Supplemental Figures

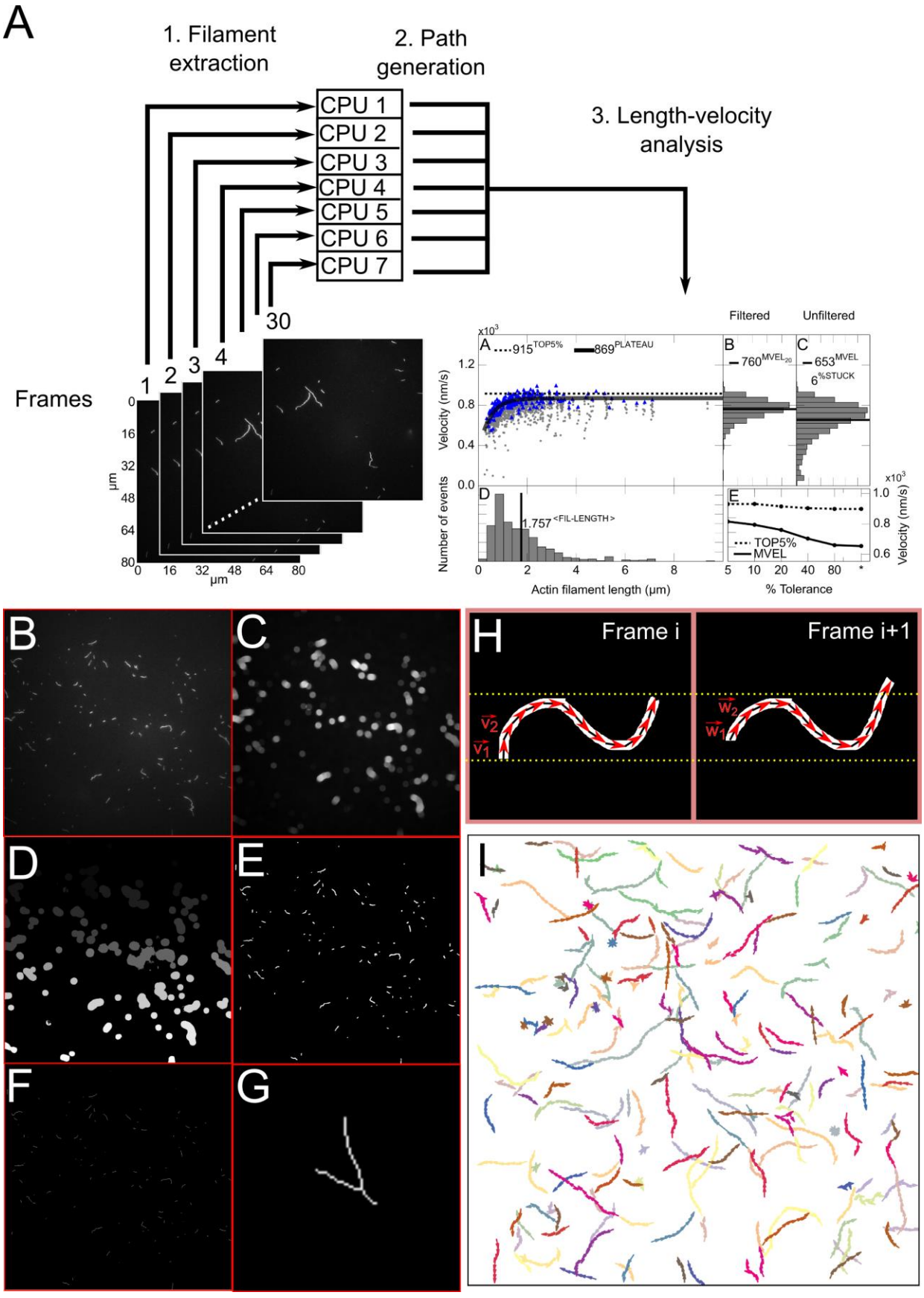

A
1. Filament extraction
2. Path generation
3. Length-velocity analysis
CPU 1
CPU 2
CPU 3
CPU 4
CPU 5
CPU 6
CPU 7
Frames
1
2
3
4
30
µm
Filtered
Unfiltered
Velocity (nm/s)
Number of events
Actin filament length (µm)
% Tolerance
915 TOP5%
869 PLATEAU
760 MVEL20
653 MVEL
6 %STUCK
1.757 <FIL-LENGTH>
TOP5%
MVEL
B
C
D
E
F
G
H
Frame i
Frame i+1
I

Figure S1, Related to Figure1. Graphical description of FAST's actin filament tracking algorithm. A. Fast performs the analysis of in vitro motility movies in 3 steps. The first step is the filament extraction from each frame in stack. Filament extraction for individual frames is performed in parallel on multiple CPUs. The second step involves the comparison of filament traces between adjacent frames to build actin-gliding paths. The final step involves the analysis of filament length-velocity data along the paths built in the second step. **(B, C, D, E, F, and G)** Extraction of filaments from individual frames. **B.** Raw image **C.** Background subtracted image **D.** Detection of areas containing filaments **E.** Separation of individual filaments **F.** Skeletonized filaments **G.** Two crossing filaments. **(H and I)** Path generation for the filaments. **H.** The same filament in two adjacent frames i and i+1. Filaments are represented as linearly connected vectors. Vectors are defined as $\vec{v}_j$ and $\vec{w}_j$ ($j \in \{1,2,3,\ldots,N\}$) for the filaments in frame i and i+1 respectively. Yellow dashed lines mark the tips of the filament in frame i. **I.** Paths constructed for the filaments in a movie.

