Advective flow: The nature of mass flow of organelles in the filamentous fungus *Neurospora crassa*[1].

Aryan Abadeh[2] and Roger R. Lew, Biology Department, York University

Revision 1.4 (20 September 2012)

OBJECTIVE

To explore the nature of mass flow in *Neurospora crassa* hyphae, by fluorescent mapping of the movement of both nuclei and mitochondria to quantify correlations between their movement velocities and their directions.

[2] Experiments were performed 01MAY2012 through 31AUG2012 in the Lew Laboratory (RRL email: planters@yorku.ca) and were funded in part by NSERC (Natural Sciences and Engineering Research Council)

INTRODUCTION

In order to survive, the filamentous hyphae of *Neurospora crassa* grow into new territories while forming an interconnected mycelium behind the colony edge where nutrients and osmolytes are actively absorbed to fuel continued growth. Depending on the growth conditions, the rate of this cellular expansion can vary. Osmotically created pressure inside the mycelial network is the major driving force for the cellular growth and expansion of N. *crassa*. Osmolytes are more concentrated inside the cell, so water enters the cell by osmosis to create the internal hydrostatic pressure (turgor) resulting in what is known as turgor-driven growth (Lew, 2011).

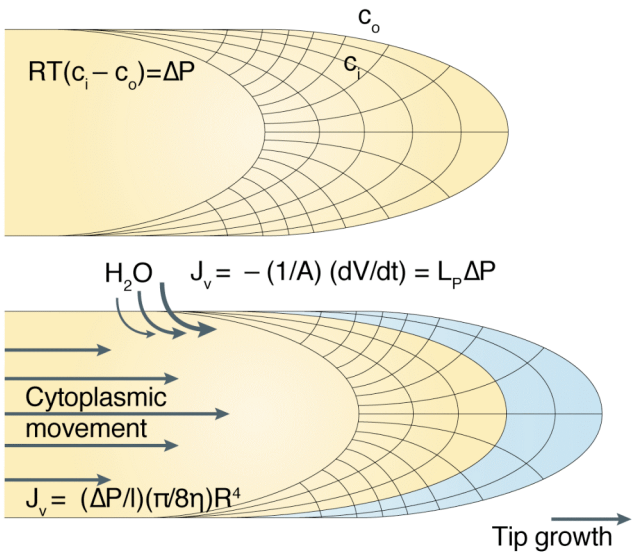

Fig. 1. Osmosis, intercellular pressure and cellular expansion. The interplay between the intercellular pressure and the constant supply of membrane material results, in the passive flow of cytoplasm and its organelles towards the apex tip. Diagram re-drawn from Lew (2011).

In order for this intercellular pressure to result in the hyphal expansion, membrane materials must be supplied consistently during cellular growth. The biochemical machinery of growth must be equitably distributed along the growing hypha. Among all organelles, the movement and positioning of nuclei and mitochondria are especially important because of their role in gene expression (Ramos-García et al, 2009) and the provision of energy for biochemical synthesis of DNA and protein (Diacumakos et al, 1965).

Movement and positioning of organelles in fungal cells may be either an active or passive process and scientists have always been faced with the question which of these two mechanisms plays the major role in hyphae growth?

To differentiate between the two processes, organelle movement involving ATP dependent motor proteins is known as an active movement, whereas movement involving cytoplasmic flow is considered passive because it is caused by osmolytes gradient and therefore only indirectly relies on the cell's ATP supply. Much of the research by cell biologists has led to the discovery of the motor proteins responsible

for the active movement of organelles, and as a result the role of cytoplasmic flow in organelle movement has been mostly ignored. The importance of passive movement and the interplay between these two processes remains yet to be answered.

Under the light microscope, cytoplasm appears to flow actively towards the growing apical tip at the colony's edge. Although the active movements of organelles in the cytoplasm can give the impression of mass flow, it is difficult to make a conclusion based on microscopic observations alone. Cytoplasmic bulk flow is defined as the movement of both the cytoplasm *and* the organelles in the cytoplasm independent of the cytoskeleton and its associated motor proteins. There have been some studies exploring the nature of mass flow in filamentous fungi: using mutants defective in motor proteins, utilizing drugs inhibiting microtubules, or injecting silicon oil droplets that will not interact with the motor protein. These demonstrated that cytoplasmic bulk flow does play a major role in displacing organelles along hyphal trunks of *N. crassa* (Lew 2005 and Ramos-García et al, 2009). *N. crassa* has been identified as a preferred model organism for this type of research due to the availability of many mutants allowing the growth of these organisms to be studied in the absence of the genes encoding motor proteins and their associated cytoskeletal components.

Here we utilized live-cell imaging with a confocal laser-scanning microscope to examine the dynamics and distribution of nuclei and mitochondria during mass flow in mature hyphae of *Neurospora crassa*. Nuclei were fluorescently labeled with green fluorescent protein (GFP) histones, and mitochondria were labeled with MitoTracker red dye. Although filamentous mitochondria were positioned both at the hyphal apex and behind, nuclei first appear at a constant distance from the hyphal apical tip creating an area at the tip known as the nuclear exclusion zone (Ramos-García et al, 2009). Based on the cytoplasmic characteristics and organelle location, growing hyphae can be divided into four regions the apical region (which contains the Spitzenkorper) and three sub-apical regions (a zone of nuclear exclusion, nuclei-rich, and vacuole-rich zones) (Ramos-García et al, 2009). The zone that we examined was the nuclei-rich one, about 500 microns behind the growing colony edge of the mycelium. The correlation between the movement of nuclei and the mitochondria was used to distinguish mass flow from the role of molecular motors and the cytoskeleton.

