)/clk)^(-2)-
0.25+abs(motX(mm)-cX)/clk);
                if (motX(mm)-cX) < 0
                    motF(mm) = -motF(mm);
                end
            case 2
                motF(mm) = motA(mm)*(4.11433/pld)*(0.25*(1-abs(motX(mm)-cX)/cld)^(-2)-
0.25+abs(motX(mm)-cX)/cld);
                if (motX(mm)-cX) < 0
                    motF(mm) = -motF(mm);
                end
            end
        end
    end

    cF = sum(motF)-F;
end
end

if valid == 1
    motF = zeros(1,length(motT));
    for mm = 1:length(motT)
        switch motT(mm)
            case 1
                motF(mm) = motA(mm)*(4.11433/plk)*(0.25*(1-abs(motX(mm)-cX)/clk)^(-2)-0.25+abs(motX(mm)-
cX)/clk);
                if (motX(mm)-cX) < 0
                    motF(mm) = -motF(mm);
                end
            case 2
                motF(mm) = motA(mm)*(4.11433/pld)*(0.25*(1-abs(motX(mm)-cX)/cld)^(-2)-0.25+abs(motX(mm)-
cX)/cld);
                if (motX(mm)-cX) < 0
                    motF(mm) = -motF(mm);
                end
            end
        end
    end

    Simu(cc+1).motX = motX;
    Simu(cc+1).motA = motA;
    Simu(cc+1).motF = motF;
    Simu(cc+1).motT = motT;
    Simu(cc+1).cX = cX;
    Simu(cc+1).t = Simu(cc).t + Dw(motnum,ratenum);

    cc = cc + 1;
else
    motX = motX1;
    motA = motA1;
    motF = motF1;
    cX = cX2;

```

```

end
end

end

%% setRates

function Ks = setRates(mm,motT,motA,motF,vel,Ks)
k_kon_min = 0.1;
k_kon_max = 10;
d_kon_min = 0.1;
d_kon_max = 10;
switch motT(mm)
    % it's a kinesin
    case 1
        % if motor is detached
        if motA(mm) == 0
            % on rate
            if vel >= -0
                Ks(mm,1) = k_kon_min;
            elseif vel >= -212
                Ks(mm,1) = -((k_kon_max - k_kon_min)/212)*vel + k_kon_min;
            else
                Ks(mm,1) = k_kon_max;
            end
            % off rate
            Ks(mm,2) = 0;
            % rate at which it steps forward
            Ks(mm,3) = 0;
            % rate at which it steps backward
            Ks(mm,4) = 0;
        % if motor is attached
        else
            % set on rate to zero since its already on
            Ks(mm,1) = 0;
            % if its an assisting load
            if motF(mm) <= 0
                Ks(mm,3) = 100;
            % if its a light hindering load
            elseif motF(mm) <= 2.5
                Ks(mm,3) = 100*exp(-.162*motF(mm));
            elseif motF(mm) <= 7.76
                Ks(mm,3) = (1/.0036)*exp(-0.57*motF(mm));
            else
                Ks(mm,3) = 1/.3;
            end

            % define the backward stepping rate
            if motF(mm) <= 7.76
                Ks(mm,4) = Ks(mm,3)/(802*exp(-.95*motF(mm)));
            else
                Ks(mm,4) = 6.61;
            end
        end
    end
end

```

```

% define the off rate
if motF(mm) <= 0
    Ks(mm,2) = 0.79 - 1.56*motF(mm);
else
    Ks(mm,2) = 0.79*exp(motF(mm)/6.1);
end
end

% it's a dynein
case 2
    % negate the force because it prefers to walk to the (-) end
    motF(mm) = -1*motF(mm);
    % if motor is detached
    if motA(mm) == 0
        % on rate
        if vel <= 0
            Ks(mm,1) = d_kon_min;
        elseif vel <= 800
            Ks(mm,1) = ((d_kon_max - d_kon_min)/800)*vel + d_kon_min;
        else
            Ks(mm,1) = d_kon_max;
        end
        % off rate
        Ks(mm,2) = 0;
        % rate at which it steps forward
        Ks(mm,3) = 0;
        % rate at which it steps backward
        Ks(mm,4) = 0;
    % if motor is attached
    else
        % set on rate to zero since its already on
        Ks(mm,1) = 0;

        if motF(mm) <= 1
            step_rat = 2.015;
        elseif motF(mm) <= 10
            step_rat = -0.1707*motF(mm) + 2.186;
        else
            step_rat = 0.479;
        end

        if motF(mm) <= 0.3
            net_step = 21.678;
        elseif motF(mm) <= 1
            net_step = 31.606 - 33.095*motF(mm);
        else
            net_step = -1.489;
        end

        Ks(mm,3) = net_step./(1-1/step_rat);

        % define the backward stepping rate

        Ks(mm,4) = Ks(mm,3)/step_rat;
    end
end

```

```

% define the off rate
if motF(mm) <= 0
    Ks(mm,2) = exp(-motF(mm)/3);

elseif motF(mm) <= 1.7
    Ks(mm,2) = exp(0.8155*motF(mm));

else
    Ks(mm,2) = 1/(0.254*(1-exp(-motF(mm)/1.97)));

end
end

end

end

%% minarr

function [r,c] = minarr(X)