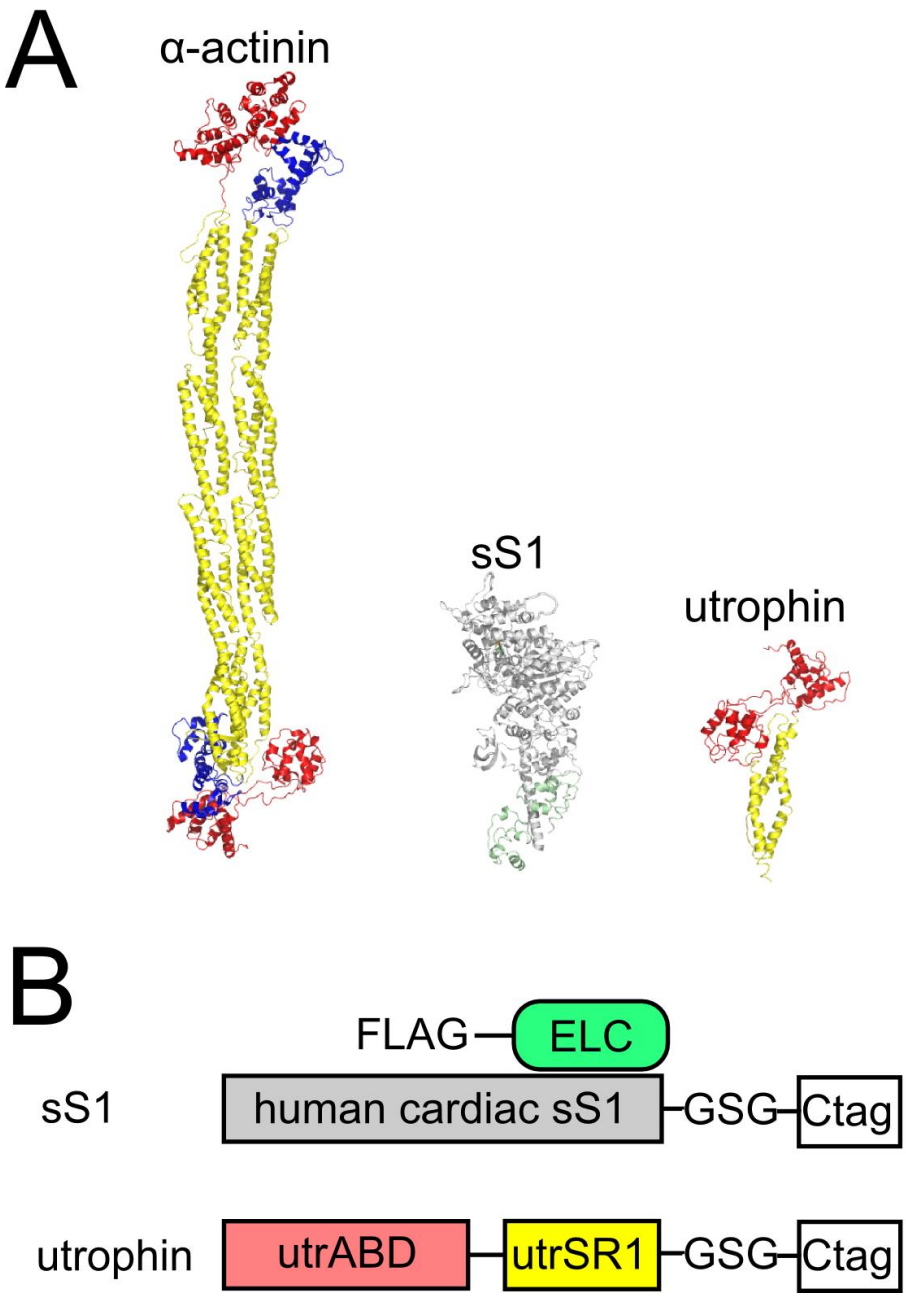


Figure S2, Related to Figure 2. Actin binding proteins and human cardiac sS1 designed for loaded in vitro motility. A. The model for utrophin construct structure in comparison to sS1 and alpha actinin models. The utrophin model is built from the structural coordinates of the actin binding domain and N-terminal first spectrin repeat of alpha actinin (pdb id: 1sjj). The sS1 model is based on the human cardiac sS1 structure (pdb id: 4db1). Structural coordinates of the missing residues in the sS1 structure are modelled from the chicken skeletal S1 structure (pdb id: 2mys). Actin binding domain of utrophin and alpha actinin are in red; spectrin repeats are in yellow; C-terminal EF finger motifs of alpha actinin are in blue; sS1 is in gray (ELC in green). Alpha actinin and utrophin

domain definitions are based on the structures of utrophin's actin binding domain (pdb id:1qag) and the first N-terminal spectrin repeat (pdb id: 3uul). **B.** The composition of the constructs designed for loaded in vitro motility. All constructs including sS1 have a C-terminal tag (Ctag) of eight residues (RGSIDTWV) that binds specifically to the SNAP-PDZ18 attachment system (Huang et al., 2009). A GSG flexible linker separates the N-terminal portion of the constructs from the Ctag. The N-terminal portion of the utrophin construct is composed of the mouse utrophin actin-binding domain (utrABD) and the first N-terminal spectrin repeat (utrSR1), which together corresponds to the first 416 residues of mouse utrophin. Utrophin construct has an N-terminal 6X Histag for affinity purification via a nickel column. The N-terminal portion of human beta cardiac sS1 is composed of the first 808 residues of human beta cardiac myosin (gene id: MYH7), which forms a complex with human essential light chain (ELC). The N-terminal Flag-tag of ELC is utilized for affinity purification of sS1. The human alpha cardiac sS1 construct is composed of the first 810 residues of the human alpha cardiac myosin (gene id: MYH6). The sS1 constructs for ATPase measurements have an eGFP tag in place of the Ctag. The eGFP emission is used for accurate sS1 concentration determination, which is critical for the accurate measurement of ATPase rates.