MATERIALS AND METHODS

Strain preparation and media. Stock cultures of a strain with a GFP-tagged histone (rid Pccg-1-hH1+sgfp+, FGSC 10174) were obtained from the Fungal Genetics Stock Center (School of Biological Sciences, University of Missouri, Kansas City, Missouri, USA) (McCluskey, 2003) and maintained on slants of Vogel's Minimal Medium (Vogel, 1956) plus 1.5% (w/v) sucrose and 2.0% (w/v) agar. This strain is used to visualize fluorescently labeled nuclei. The rid allele allows stable crosses (it is RIP defective), Pccg is the ccg promoter, hH1 is the histone gene to which is attached the GFP (sgfp). For experiments imaging nuclei and mitochondria, conidia were spread along one side of a 60 mm (actual 55 mm) Petri dish containing OM plus 1% agar. OM contains [% w/v]: glucose [1], peptone [0.1], yeast extract [0.01], KH_2PO_4 [0.1], and $MgSO_4 \bullet 7H_20$ [0.03]. The peptone and yeast extract were obtained from Difco. The plates were incubated overnight at 28ºC. The next morning, 2.5 ml of OM (without agar) containing MitoTracker Red FM (Invitrogen Molecular Probes, Catalog Number M22425) was poured over the top of the agar so that the fungi grew submerged in the medium to load the dye into the hyphae. The MitoTracker was prepared by dissolving 50 micrograms of the dye in 67 µl of Methanol (2 mM stock). 30 µl of the stock was added to 5 ml of OM (final concentration 12 µM). Then the plates were incubated for 5-6 hours at 28ºC in the dark. Two plates were prepared per imaging experiment.

In initial experiments, the plates were kept at room temperature (22ºC) in the dark. Because of fast growth, the plates were kept at 10ºC in one experiment, at which point we realized that the effectiveness of dye loading into the hyphae is strongly dependent on temperature and is much lower at either 10º or 22º compared to 28º. Therefore, all subsequent experiments pre-incubated the colonies at 28ºC to maximize dye loading.

Dual fluorescent imaging using the Confocal Microscope. In order to explore the nature of mass flow in *Neurospora crassa* hyphae the cytoplasmic movement along the hyphal trunk was studied using dual-flurochrome labeling of the nuclei and the mitochondria. The nuclei (labeled with GFP) were imaged with an excitation wavelength of 488nm and the mitochondria (labeled with MitoTracker Red) were imaged with an excitation wavelength of 579nm. For both, the emission wavelength was 622 nm. To reduce photobleaching, we minimized the exposure time and reduced the laser intensity for both lasers from 25% intensity to a value of 18%. Decreasing the confocal aperture not only reduces the photobleaching but also reduces the fluorescence signal, so the confocal aperture was set at 2.0 as a compromise between minimizing photobleaching and maximizing the signal of the MitoTracker Red. The MitoTracker probe labels mitochondria with a highly polarized membrane potential. The dye intensity was stronger at the hyphal tips, probably due to both the high density of mitochondria at the tips and their highly polarized membrane potentials (Levina and Lew, 2006). Images were collected from hyphal trunks behind the growing edge of the colony. The hyphal trunks were selected on the basis of noticeable cytoplasm movement and strong dye intensity of the mitochondria. Kalman filtering was utilized in order to produce an optimal image from the initial noisy observations.

But in order to reduce the imaging processing time and acquire images with a time-lapse of 2.0 second the number of Kalman filter steps was reduced from 3 to 1.

Time series imaging using the confocal microscope. Scanning was performed using ZEISS EC Plan-Neofluar ×100 oil immersion objective (numerical aperture 1.3). The software program FLUOVIEW was used to control the confocal microscope and acquire the digital images. Smaller regions of interest were selected for imaging. This minimized the time required to scan the image (since the entire field of view was not being scanned). In order to quantify the organelle flow, 120 images were collected using a time-lapse interval of 1.0 second with a zoom value of 2.0. The image stacks were saved in a TIF format for subsequent analysis using ImageJ (Rasband, 2012).

Quantifying the organelle flow using image analysis software. Digital image processing is a technique used in research to enhance and analyze acquired images without distorting the signal or increasing the noise level. The software program ImageJ (Rasband, 2012) was used to geometrically transform (rotate) and digitally enhance (Gaussian filter) the image stacks acquired on the laser confocal microscope.

Gaussian filtering was employed to enhance our ability to track individual mitochondria as they moved along the hypha. This filtering technique uses a two-dimensional Gaussian function to reduce the noise level (Fig. 2, Box 1). The outcome of this enhancement technique was a blurred image with detectable smooth edges.

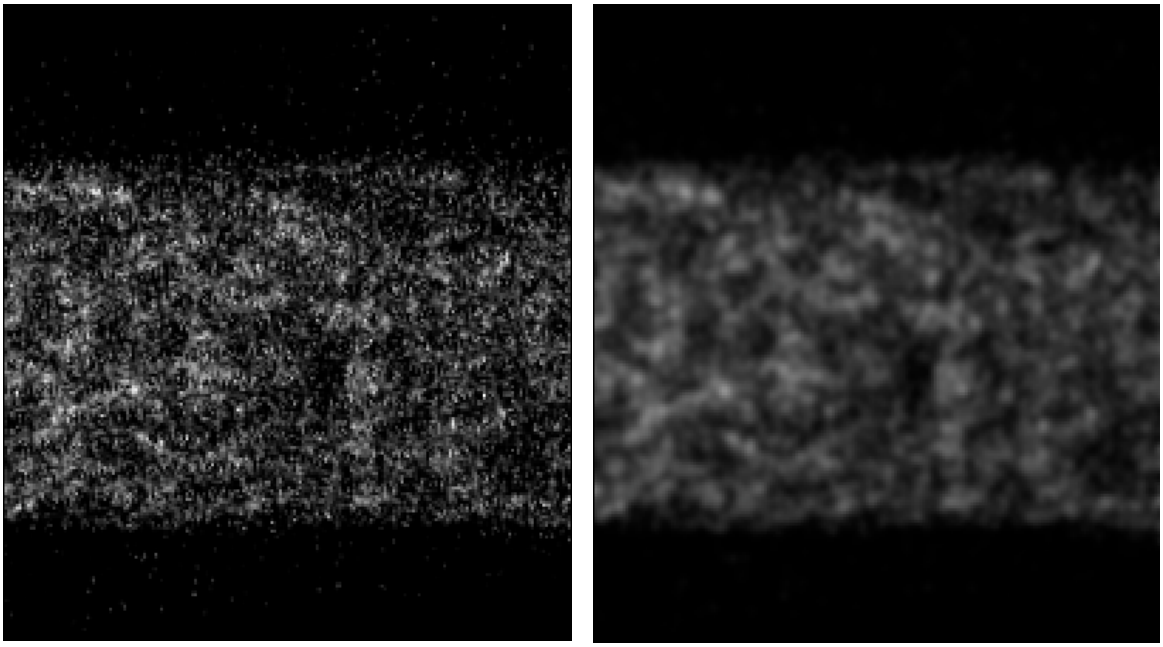

Fig. 2. Screenshot of a Gaussian filter with radius 1.5 pixels applied to the initial digital image obtained using a ×100 objective (left panel). Filtering reduced the noise level and enhanced the ease of tracking individual organelles (right panel).