% find column where min is
Y = min(X);
c = find(Y == min(Y));

% only return one answer if there is a tie
if length(c) > 1
    X(:,c(2):end) = [];
end
c = c(1);

% find row where min is
Y = min(X');
r = find(Y == min(Y));

end

%% simEvent

function [motX,motA,valid] = simEvent(motnum,ratenum,motX,motA,motT,motS,cX,pfs,clk,cld)
valid = 1;
motnum;
ratenum;
switch ratenum
    % attach that motor to the microtubule
    case 1
        motA(motnum) = 1;
        motX(motnum) = pfs*round((cX + pfs*round(3*randn)))/pfs);
    % detach that motor from the microtubule
    case 2
        motA(motnum) = 0;

```

```

    motX(motnum) = cX;
% step that motor forward
case 3
    switch motT(motnum)
        case 1
            if abs(motX(motnum)+motS(motnum)-cX) < clk
                motX(motnum) = motX(motnum) + motS(motnum);
            else
                valid = 0;
            end
        case 2
            if abs(motX(motnum)+motS(motnum)-cX) < cld
                motX(motnum) = motX(motnum) + motS(motnum);
            else
                valid = 0;
            end
        end
% step that motor backward
case 4
    switch motT(motnum)
        case 1
            if abs(motX(motnum)-motS(motnum)-cX) < clk
                motX(motnum) = motX(motnum) - motS(motnum);
            else
                valid = 0;
            end
        case 2
            if abs(motX(motnum)-motS(motnum)-cX) < cld
                motX(motnum) = motX(motnum) - motS(motnum);
            else
                valid = 0;
            end
        end
    end
end
end
end

```

**BRANDON MENTLEY**

brandon.mentley@gmail.com

610-256-6959

School Address446 Blue Course Drive, Apt. 103  
State College, PA 16803Permanent Address2813 Sandpiper Drive  
Audubon, PA 19403**Objective:** I am seeking an internship or co-op opportunity for spring or summer of 2016 prior to entering the master's program in biomedical engineering at Boston University in fall 2016.**Education:** **The Pennsylvania State University** **University Park, PA****GPA 3.80** Member of the Schreyer Honors College

Bachelor of Science in Bioengineering, Chemical Engineering Option Expected December 2015

**Courses:** Bio-continuum Mechanics, Biomedical Instrumentation and Measurements, Mass Transport in Biological Systems, Cell and Molecular Bioengineering, Reaction Kinetics of Biological Systems, Physiological Simulation Laboratory (MATLAB and Simulink), Physiology, Differential Equations, Organic Chemistry, Biochemistry

- Experience:**
- **The Pennsylvania State University** – Undergraduate Research Assistant **2013 – Present**  
Created MATLAB scripts to simulate several models of the bidirectional motility of opposite-polarity motor proteins involved in intracellular transport
  - **The Pennsylvania State University** – Biomedical Senior Design Project **08/2014 – 12/2014**  
Worked with an interdisciplinary engineering team to redesign a fire helmet to reduce the risk of injuries resulting from oblique impacts to the head
  - **University of Pittsburgh Medical Center** – Research Fellow **05/2013 – 08/2013**  
Conducted two *in vivo* studies using optical coherence tomography to measure aqueous humor outflow structures in the human eye
  - **Qdoba Mexican Grill; Subway** – Line Server & Cashier **Summers 2011 & 2012**
  - **Valley Forge Martial Arts** – Assistant Instructor **2002 – 08/2011**

- Honors:**
- Penn State: President Sparks Award; President's Freshman Award; Dean's List
  - Valedictorian of high school class of 456 students
  - Eagle Scout in the Boy Scouts of America
  - National Merit Scholarship Finalist – Awarded Lockheed Martin National Merit Scholarship
  - AP Scholar with Distinction Award
  - Pennsylvania Higher Education Assistance Agency Certificate of Merit
  - World Tang Soo Do Region 8 Scholarship

**Publication:** Kagemann, L., Wang, B., Wollstein, G., Ishikawa, H., Mentley, B., Sigal, I., et al. Trabecular Meshwork Response to Pressure Elevation in the Living Human Eye. *J. Vis. Exp.* (100), e52611, doi:10.3791/52611 (2015).

- Activities:**
- **Member of the International Fraternity of Phi Gamma Delta** **2012 – Present**  
Chairman of Scholarship Committee; Involved in fundraising to benefit multiple charities
  - **Penn State Biomedical Engineering Society** **2012 – Present**
  - **Penn State–Methacton High School Alumni Organization** **2013**  
Assisted with the organization of fundraisers to benefit research and victims of pediatric cancer
  - **Penn State Outing Club** **2011 – 2012**
  - **Boy Scouts of America Troop 313** **2004 – 2011**  
Attained rank of Eagle Scout; Elected to Order of the Arrow Honor Society
  - **Valley Forge Martial Arts – Tang Soo Do (Korean martial art)** **2000 – Present**  
Attained rank of Third Degree Black Belt in Tang Soo Do and First Degree Black Belt in Sin Moo Hapkido; member of competition creativity demonstration team
  - **National Honor Society, Methacton High School** **2009 – 2011**