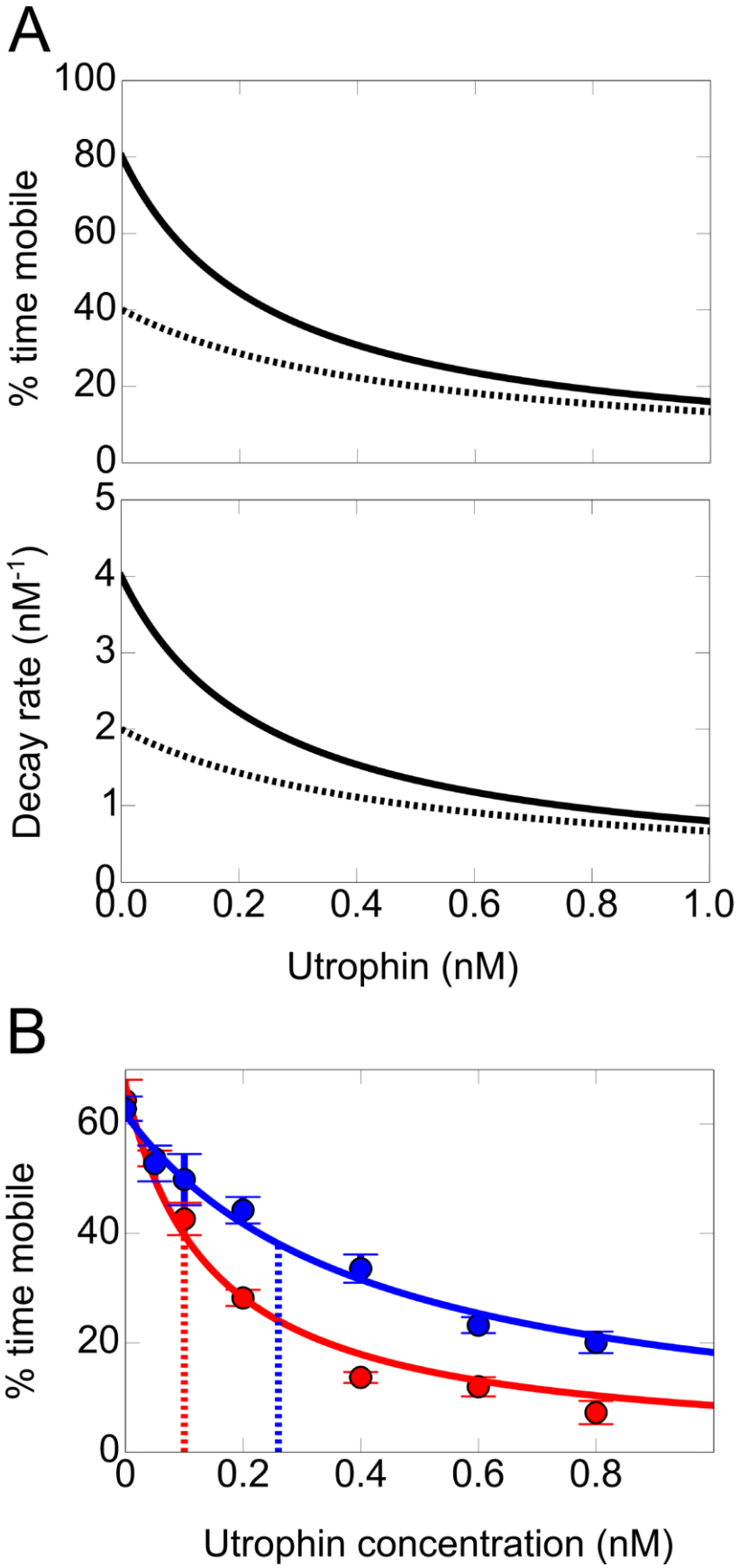
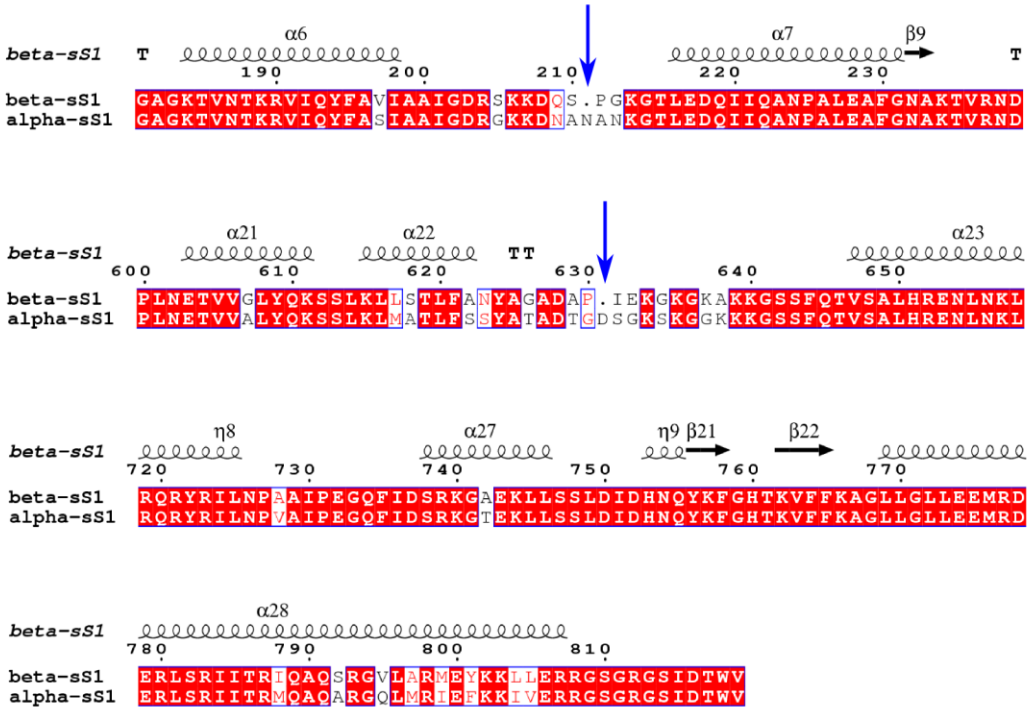

A
100
80
60
40
20
0
% time mobile
5
4
3
2
1
0
Decay rate (nM-1)
0.0
0.2
0.4
0.6
0.8
1.0
Utrophin (nM)
B
60
40
20
0
% time mobile
0
0.2
0.4
0.6
0.8
Utrophin concentration (nM)

Figure S3, Related to Figure 3. Stop model decay rate as a function of utrophin concentration and sS1 concentration dependence of loaded motility data. A. Simulated percent time mobile (upper) and percent time mobile decay rate (lower) as a function of utrophin concentration from the Stop Model. Solid and dashed lines are for two myosins with high and low unloaded percent time mobile respectively; but with the same K_S in the Stop Model (see Eq. 1). The decay rate vs. utrophin profile is linearly proportional to the percent time mobile vs. utrophin profile (see Supplemental Experimental Procedures). Myosins with higher unloaded percent time mobile values have higher apparent decay rates with utrophin as predicted by the Stop Model. **B.** Representative loaded in vitro motility data for beta cardiac sS1 at 0.02 (red) and 0.03 (blue) mg/ml motor concentrations. Lines show the best Stop Model fit to unfiltered percent time mobile loaded motility data. Dashed lines indicate the K_S determined from the fit. Confidence intervals are the standard error of mean from at least four movie replicates.

A


beta-sS1 T α6 190 200 210 α7 β9 220 230 T
beta-sS1 GAGKTVNTKRVIQYFAVIAAIGDRSKKDQS.PGKGTLEDQIIQANPALEAFGNAKTVRND
alpha-sS1 GAGKTVNTKRVIQYFASIAAIGDRGKKDNANANKGTLEDQIIQANPALEAFGNAKTVRND
beta-sS1 α21 α22 TT 600 610 620 630 640 α23 650
beta-sS1 PLNETVVGLYQKSSLKLLSTLFANYAGADAP.IEKGKGKAKKGSSFQTVSALHRENLNKL
alpha-sS1 PLNETVVALYQKSSLKLMATLFSSYATADTGDSGKSKGGKKKGSSFQTVSALHRENLNKL
beta-sS1 η8 720 730 α27 740 750 η9 β21 760 β22 770
beta-sS1 RQRYRILNPAAIPEGQFIDSRKGAEKLLSSLDIDHNQYKFGHTKVFFKAGLLGLLEEMRD
alpha-sS1 RQRYRILNPVAIPEGQFIDSRKGTEKLLSSLDIDHNQYKFGHTKVFFKAGLLGLLEEMRD
beta-sS1 α28 780 790 800 810
beta-sS1 ERLSRIITRIQAQSRGVLARMEYKKLLERRGSGRGSIDTWV
alpha-sS1 ERLSRIITRMQAQARGQLMRIEFKKIVERRGSGRGSIDTWV