Box 1 | two dimensional Gaussian function

$$G_\sigma(r) = \frac{1}{2\pi\sigma^2} e^{\frac{-r^2}{2\sigma^2}}$$

Where sigma denotes the width (standard deviation) of the bell-shaped function and r is the distance (radius) from the center. The function is used to weight the filtering of the two dimensional array of pixels. Pixels at the center of the two-dimensional array are weighted greater than those at the periphery. The radius of the array controls the amount of filtering. In ImageJ, this was set to 1.5 pixels, which created a useful image that maximized the ease of tracking individual mitochondria (Fig. 2).

For a given experiment, a sample size of 5–7 mitochondria and nuclei were selected in each digital image. Their movement along the hyphal trunk was tracked using the tracing tool (Fig. 3) and their displacement over a 2.0-second interval (that is, 3 consecutive images taken with 1.0-second intervals) was used to measure the average velocity ($\mu m\ s^{-1}$) and direction using MATLAB.

Data analysis and visualization using MATLAB. In order to calculate the direction of movement and average the velocity for both the mitochondria and nuclei, it was decided to measure the x and the y coordinates for each individual organelle in 2-second intervals. An average of 10 nuclei and 8 mitochondria were selected and their x and y coordinates were determined in 3 consecutive slices (or an interval of 2 seconds). The x and y coordinates were measured using ImageJ; the measurements were then transferred to an Excel (Microsoft) worksheet and processed in MATLAB (Box 2). As shown in Figure 4 the y coordinate for each organelle was used as a representation of the location in the y-axis (relative to the width of the hyphae). For linear regression analysis, the rates of movement and direction for all nuclei and mitochondria at a specific time were averaged, and paired for analysis of correlations.

Box 2 | MATLAB functions

dataGenerator(coordinates.xls, averageVelocity.txt) was used to calculate the average velocity vector for the particles of each of the two organelles. The output was the x and y coordinates of the initial and final points for the average velocity vectors in columns 1-4 and the time in column 5 (the code is provided in Appendix I).

graphGenerator(averageVelocityNuclei.txt, averageVelocityMitochondria.txt) was used to map the cytoplasmic flow drawing the average velocity vectors. The length of these vectors was an indication of the magnitude of the velocity for each organelle in the cytoplasm and the arrowheads show the directionality (Appendix I).

scatterGraph(averageVelocityNuclei.txt, averageVelocityMitochondria.txt) was used to graph the scatter plot for the average velocity of the nuclei compared to the mitochondria and also the scatter plot for the direction of movement of the nuclei compared to the mitochondria (Appendix I).

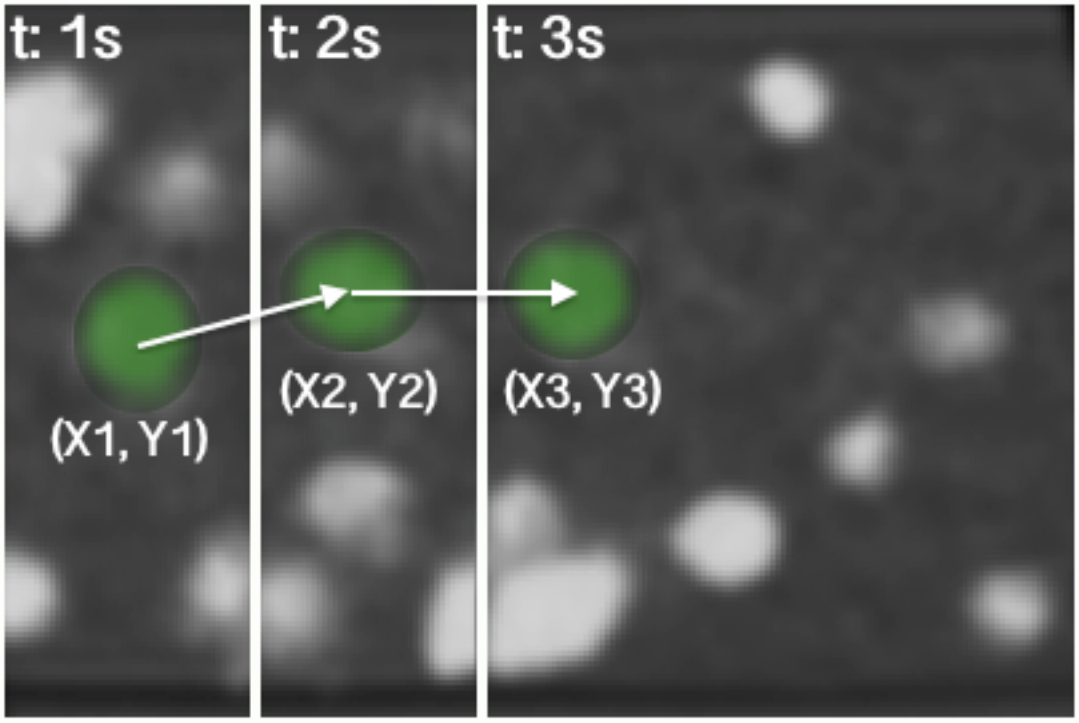

Fig. 3. Example of nuclei movement along the hyphal trunk. Nuclei were labeled using the GFP fluorescent dye (excitation wavelength of 488nm) and appeared green under the confocal microscope. Their movement was tracked using the tracing tool within the ImageJ software program and the measurements were used to perform the average velocity calculations.

Average velocity was calculated by first measuring the displacement in the x and y direction separately (see Fig. 3 for x, y coordinate measurements):

$$\Delta x = (x_3 - x_2) + (x_2 - x_1) \text{ and } \Delta y = (y_3 - y_2) + (y_2 - y_1).$$

Knowing the total displacement in the x and y direction, the average displacement in each direction was calculated, notice that having a time-lapse of 1.0 second the average displacement in each direction is the same as the average velocity on that direction, the value of 2 in the denominator corresponds to both the number of measurements (2 consecutive movement of the organelle) and the time interval of 2.0 seconds:

$$\Delta x_{avg} = \Delta v_x = \frac{\Delta x}{2} \text{ and } \Delta y_{avg} = \Delta v_y = \frac{\Delta y}{2}.$$

It was then decided to store the average velocity measurements in terms of the coordinates of the initial and final points instead of the more common representation of using the magnitude and direction.

To map the cytoplasmic movement, organelles were represented by their average velocity vectors in terms of their location in the hyphal width (y-axis) and time (x-

axis). Velocity vectors are centered at the mid point (t_2) between the initial and final time ($\Delta t = t_3 - t_1$) (Fig. 4).

Velocity calculations were performed in the μm s^{-1} using a scale of 7.478 pixels.μm^{-1}. (based on micrometer scale calibrations for the ×100 objective with ×2 zoom).

RESULTS

To study the nature of flow in the *Neurospora crassa*, the hyphae showing cytoplasmic movement were identified and tracked towards the tip to obtain the highest fluorescent intensity for the mitochondria (nuclei were consistently very bright). Microscopic observations of the hyphae revealed that the movement of the uniformly distributed nuclei and mitochondria was always towards the tip.

This cytoplasmic flow was mapped as the movement velocities for the nuclei and the mitochondria. An example of the velocity vectors showing the magnitude and the direction of movement for an average of 10 nuclei and 8 mitochondria over time are shown in Figure 4.

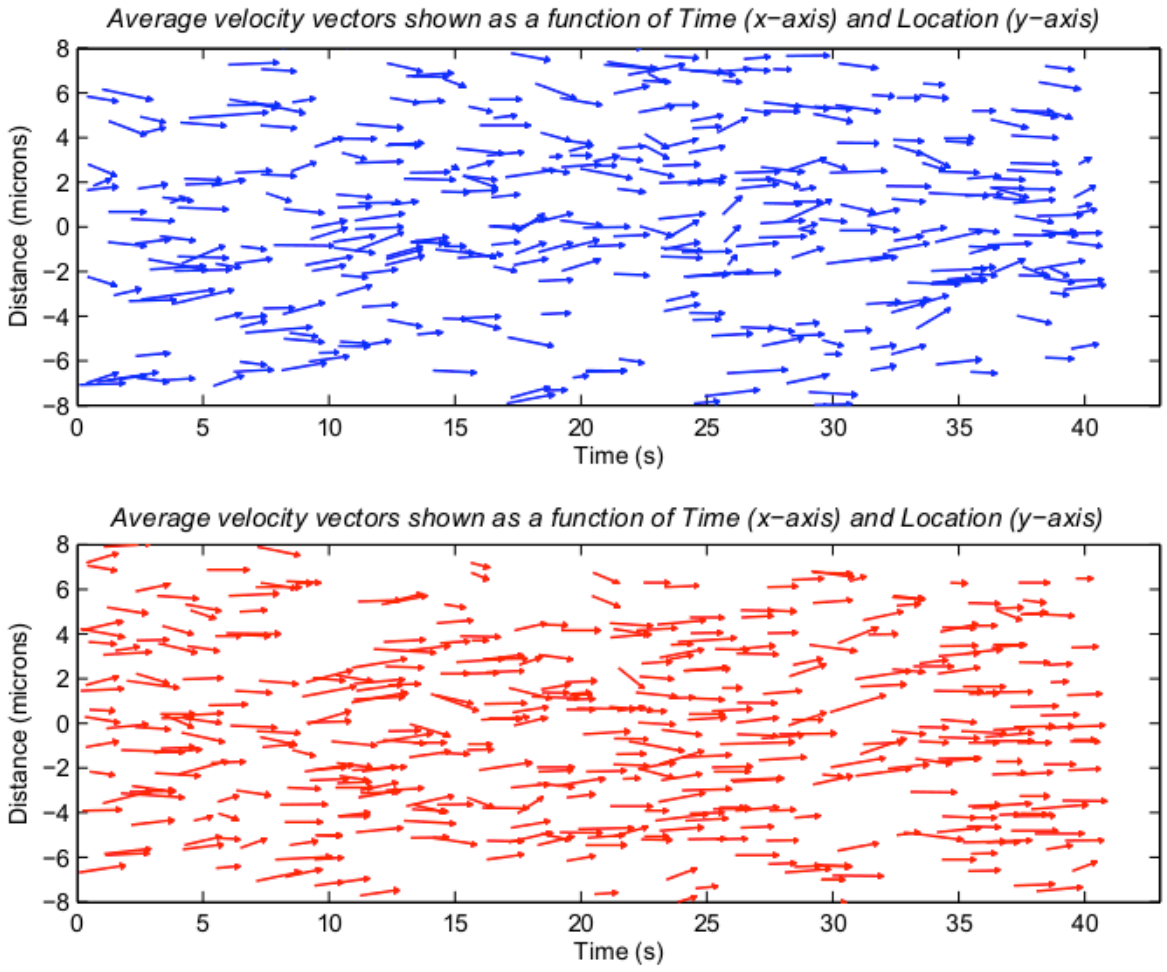

Fig. 4. The cytoplasmic map for the movement of the nuclei (top graph) and the mitochondria (bottom graph) is shown. Y-axis represents the width of the hyphae and the x-axis represents time. The length of the arrows is representative of the rate of motility and the arrowheads represent the direction of movement.

Mitochondria were observed as filamentous convoluted structures under the confocal microscope, they were uniformly distributed in all 4 regions of the hyphae. They appeared brighter near the tip suggesting tip-localized mitochondria maintain a relatively higher membrane potential (Levina and Lew, 2006). Although filamentous in structure mitochondria more often appear vermiform in structure, as shown by fluorescence and electron transmission microscopy (Fig. 5). This made visualization more challenging compared to nuclei, which exhibited well-defined circular to oblong shapes.