B

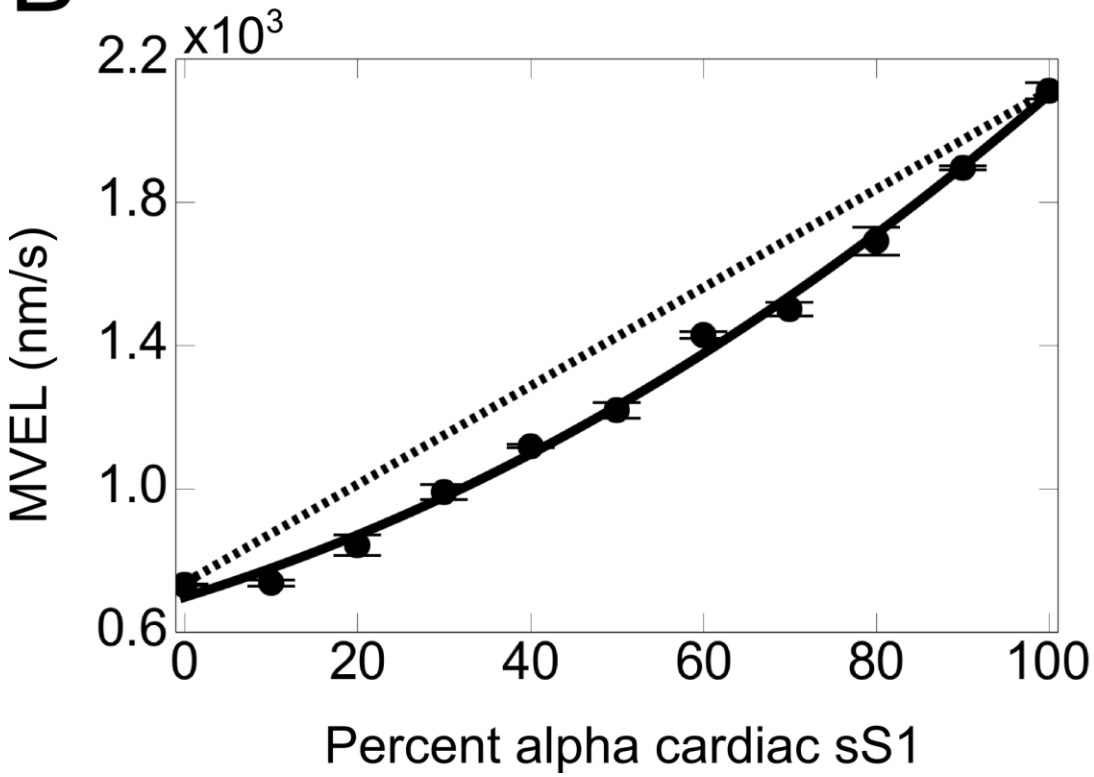

x10^3
2.2
1.8
1.4
1.0
0.6
MVEL (nm/s)
0
20
40
60
80
100
Percent alpha cardiac sS1

Figure S4, Related to Figure 4. Sequence alignment of alpha and beta cardiac sS1 constructs and alpha and beta cardiac sS1 mixing experiments. A. Sequence alignment of alpha and beta cardiac sS1 constructs. Top row in the alignment shows the secondary structure of beta cardiac sS1 determined from the cardiac myosin model derived from cardiac myosin structure deposited in pdb (pdb id: 4DB1). In the secondary structure row, ribbon is for α-helix and horizontal arrow is for β-strand. Red highlight is for identical residues between alpha and beta cardiac sS1. Blue vertical arrow shows the insertions in alpha cardiac sS1 with respect to beta cardiac sS1. Both sequences end with GSG linker and the affinity clamp recognition sequence. α28 shows the lever arm helix which is identical in length for alpha and beta cardiac sS1. Human α- and β-cardiac sS1 alignment is performed using clustalW (Larkin et al., 2007). Only the relevant portions of the alignment are displayed. **B.** Representative MVEL data for alpha and beta cardiac sS1 mixing experiments. MVEL is from unfiltered mobile filament velocity distribution. Confidence intervals are the standard error of mean from at least four movie replicates. Solid line shows the best quadratic equation fit to the data. Dashed line shows the linear sum of alpha (100% alpha) and beta cardiac sS1 (0% alpha) velocities weighted by their ratios in the mix. Downward curvature in solid line with respect to the dashed suggests that beta cardiac sS1 generates higher ensemble force than alpha cardiac sS1.

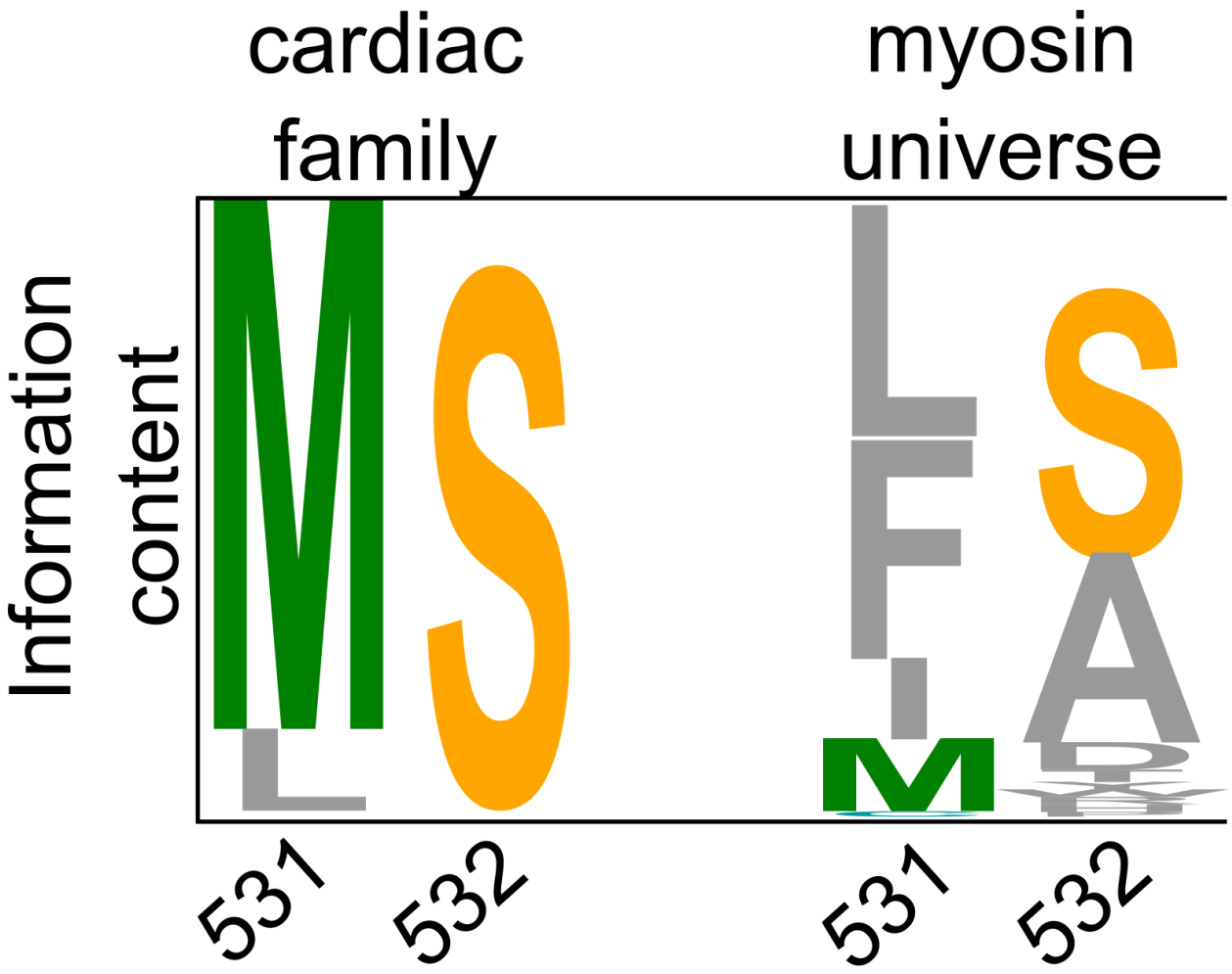


Figure S5, Related to Figure 7. Residue conservation profiles for positions 531 and 532. The Hidden Markov Model logo showing the likelihood of observing different residues at positions 531 and 532 for the cardiac myosin family (left) and the myosin universe (right). The height of each letter is proportional to the probability of observing the residue at that position. The logos are generated from multiple alignments retrieved from the protein family database PFAM (Finn et al., 2014) using Skylign (Wheeler et al., 2014).

Supplemental Tables

Table S1, Related to Figures 4-6. Unloaded velocity, ATPase, loaded in vitro motility and omecamtiv mecarbil binding parameters for human alpha and beta cardiac sS1, and the mutants M531R and S532P.

	TOP5%[a]	PLATEAU[a]	kcat[b]	K_m[b]	K_S[c]	K_D[d]
beta	960	900	5.29	50.03	1.0	0.14
	± 80	± 80	± 0.03	± 5.19		± 0.04
alpha	2910	2740	15.44	96.22	0.18	0.17
	± 140	± 140	± 1.01	± 1.18	± 0.02	± 0.05
M531R	910	880	8.82	16.95	1.73	-
	± 50	± 50	± 0.1	± 5.32	± 0.29	
S532P	610	530	1.74	69.80	0.65	-
	± 30	± 30	± 0.06	± 15.55	± 0.07	

[a] TOP5% and PLATEAU are in nm/s. Filaments that are tracked longer than 10 frames are included in the analysis. A 20% tolerance filter is applied to eliminate intermittently moving filaments with a velocity dispersion higher than 20% of their mean within a 5-frame window. Each velocity point in the analysis is the average velocity over 5 frames. TOP5% is the mean of filament velocities ranking in top 5%. The PLATEAU velocity is from the fit of the single exponential decay to maximum filament velocities (see Figure 1 legend and Experimental Procedures). 95% confidence intervals are the standard error of means from at least four replicate movies from three different preparations of sS1.

[b] kcat and K_m are in s^{-1} and μM. kcat and K_m are from the fit of the Michaelis-Menten equation to ATPase rates measured at different F-actin concentrations. Rates are the average of at least three replicates from three different preparations of sS1. For ATPase measurements, sS1 with a C-terminal eGFP tag is used, which is different from the constructs used for in vitro motility studies. 95% confidence intervals are from 100 iterations of fitting the Michaelis-Menten equation to bootstrapped ATPase data (see Experimental Procedures).

[c] K_S are from the fit of the Stop Model to loaded in vitro motility percent time mobile data. K_S is in nM. Mutant and α-cardiac sS1 K_S values are normalized with respect to β-cardiac sS1values. Each loaded in vitro motility data comparing α-cardiac sS1 and the mutants with respect to β-cardiac sS1 is measured separately with different sS1 preparations. Each comparison is performed with at least three different sS1 preparations. 95% confidence intervals are from 100 iterations of fitting the Stop Model to bootstrapped loaded in vitro motility data comparing the variant and beta cardiac sS1.

[d] K_D is the disassociation constant for omecamtiv mecarbil binding to cardiac sS1 determined from fitting the binding isotherm to mean velocity data (see Supplemental Experimental Procedures). Each omecamtiv mecarbil titration experiment was measured separately with three different sS1 preparations. 95% confidence intervals are from 100 iterations of fitting the binding isotherm to bootstrapped velocity data. K_D is only measured for α- and β-cardiac sS1. K_D is in μM.

Supplemental Movies

Movie S1, Related to Figure 2. α-cardiac sS1 unloaded in vitro motility.

Movie S2, Related to Figure 2. α-cardiac sS1 loaded in vitro motility at 0.1 nM utrophin.

Movie S3, Related to Figure 2. β-cardiac sS1 unloaded in vitro motility.

Movie S4, Related to Figure 2. β-cardiac sS1 loaded in vitro motility at 0.1 nM utrophin.

Supplemental Experimental Procedures

A. Fast Automated Spud Trekker (FAST) Algorithm Description

Fast Automated Spud Trekker (FAST) is written in Python in an object-oriented fashion. FAST makes use of existing python libraries such as Numpy (van der Walt et al., 2011) for matrix algebra, Matplotlib (Hunter, 2007) for plotting graphs, Scipy (Jones et al., 2001-) for least squares fitting, and OpenCV (Bradski) and Scikit-Image (van der Walt et al., 2014) for image processing. For parallel processing of movie frames, a unix bash shell script, PPSS, is used (louwrentius, 2011). This document is intended to provide a brief description of the main algorithms used in FAST. Under each subheading, a snippet of code for the algorithm described is provided. For the details of the FAST classes and routines, see the FAST package available at http://spudlab.stanford.edu/FAST.

FAST is currently available for Linux OS (Ubuntu 14.04). Academic license for FAST is available for free on http://spudlab.stanford.edu/FAST.

Summary of FAST graphical output

One feature of FAST is to estimate the ideal or maximum gliding velocity from the in vitro motility assay. Maximum velocity is estimated from a distribution that is filtered with respect to filaments with fluctuating velocities. In FAST, the filtering method is referred to as “tolerance filtering” (see maximal velocities in length-velocity analysis section). With tolerance filtering, track segments with velocity dispersion higher than a fraction of the average segment velocity are discarded. Filtered and unfiltered distributions are displayed in FAST output side by side to show the effect of tolerance filtering.

FAST provides two measures for maximum unloaded velocity: 1) the maximum plateau velocity (PLATEAU) determined from single exponential decay function fit to the maximum actin filament velocities as a function of actin filament length (see length-velocity analysis). 2) the model-independent maximum velocity, TOP5%, is determined from the mean velocity of the filament velocities ranking in the top 5%.

Another feature of FAST is to estimate the percentage of stuck filaments (%STUCK), which is one of the non-ideality measures for the assay. Filaments are identified as stuck if their effective speed is less than 80 nm/s along their track, 80 nm representing one pixel on the camera used. In FAST, stuck filaments are segregated from mobile filaments and only mobile filaments contribute to velocity distribution. Together with %STUCK, filament length distribution is a useful metric in assessing the quality of motility in unloaded conditions.

For quick assessment of the effect of tolerance filtering on velocity distribution, FAST output displays how the mean of the velocity distribution (MVEL) and TOP5% velocities change with decreasing tolerance in filtering. For human β-cardiac sS1, MVEL increases with decreasing tolerance filtering and TOP5% velocity remains relatively

constant, indicating that low velocity values are contributed by intermittently moving filaments and TOP5% velocities are from smoothly moving filaments.