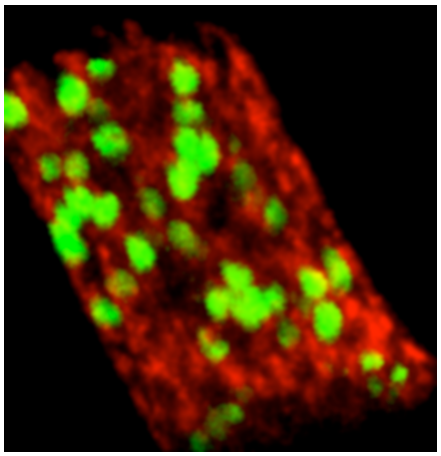

Fig. 5a. A three-dimensional reconstruction from z-sectioned hyphae with fluorescing nuclei (green) and mirochondria (red). Note the spherical to oblate nuclei and vermiform to filamentous shapes of the mitochondria.

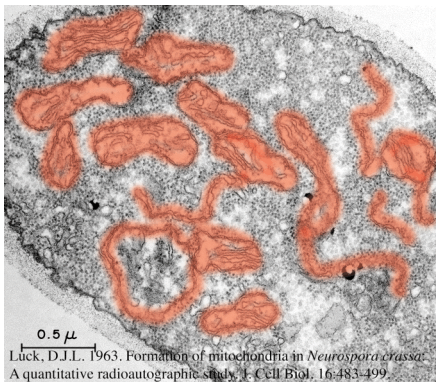

Fig. 5b. Electron transmission microscopy of a *Neurospora* hyphae in cross-section showing that the mitochondria are present both as oblate structures and more vermiform to filamentous shapes that correspond reasonably well to the obtained fluorescence images of mitochondria in hyphal trunks. Putative mitochondria are highlighted in red. The original image was published in Luck (1963).

The relationship between the rate of nuclear movement and mitochondria movement is shown in Figure 6. To confirm that nuclei traveled toward the hyphal apex at essentially the same speed as the mitochondria, a linear regression analysis was performed. Line of best fit was drawn for the complete data sets having a slope of 0.85917 (R = 0.91053). This is slightly lower than the expected slope of 1.0 —if nuclei and mitochondria move at the same velocity due to mass flow— but this difference is unlikely to be significant because average velocities are similar, and not statistically different on the basis of a 2-tail t-test (P = 0.1538) (Table 1).

Fig. 6. Linear regression analysis of the rate of nuclei movement and the rate of mitochondria movement in the hyphae of *Neurospora crassa*. Each point in the plot represents the correlation between the velocity (μm s^{-1}) of the nuclei and the mitochondria (μm s^{-1}). All data points are shown and the regression was calculated for the entire set of data.

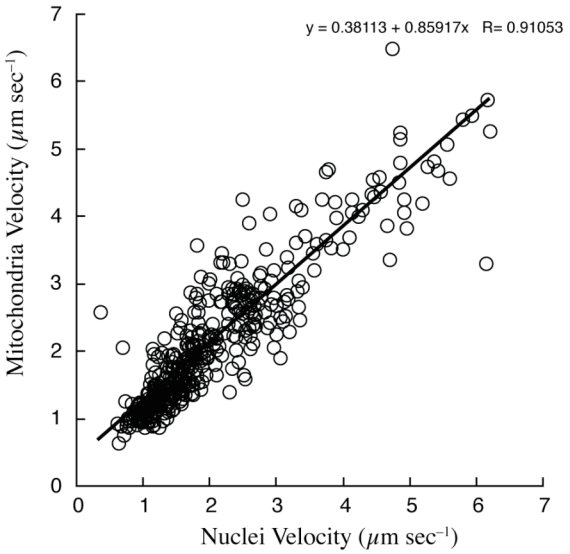

Table 1. The velocity of organelle movement. The means and standard deviations are tabulated for nuclei and mitochondria (in μm s^{-1}), as is the 2-tail t-test.

N. *Crassa* strain	Mean	SD	N	T-test
Nuclei	1.9737	1.1028	440	0.1538
Mitochondria	2.0769	1.0406	440	

It was evident that the nuclei and the mitochondria essentially moved in the same direction along the hypha, towards the growing colony edge as shown in Figure 7.

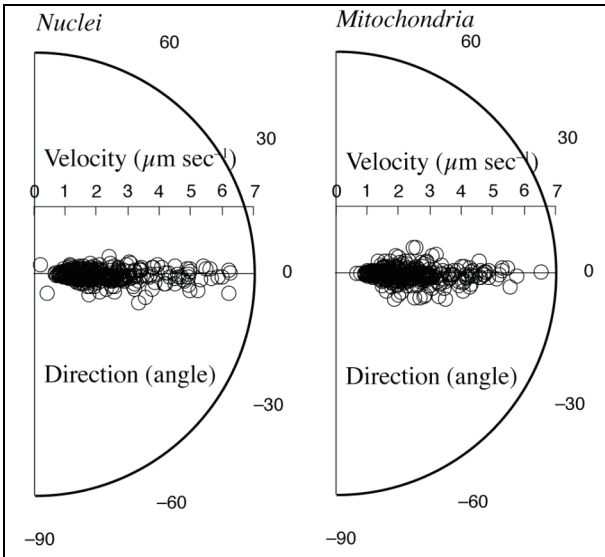

Fig. 7a: Polar plots of nuclei and mitochondria velocity and direction and mitochondria. Nuclei are shown in the left panel, mitochondria in the right panel.

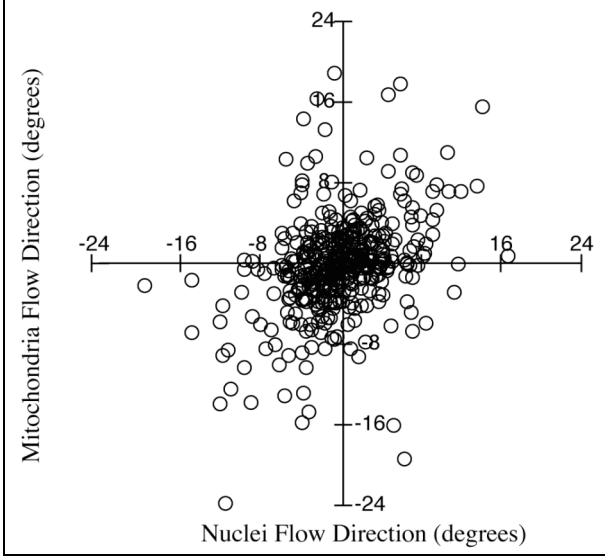

Fig. 7b: Scatter plot analysis for the directionality of the movement of the nuclei and the mitochondria. Each point plots the directionality in degrees for the mitochondria and the nuclei in a given time frame. A few outliers (values above 24° and below -24°) have been excluded from the dataset. Note that the date essentially clusters at 0 degrees, indicating that the directionality is essentially the same for both organelles.