FAST completes the analysis of in vitro motility movies in 3 steps

Step 1 - Filament Extraction

The first step in the execution of FAST is the filament extraction (see Figure S1A). Filament extraction involves automated **a)** background subtraction, **b)** detection of areas containing filaments, **c)** separation of individual filaments, and **d)** filtering of crossing filaments. Using PPSS parallel processing shell script, filament extraction for each frame is performed in parallel by a different processor, thereby utilizing all processors operated under the unix environment efficiently.

a) Background subtraction

For every pixel in the image (see Figure S1B), the lower 5-percentile intensity is subtracted from the 95-percentile value within a disk of 15-pixel radius to get the background-subtracted image (see Figure S1C).

```
#Determine the 95-percentile intensity within a radius of 15 pixels
img_1         = rank.percentile(self.img,self.disk_win,p0=0.95)

#Determine the 5-percentile intensity within a radius of 15 pixels
img_0         = rank.percentile(self.img,self.disk_win,p0=0.05)

#Subtract the 5-percentile from 95-percentile map
self.img_diff = img_1 - img_0
```

b) Detection of areas containing filaments

The mean intensity of the background-subtracted image is used as the cutoff to pick areas that potentially have filaments (see Figure S1D). Segregated patches that have higher pixel intensity are classified as filament islands. Each filament island may have more than one filament.

```
#Cutoff value is the mean intensity of the background subtracted image
self.cutoff_diff                    = np.mean(self.img_diff)

#Define a mask for the pixels with intensity higher than cut-off value
self.mask_diff                      = self.img_diff > self.cutoff_diff

#Label each separate area on the mask with a different number
self.labels_island,self.num_island  = label(self.mask_diff)
```

c) Separation of Individual filaments

In each filament island, the local Otsu threshold value (Otsu, 1975) that maximizes the separation between background and filament intensity is used to determine the location of filaments (see Figure S1E). After a rough selection of filaments is performed, empty

pixels in filament representation are filled and filament edges are smoothened. Filaments that are within 5 pixels of the frame edges are excluded from further analysis.

```
#Pick only pixels with integer intensity values
valid              = self.img_reduced > 0

#Determine the Otsu threshold value
cutoff             = threshold_otsu(self.img_reduced[valid])

#Filament in coarse representation
self.fil_reduced = self.img_reduced > cutoff
self.img_fil     = self.fil_reduced*self.img_reduced

#Label the filaments
fil_labels, fil_features = label(self.fil_reduced)

#In a cluster there may be more than a single filament - each cluster
corresponds to a filament
fine_clusters = watershed(self.fil_reduced,fil_labels,mask=self.fil_reduced)

#Start with an empty list of filaments
self.filaments = []

for i in range(1,fil_features+1):

    xy_bool       = fine_clusters == i
    xy            = np.nonzero(xy_bool)

    #If it is a single pixel island - ignore
    if len(xy[0]) < 2:
        continue

    #Define a new filament
    new_filament        = Filament()

    #Assign the label
    new_filament.label  = self.frame.filament_counter

    #Assign the current to island to the filament
    new_filament.island = self

    #Shrink the size of the filament image
    new_filament.reduce_image(xy)

    #Density of the filament in terms of intensity
    new_filament.fil_density =
np.sum(1.0*self.img_reduced*xy_bool)/np.sum(xy_bool)

    #Fill in the holes in a filament
    new_filament.img_reduced = binary_fill_holes(new_filament.img_reduced)

    #Binary opening-closing to remove extra short branches
    new_filament.img_reduced =
binary_closing(new_filament.img_reduced,structure=disk_1)

    #Add new filament to the list
```

```
        self.filaments.append(new_filament)

        #Increment filament counter
        self.frame.filament_counter += 1
```

d) Filament skeletonization

Filaments that are individually separated in the previous step are skeletonized for their line representation (see Figure S1F).

```
#Skeletonize the image
self.img_skeleton = skeletonize(self.img_reduced)
```

After skeletonization, the number of tips is counted for each filament. Pixels with only one neighboring pixel are classified as tips. Filaments with a number of tips higher than 2 are considered as crossing and eliminated from further analysis (Figure S1G).

Step 2 – Path Generation

After the line representations of the filaments are extracted for each frame, filaments in adjacent frames are compared to construct the paths. To facilitate comparison, each skeletonized filament is represented as connected vectors from one tip to the other (see Figure S1H).

Two filaments between adjacent frames are connected based on selection criteria that take into account only the geometric features of the filaments. There are three scores that together define the selection criteria.

a) Score 1- Euclidean distance and length difference between filaments

The first criterion is that two filaments need to be in close proximity with each other and the length difference should not be high to be considered for comparison. Only filaments that are within a radius of 25 pixels and length difference of 5 pixels are chosen for comparison between adjacent frames.

b) Score 2 - Orthogonal distance between filaments

Our observations suggest that filaments glide along their axis and displacement orthogonal to the filament axis is minimal. To estimate the orthogonal distance between filaments, we take the sum of the cross-products of the vectors $\vec{v}_j$ and $\vec{w}_j$ for $j \in \{1,2,3,\ldots,N\}$ and normalize by the shorter filament's length. For filaments that are different in length, hence the number of vectors, a cumulative sum is performed shifting the shorter filament along longer filament by one pixel at a time. By this approach filaments that are different in length are further penalized.

$$\left|\perp\right| = \left| \frac{\sum_{k=0}^{\Delta L} \sum_{j=1}^{N} v_{j+k} \otimes w_j}{N} \right|$$

In the equation above, $\left|\perp\right|$ is the orthogonal distance between the filaments, N is the length of the shorter filament and ΔL is 1 plus the length difference between the filaments. For all $i+1^{th}$ frame-filaments that are compared with the filament in the i^{th} frame, the orthogonal distance is quantified. Among all possible candidates, the filament that has the lowest orthogonal distance value is selected as the first best choice. If the orthogonal distance difference between the first and second best choice is higher than $10^{0.5}$ and the orthogonal distance for the first best choice is less than 10, the first best choice is passed onto the next test.

c) Score 3 - Shape overlap between filaments

The second criterion for filament comparison between adjacent frames is the shape overlap. Going from frame i to i+1, we don't expect filament shape to change drastically. To quantify the shape overlap score, the dot product of filament vectors are summed.

$$\text{overlap} = \frac{\sum_{k=0}^{\Delta L} \sum_{j=1}^{N} v_{j+k} w_j}{N \Delta L}$$

The overlap value is 1 for identical filaments oriented in the same direction, 0 for filaments that are completely orthogonal to each other, and -1 for identical filaments oriented in opposite direction. Filaments that have an overlap score absolute value higher than 0.4 pass the criteria for shape overlap.

It is possible that one filament in frame i+1 has two matching filaments in frame i based on the three selection criteria. In this case, the filament in frame i+1 is linked to the filament in frame i with the lowest orthogonal distance.

d) Determination of filament gliding direction and instantaneous velocity

Once the two filaments in adjacent frames are connected, the next step is the determination of the relative orientation of filaments, filament gliding direction and the instantaneous velocity.

Relative filament orientations are determined based on the sign of the overlap score. If the overlap score is positive, filaments are oriented in the same direction. If the overlap score is negative, filaments are oriented in opposite direction. Filaments in opposite orientation of their reference filament are reversed at this step. The reference filament is the filament in the earlier frame.

Filament gliding direction is determined from the average orientation of distance vectors between the filaments with respect to the reference filament axis. The distance vector between $\vec{v}_j$ and $\vec{w}_j$ is defined as $\Delta\vec{d}_{j,k} = \vec{v}_j - \vec{w}_k$.

$$\text{gliding direction} = \sum_{k=0}^{\Delta L}\sum_{j=1}^{N} v_{j+k}\Delta d_{j+k,j}$$

If the value of gliding direction is positive, the filament is moving in the same orientation of the filament. If the value of gliding direction is negative, the filament is moving in the opposite orientation of the filament. The expression below is for instantaneous velocity.

$$\text{instantaneous velocity} = \frac{\sum_{k=0}^{\Delta L}\sum_{j=1}^{N}\left|\Delta d_{j+k,j}\right|}{N\Delta L}$$

e) Connection of broken paths

As discussed above, crossing filaments are eliminated from the analysis; therefore filament crossing leads to new path generation, and the history of the filament is lost. To connect broken paths of the same filament, we apply a selection criteria similar to the one described for connecting filaments in adjacent frames.

For each path built, upcoming paths that start within the 5-frame window of the earlier path's final frame are considered for connection. The comparison is made between the final filament trace of earlier path and the first filament trace of the upcoming path. If the overlap score for the two filaments is higher than 0.4, the orthogonal distance between the filaments is less than 1, the Euclidean distance between the filaments is less than d x 25 (where d is the number of frames between the paths), and the velocity of travel between the connection points of the paths is in between 0.5 and 2 times the average velocity of the paths, the two paths are merged as a single path (see Figure S1I).

Length-velocity analysis

From the constructed filament tracks, several analyses are performed to extract velocity measures relevant for different aspects of myosin function.

a) Maximal velocities

The first analysis involves determining maximal velocities by filtering out completely stuck filaments and filaments that glide with intermittent stops that cause filaments to glide at suboptimal velocities. Each "velocity point" in our analysis refers to the average of 5 consecutive instantaneous velocities for a single filament. Therefore, in total N-5 velocity points (gray points in Figure 1A) are measured for a single filament tracked for N frames.

Stuck filaments are determined based on a simple selection criterion. If average filament velocity along a path is less than 80 nm/s, the filament is defined as stuck. The reason for choosing 80 nm/s is that in our imaging system, 1 pixel is 80 nm and for beta cardiac myosin motility movies are taken at 1 frame per second. For our analysis, we pick only paths that are longer than or equal to 10 frames, which prevents us from false labeling of filaments with short paths.

To filter out filaments with intermittent stops, we apply a selection criterion that is defined as tolerance filtering. With tolerance filtering, only filament path segments with a velocity dispersion less than some fraction of the average segment velocity is kept for further analysis.

$$\sigma([V_i, V_{i+1}, \ldots, V_{i+N}]) \leq c.\mu([V_i, V_{i+1}, \ldots, V_{i+N}])$$

In the expression above σ and μ stand for standard deviation and average operators respectively for velocities within an N+1 frame window, and c stands for the fraction value. Both the size of the segment and the fraction value are user defined. 20% tolerance filtering means c is equal to 0.2. For all our analyses, we choose a 5-frame window size.

FAST provides two maximal velocity measurements. The first maximal velocity is called TOP5% velocity. TOP5% velocity is the mean of velocities ranking higher than the top 5-percentile. In all our analyses in this manuscript, we determined the TOP5% of a collection of paths that are longer than 10 frames, subjected to 20% tolerance filtering with 5-frame averaging window size. The second maximal velocity is called PLATEAU. PLATEAU is determined by fitting a single exponential decay function to the “maximum instantaneous velocities” of filament paths that pass the 20% tolerance filtering criterion. “Maximum instantaneous velocities” (blue points in Figure 1A) are the maximum of instantaneous velocities (gray points in Figure 1A) measured for each filament.

$$V = V_{PLATEAU} - Ae^{-L/\tau}$$

In the equation above, $V_{PLATEAU}$ is the maximal PLATEAU velocity, τ is the decay parameter and L is the actin filament length.

b) Velocity histograms

For loaded in vitro motility, we define MVEL as the mean of the unfiltered velocity distribution. For filtered velocity distributions, mean velocity is referred as $MVEL_X$ where x is for the percent tolerance parameter (e.g. $MVEL_{20}$ is the mean of the velocity distribution filtered by 20% tolerance filtering). Velocity distributions are from frame-to-frame instantaneous velocities averaged within a window of a certain number of frames. The window size for all of our analyses is 5 frames. For stuck filament paths, frame-to-frame instantaneous velocities are assigned to 0 and excluded from the velocity

distribution. Paths contribute in velocity distributions proportional to their length in number of frames.

Percent stuck is another parameter relevant for loaded in vitro motility. Percent stuck is defined as the fraction of velocities that are equal to 0 in velocity distribution including stuck filaments.

$$\text{percent stuck} = \frac{\sum_{i=1}^{N} \delta(V_i)}{\sum_{i=1}^{N} 1}$$

In the expression above, N is the number of velocities in the distribution. $\delta(V_i)$ is equal to 1 if V_i is equal to 0, and it is equal to 0 otherwise.

B. Loaded In Vitro Motility Model Description

Derivation of percent time mobile from percent mobile filaments and MVEL

As described in the main text, percent mobile filaments and MVEL alone do not report the complete inhibition of actin gliding by utrophin. Percent time mobile is the combined parameter that reports complete inhibition.

Percent time mobile is derived combining percent mobile filaments and MVEL. Percent mobile filaments is the time weighted fraction of mobile filaments. Since mobile filaments temporarily stop gliding, we need to consider only the time spent gliding to determine percent time mobile.

The mobile fraction of gliding is estimated from the ratio of MVEL and TOP5% velocities. If filaments don't encounter obstacles, they glide at maximum velocity. Due to obstacles, filaments stop and temporary stops cause MVEL to be less than TOP5% velocity measured in unloaded conditions.

$$MVEL(\Delta t_{mobile} + \Delta t_{stuck}) = TOP5\% \Delta t_{mobile}$$

$$\frac{\Delta t_{mobile}}{\Delta t_{mobile} + \Delta t_{stuck}} = \frac{MVEL}{TOP5\%}$$

Percent time mobile is determined from the product of percent mobile filaments and MVEL to TOP5% ratio. The ratio of MVEL and unloaded TOP5% is equal to the mobile time fraction of gliding filaments, thus multiplying this ratio with percent mobile filaments yields the actual mobile time fraction of all filaments.

$$< \% \text{ time mobile} >=< \% \text{ mobile filaments} > \frac{MVEL}{TOP5\%}$$

Stop Model for percent time mobile

Percent time mobile is defined as the total duration of filaments in the "mobile" state divided by the total duration of filaments in the "stuck" and "mobile" states.

$$< \% \text{ time mobile} >= \frac{t_{mobile}}{t_{mobile} + t_{stuck}} \qquad \text{Eq. S1}$$

Stuck propensity (X_{stuck}) is defined as the ratio of stuck and mobile time durations.

$$\chi_{stuck} = \frac{t_{stuck}}{t_{mobile}}$$

We rearrange Eq. S1 dividing the numerator and denominator by t_{mobile}.

$$<\%\ \text{time mobile}>= \frac{1}{1+\chi_{stuck}} \qquad \text{Eq. S2}$$

We observe that there is a linear relationship between stuck propensity and utrophin concentration (see Figure 3D). If we assume the linear relationship between stuck propensity and utrophin concentration, we reach an expression that is similar to thermodynamic association-disassociation equilibrium.

$$\chi_{stuck} = \frac{[F_{stuck}]}{[F_{mobile}]} = \frac{[\text{utrophin}]}{K_S} \qquad \text{Eq. S3}$$

In expression above, F_{stuck} (i.e. t_{stuck}) and F_{mobile} (i.e. t_{mobile}) are the concentrations of stuck and mobile filaments respectively, [utrophin] is utrophin concentration in nM and K_S is the disassociation constant for utrophin-actin interaction in nM. We propose that K_S depends on myosin's ensemble force; higher ensemble forces rupture utrophin-actin interaction more efficiently thus causing K_S to be higher.