Table 2. The direction of organelle movement. The means and standard deviations are tabulated for nuclei and mitochondria (in degrees), as is the 2-tail t-test.

N. *Crassa* strain	Mean	SD	N	T-test
Nuclei	1.9737	1.1028	440	0.8100
Mitochondria	2.0769	1.0406	440	

DISCUSSION

In order to study the nature of mass flow it was decided to track the movements of two distinct organelles (nuclei and mitochondria). Correlated organelle movements —both velocity and direction— provide evidentiary support for the primary role of mass flow in movements of cytoplasm in the trunk hyphae. It's important to note that movement of nuclei in a direction opposite (retrograde) to that of the normal tip-directed movement was occasionally observed, consistent with some contribution by a cytoskeleton/motor system, but was very rare.

Although green fluorescently labeled nuclei and red labeled mitochondria have very different morphologies and functions, they both moved in the same direction towards the growing hyphae. The major orientation was parallel to the growing axis of the hyphae. With respect to velocity, the slope of the linear regression of mitochondria velocity *versus* nuclei velocity was 0.859, that is, less than the expected slope of 1. Whether this is a real difference is unclear, because the statistical tests indicate that overall velocities are statistically the same for the two organelles. One possible reason for slower movement of the mitochondria is their more filamentous and interconnected structure. This could result in significant 'drag', slowing the movement of mitochondria compared to the independent and more or less spherical nuclei.

It was hypothesized that if cytoplasmic bulk flow was in fact responsible for the movement of the nuclei and other organelles towards the growing edge of the colony, then a correlation between the velocities of nuclei and mitochondria should be evident. Confocal microscopy and dual fluorescent labeling allowed us to map this by tracking the movement of individual organelles in 2-second intervals for 120 seconds. We were able to construct vector maps of this flow in selected regions of the hyphae over time (Fig. 4). The movement of an average of 10 nuclei and 8 mitochondria were tracked at every time interval for a total of 40 2-second intervals in 11 data sets. Each point plotted in Figure 6 represents the correlation between the speed of the nuclei and the mitochondria in the same region of the hypha, from the center to the wall. All data points are shown, the correlation is clear. Mass flow dominates organelle movements *Neurospora crassa* hyphae.

REFERENCES

Diacumakos, E. G., Garnjobst, L., Tatum, E. L. (1965). A cytoplasmic character in *Neurospora crassa*. The role of nuclei and the mitochondria. *J. Cell Biol.* **26:**247–243.

McCluskey, K. (2003). The Fungal Genetics Stock Center: from molds to molecules. *Adv Appl Microbiol* **52:**245–262.

Levina, N. N., Lew, R. R. (2006). The role of tip-localized mitochondria in hyphal growth. *Fungal Genet. Biol.* **43:**65-74.

Lew, R. R. (2005). Mass flow and pressure-driven hyphal extension in *Neurospora crassa*. Microbiology **151:**2685-2692.

Lew, R. R. (2011). How does a hypha grow? The biophysics of pressurized growth in fungi. Nature Reviews Microbiology. **9:**509-518.

Mourino-Perez, R. R., Roberson, R. W., and Bartnicki-Garcia, S. (2006). Microtubule dynamics and organization during hyphal growth and branching in *Neurospora crassa*. Fungal Genet. Biol. **43:**389-400.

Ramos-García, S. L., Roberson, R. W., Freitag, M., Bartnicki-García, S., Mouriño-Pérez, R. R. (2009). Cytoplasmic bulk flow propels nuclei in mature hyphae of *Neurospora* crassa. Eukaryot Cell. **8:**1880–1890.

Rasband, W. S. ImageJ, U. S. National Institutes of Health, Bethesda, Maryland, USA, http://imagej.nih.gov/ij/, 1997-2012.

Luck, D.J.L. (1963). Formation of mitochondria in *Neurospora crassa*: A quantitative radioautographic study. J. Cell Biol. **16:**483-499.

APPENDIX I — MATLAB FILES FOR DATA ANALYSIS

DATAGENERATOR.M

```
function data = dataGenerator(fileIn, fileOut)

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%                                                                               %
% September 13, 2012 - Version1.2                                               %
% Dr. Roger R. Lew Laboratory                                                   %
%                                                                               %
% Uses 3 entities in each increment through the input data to calculate         %
% the avegrae velocity for a 2 second interval.                                 %
%                                                                               %
% INPUT: Data are input to the function as an excel file, ".xsl" format:        %
% >> dataGenerator('Input file name', 'output file name')                       %
%                                                                               %
% OUTPUT: resultant velocities are saved in 3 columns, as a ".txt" file         %
% in the same directory.                                                        %
%                                                                               %
% Column 1: represents the x coordinate                                         %
% Column 2: represents the y coordinate                                         %
% Column 3: represents the mid-time                                             %
%                                                                               %
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

clc


% Reads an Excel file, table entries are copied into the matrix "data"
data = xlsread (fileIn);


% Total number of entities in the excel file
dataSize = size(data);
bigCount = dataSize(1);

% Allocating space for the calculated average velocities, dataSize/3,
% because 1 average velocity is calculated for every 3 entities.
finalV = zeros(dataSize/3,5);

maxValue = data(1,2);
minValue = data(1,2);

% Incrementing 1 units, starting from 2 to find the value for minValue and
% maxValue variables equivalent of the largest and the smallest
% y-coordinates in the data set, these values are used to calculate the
% width of the hyphae [Hyphae Width = maxValue - minValue]

for icounter = 2 : 1 : bigCount

    if data(icounter, 2) > maxValue
    maxValue = data(icounter, 2);
    end
    if data(icounter, 2) < minValue
    minValue = data(icounter, 2);
    end


end
```

```
x = 1;
y = 1;

% Incrementing 3 units, starting from 1 until bigCount is reached
for icounter = 1 : 3 : bigCount

        % vector contains all the entities for values icounter, icounter+1,
        % and icounter+2, basically the three location coordinates for an
        % organelle in a 2-second interval
        vector = (data(icounter:1:icounter+2, :));

        t = (data (icounter, 3));

        % v1, v2, v3, contain the x-coordinate, y-coordinate, and time for
        % the 3 entities present in vector
        v1 = (vector (1,:));
        v2 = (vector (2,:));
        v3 = (vector (3,:));


        % Every single entity in v(n) is subtracted from its coreesponding
        % value in v(n-1)
        iv = v2 - v1;
        iiv = v3 - v2;

        % To determine the origin of the graph, basically half Y values are
        % above and half are bellow this number.
        origin = (maxValue + minValue)/2;

        % x0
        finalV (x,y) = v1(1);
        y = y + 1;
        % y0
        finalV (x,y) = v1(2) - origin;
        y = y + 1;
        % x1
        finalV (x,y) = v1(1) + (iv(1) + iiv(1))/2;
        y = y + 1;
        % y1
        finalV (x,y) = (v1(2) - origin) + (iv(2) + iiv(2))/2;
        y = y + 1;

        % t - UPDATED091312
        finalV (x,y) = (2*t)+(t-1);

        x = x + 1;
        y = 1;

end

% All data will be written in a .txt file with the name declared by the
% user, as the second entry of the function.
% >> dataGenerator ('input_file_name','output_file_name')
% finalV =
% "x0-coordinate y0-coordinate x1-coordinate y1-coordinate t"
dlmwrite([fileOut, '.txt'], finalV, 'delimiter', '\t', 'newline', 'unix');
end
```

GRAPHGENERATOR.M

```
function data = graphGenerator(fileOneIn, fileTwoIn)

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%                                                                             %
% September 13, 2012 - Version1.2                                             %
% Dr. Roger R. Lew Laboratory                                                 %
%                                                                             %
% Uses the text file generated by dataGenerator.m as input, takes in two      %
% text files for the two organelles and maps the organelle flow               %
%                                                                             %
% INPUT: Uses the text files generated by dataGenerator.m as input            %
% entities.                                                                   %
% >> graphGenerator('Input_file_name_1', 'Input_file_name_2')                 %
%                                                                             %
% OUTPUT: resultant velocity vectors are graphed representing the             %
% direction and the speed of the movement by vectors length and arrow head%
% directionality.                                                             %
%                                                                             %
% Velocity vectors in the graph of hyphae width versus time                   %
%                                                                             %
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

clc

% The input, fileOneIn is simply the output generated by dataGenerator.m
% function - Nuclei
data = dlmread ([fileOneIn '.txt']);
dataSize = size(data);

% Total number of entities in the input text files
bigCount = dataSize(1);

% Creating a subplot, to show the map of the flow for both the Nuclei and
% the Mitochondria
subplot(2,1,1);

% Top Figure - The Nuclei
for icounter = 1 : 1 : bigCount


        % UPDATED091312
        x0 = data(icounter, 5) - (data(icounter, 3) - data(icounter, 1))/2;
        x1 = data(icounter, 5) + (data(icounter, 3) - data(icounter, 1))/2;

        y0 = data(icounter, 2);
        y1 = data(icounter, 4);


        % To use plot_arrow.m - Erik Johnson:
        % http://www.mathworks.com/matlabcentral/fileexchange/278-arrow-m/c
        % ontent/arrow.m
        plot_arrow( x0,y0,x1,y1,'headwidth',0.25,'headheight',0.2, 'color',
        'b', 'edgecolor', 'b', 'facecolor', 'b');
        hold on;


end

hold off;
```

```
% The input, fileTwoIn is simply the output generated by dataGenerator.m
% function - Mitochondria
data = dlmread ([fileTwoIn '.txt']);
dataSize = size(data);

% incrementing every 3 entries and calculate the average velocity and the
% directionality angle

bigCount = dataSize(1);

% Bottom Figure - The Mitochondria
subplot(2,1,2);

for icounter = 1 : 1 : bigCount


        % UPDATED091312
        x0 = data(icounter, 5) - (data(icounter, 3) - data(icounter, 1))/2;
        x1 = data(icounter, 5) + (data(icounter, 3) - data(icounter, 1))/2;

        y0 = data(icounter, 2);
        y1 = data(icounter, 4);


        % To use plot_arrow.m - Erik Johnson:
        % http://www.mathworks.com/matlabcentral/fileexchange/278-arrow-m/c
        % ontent/arrow.m
        plot_arrow( x0,y0,x1,y1,'headwidth',0.25,'headheight',0.2, 'color',
        'r', 'edgecolor', 'r', 'facecolor', 'r');
        hold on;

end

hold off;

end
```

SCATTERGRAPH.M

```
function data = scatterGraph (dataSetOne, dataSetTwo)

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%                                                                              %
% May 11, 2012 - Version1.0                                                    %
% Dr. Roger R. Lew Laboratory                                                  %
%                                                                              %
% The scatter graph for the directionality and the velocity of the Nuclei      %
% and the Mitochondria is generated. In addition, text files containing        %
% the entities presented on the sctter plots' axis are generated in the        %
% same directory. In calculating the velocity and direction at every           %
% instant of time, the average for all the entries at that instant of time%
% is calculated.                                                               %
%                                                                              %
% INPUT: Text files generated by dataGenerator.m are used as the input         %
% >> scatterGraph('Input_file_name_Nuc','Input_file_name_Mito')                %
%                                                                              %
% OUTPUT: 3 scatter plots and 6 text files are generated as outputs in the%
% same directory. The text files represent the data on each axis of the 3 %
% scatter plots                                                                %
%                                                                              %
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

clc

% Input the data entries from data.txt into matrix "data"
dataOne = dlmread ([dataSetOne '.txt']);

% Total number of entries in the data set.
dataSize = size(dataOne);
bigCount = dataSize(1);


% Every 3 slices in a set of 120 slices, representing 120 seconds, are used
% thus 40 entries for each scatter plot are used (120/3)
timeMax = 40;

% Allocating memory, a matrix containing zeros, for the 40 entities.
avgVectorN = zeros(timeMax,2);


% This variable will be changes in the following for loop to each instant
% of time, and is used as an indication for average data having similar
% timeMemoryVar.
timeMemoryVar = 2;

whileCounter = 0;

% Total displacement in the x direction at time: timeMemoryVar
xDisplacement = 0;

% Total displacement in the y direction at time: timeMemoryVar
yDisplacement = 0;

rowCounter = 1;


for icounter = 1 : 1 : bigCount

    % Calculates the total displacement in the x and y direction for
    % entities at t = timeMemeoryVar.
    if dataOne(icounter, 5) == timeMemoryVar;
```

```
        xDisplacement = xDisplacement + ( dataOne(icounter, 3) - dataOne
        (icounter, 1) );
        yDisplacement = yDisplacement + ( dataOne(icounter, 4) - dataOne
        (icounter, 2) );
        whileCounter = whileCounter + 1;
    end

    if dataOne(icounter, 5) ~= 119;
        if timeMemoryVar ~= dataOne(icounter+1, 5)

            % Calculates the average displacement in the x and y direction
            % for entities at t = timeMemoryVar by dividing the total
            % displacement in each direction by number of entities at each
            % instance
            avgVectorN (rowCounter, :) = [xDisplacement/whileCounter
                yDisplacement/whileCounter];

            % Each time instance is 3 seconds apart from the other
            timeMemoryVar = timeMemoryVar + 3;
            whileCounter = 0;
            xDisplacement = 0;
            yDisplacement = 0;
            rowCounter = rowCounter + 1;

        end
    end
    if icounter == bigCount;
        avgVectorN (rowCounter, :) = [xDisplacement/whileCounter
            yDisplacement/whileCounter];
    end



end


dataTwo = dlmread ([dataSetTwo '.txt']);
dataSizeM = size(dataTwo);
bigCount = dataSizeM(1);


avgVectorM = zeros(timeMax,2);

timeMemoryVar = 2;
whileCounter = 0;
xDisplacement = 0;
yDisplacement = 0;

rowCounter = 1;

for icounter = 1 : 1 : bigCount

    % Calculates the total displacement in the x and y direction for
    % entities at t = timeMemeoryVar.
    if dataTwo(icounter, 5) == timeMemoryVar;

        xDisplacement = xDisplacement + ( dataTwo(icounter, 3) -
        dataTwo(icounter, 1) );
        yDisplacement = yDisplacement + ( dataTwo(icounter, 4) -
        dataTwo(icounter, 2) );
        whileCounter = whileCounter + 1;
    end

    if dataTwo(icounter, 5) ~= 119
```

```
        if timeMemoryVar ~= dataTwo(icounter+1, 5)

            % Calculates the average displacement in the x and y direction
            % for entities at t = timeMemoryVar by dividing the total
            % displacement in each direction by number of entities at each
            % instance
            avgVectorM (rowCounter, :) = [xDisplacement/whileCounter
                yDisplacement/whileCounter];

            % Each time instance is 3 seconds apart from the other
            timeMemoryVar = timeMemoryVar + 3;
            rowCounter = rowCounter + 1;
            whileCounter = 0;
            xDisplacement = 0;
            yDisplacement = 0;

        end
    end
    if icounter == bigCount
        avgVectorM (rowCounter, :) = [xDisplacement/whileCounter
            yDisplacement/whileCounter];
    end


end

% Creates a text file containing the average displacement at each direction
% and for each organelle, in 2 columns containing the average displacement
% in the x direction and y direction.

% Two seperate text files are generated one for the Mitochondria and one
% for the Neuclei
dlmwrite(['avgDisplacementMito', '.txt'], avgVectorM, 'delimiter', '\t',
'newline', 'unix');
dlmwrite(['avgDisplacementNucl', '.txt'], avgVectorN, 'delimiter', '\t',
'newline', 'unix');

velocityNucl = zeros(timeMax,1);
velocityMito = zeros(timeMax,1);

directionNucl = zeros(timeMax,1);
directionMito = zeros(timeMax,1);

for icounter = 1 : 1 : timeMax

        deltaYN = avgVectorN(icounter,2);

        deltaXN = avgVectorN(icounter,1);

        % Representing the magnitude of the velocity
        velocityNucl(icounter, 1) = sqrt( deltaXN^2 + deltaYN^2);
        % Representing the angle - direction of movement
        directionNucl (icounter, 1) = atand(deltaYN/deltaXN);


        deltaYM = avgVectorM(icounter,2);

        deltaXM = avgVectorM(icounter,1);

        % Representing the magnitude of the velocity
        velocityMito(icounter, 1) = sqrt(deltaXM^2 + deltaYM^2);
        % Representing the angle - direction of movement
        directionMito (icounter, 1) = atand(deltaYM/deltaXM);
```

```
end

% Generating the text files containing the magnitude of the average
% velocity for the Mitochondria and the Nuclei
dlmwrite(['speedMito', '.txt'], velocityMito, 'delimiter', '\t', 'newline',
'unix');
dlmwrite(['speedNucl', '.txt'], velocityNucl, 'delimiter', '\t', 'newline',
'unix');
dlmwrite(['Nuc-Mito(speed)', '.txt'], velocityNucl-velocityMito,
'delimiter', '\t', 'newline', 'unix');

% Generating the scatter plot showing the magnitude of the average
% velocity for the Nuclei against the Mitochondria
scatter(velocityNucl,velocityMito,5,[.5 0 0],'filled');
xlabel('Nuclei Speed (microns/second)')
ylabel('Mitochondria Speed (microns/second)')

% Generating the scatter plot containing the magnitude of the average
% velocity for the Nuclei against the difference between the Nuclei and the
% Mitochondria
figure
scatter(velocityNucl,velocityNucl-velocityMito,5,[.5 0 0],'filled');
xlabel('Nuclei (microns/second)')
ylabel('Difference (vNuclei - vMitochondria) (microns/second)')

% Generating the text files containing the directionality angle in degree
% for the movement of the the Nuclei and the Mitochondria
dlmwrite(['directionMito', '.txt'], directionMito, 'delimiter', '\t',
'newline', 'unix');
dlmwrite(['directionNucl', '.txt'], directionNucl, 'delimiter', '\t',
'newline', 'unix');

% Generating the scatter plot showing the directionality angle in degree
% for the movement of the the Nuclei against the Mitochondria
figure
scatter(directionNucl,directionMito,5,[.5 0 0],'filled');
xlabel('Nuclei (degree)')
ylabel('Mitochondria (degree)')


end
```