In unloaded conditions there is a fraction of stuck filaments, therefore χ_{STUCK} in the absence of utrophin is greater than 0.

$$\chi_{STUCK}([u]) = \chi_{STUCK}(0) + \frac{[\text{utrophin}]}{K_S} \qquad \text{Eq. S4}$$

We derive the utrophin dependent percent mobile expression by substituting χ_{STUCK} in Eq. S2 with Eq. S4.

$$<\%\ \text{time mobile}>= \frac{1}{1+\chi_{STUCK}(0) + \frac{[\text{utrophin}]}{K_S}} \qquad \text{Eq. S5}$$

For percent mobile in unloaded conditions, we define a new paramter, M_0.

$$M_0 = \frac{1}{1+\chi_{STUCK}(0)} \qquad \text{Eq. S6}$$

Finally, the expression for percent mobile is rearranged by incorporating M_0 in Eq, S5.

$$<\%\ \text{time mobile}>= \frac{K_S M_0}{K_S + M_0[\text{utrophin}]} \qquad \text{Eq. S7}$$

We fit Eq. S7 (referred as Eq. 1 in main text) to percent time mobile data to determine M_0 and K_S for mutant and WT human cardiac sS1.

Parabolic velocity dispersion expression for loaded in vitro motility

We observe that utrophin causes filaments to glide with intermittent stops causing velocities to fluctuate between high and low velocities. Based on this observation, we expect a parabolic relationship between velocity dispersion and average velocity, <V>, in presence of utrophin. Velocity dispersion is defined as the standard deviation of a velocity trajectory with velocities alternating between maximal unloaded velocity V_u and 0.

$$\text{velocity dispersion} = \sqrt{< V^2 > - < V >^2} \qquad \text{Eq. S8}$$

$$< V >= p_{MOVE} V_U \qquad \text{Eq. S9}$$

$$< V^2 >= p_{MOVE} V_u^2 \qquad \text{Eq. S10}$$

p_{MOVE} is the fraction of time spent gliding at V_u in a filament's trajectory.

We rewrite $<V^2>$ as a function of <V> and V_u.

$$< V^2 >=< V > V_u \qquad \text{Eq. S11}$$

Substituting the expression for $<V^2>$ in Eq. S8 with Eq. S11, we derive the expression for velocity as a function of average velocity.

$$\text{velocity dispersion} = \sqrt{< V > V_u - < V >^2} \qquad \text{Eq. S12}$$

This equation can be further simplified by dividing both sides V_u to yield normalized velocity dispersion ranging between 0 and 0.5.

$$\text{normalized velocity dispersion} = \sqrt{p_M - p_M^2} \qquad \text{Eq. S13}$$

Eq. S13 has the maximum value of the normalized velocity dispersion when p_m is 0.5. In other words velocity dispersion is maximum when <V> is half the V_u.

SMS velocity decay rate as a function utrophin concentration

Here, we derive the expression for the decay rate of Stop Model percent time mobile as a function of utrophin concentration.

Percent time mobile decay rate (k) is defined as the relative decay of percent time mobile with utrophin.

$$k([u]) = -\frac{d < \%\ \text{time mobile} >}{d[u]} \frac{1}{< \%\ \text{time mobile} >} \qquad \text{Eq. S14}$$

After some mathematical transformations, we reach the final expression for the decay rate.

$$k([u]) = \frac{<\%\ \text{time mobile}>}{K_S} \qquad \text{Eq. S15}$$

We find that decay rate linearly depends on percent time mobile. As percent time mobile decreases, the apparent impact of utrophin on percent time mobile decreases as well.

C. Rest of Supplemental Experimental Procedures

Expression and purification of utrophin construct

For expression of the constructs, transformed cells were grown overnight in 20 ml Luria Broth at 37ºC. In the morning, overnight cultures were transferred into 1L cultures. Cells are grown at 37ºC for 4-5 hours. Once the OD_{600} of the culture reached 0.6-0.8, cells were induced with 1 mM IPTG. The culture was incubated at 37ºC for an additional 4 hours before cells were harvested. Harvested cells were resuspended in 150 mM NaCl, 10 mM Imidazole, 25 mM Tris-Cl, 1 mM DTT, pH 8.0 solution (Buffer A). Cells were lysed at 4ºC using an Emulsiflex (Avestin, Canada) homogenizer for 5 minutes three times. Lysed cells were sedimented at 16000 rpm in JA-20 rotor for 10 min at 4ºC and the pellet retained for an inclusion body preparation. The pellet was resuspended in 8 M Urea, 150 mM NaCl, 10 mM Imidazole, 25 mM Tris-Cl, 1 mM DTT, pH 8.0 solution (Buffer B) at 4ºC and sedimented at 16000 rpm in JA-20 rotor at 4ºC. The supernatant was passed through a bench-top nickel column at 4ºC, and the column was washed with buffer B. Denatured protein was refolded on the column by passing buffer A through the column. Utrophin bound to the nickel column was eluted with 150 mM NaCl, 500 mM Imidazole, 25 mM Tris-Cl, 1 mM DTT in single step. The eluted protein was concentrated and dialyzed overnight against 150 mM NaCl, 25 mM Tris-Cl, 1 mM DTT, pH 8.0 at 4ºC. The dialyzed protein was flash frozen in liquid nitrogen and stored at -80ºC.

Expression and purification of SNAP-PDZ18

Briefly, Rosetta (DE3) cells were transformed with pHFT2 plasmid carrying SNAP-PDZ18. A 1L bacterial culture grown in Luria Broth medium at 37ºC was induced with 1 mM IPTG once the OD_{600} of the culture reached 0.6-0.8. 4-6 hours after induction, cells are sedimented by centrifugation and resuspended in 150 mM NaCl, 25 mM Tris-Cl, pH 8.0 (Buffer X). Resuspended cells were lysed using an Emulsiflex (Avestin, Canada) homogenizer. Lysed cells were sedimented and the supernatant loaded onto the bench-top nickel column. After a column wash with buffer X, protein was eluted off the column with 500 mM Imidazole 150 mM NaCl, 25 mM Tris-Cl, pH 8.0 at 4ºC. Eluted protein was concentrated and the buffer was exchanged to buffer C. The SNAP-PDZ18 was flash frozen with liquid nitrogen and stored at -80ºC.

Unloaded and loaded in vitro motility measurements

Coverslips were coated with 0.2% nitrocellulose (Ernst Fullam) dissolved in amyl acetate (Sigma) and air dried for at least 0.5 hour. Using double-sided tape, four channels were created between the cover slip and the slide. On each slide, the motility assay under four different conditions were performed simultaneously. The reagents were loaded into the channels in the following order: 1) 10 µl of 2 µM SNAP-PDZ18 diluted in Assay Buffer (AB; 25 mM Imidazole (pH 7.5), 25 mM KCl, 4 mM $MgCl_2$, 1 mM EGTA, and 1 mM DTT) was loaded and incubated for 2 minutes. 2) 20 µl of 1 mg/ml BSA diluted in AB (ABBSA) was loaded to block the surface from nonspecific subsequent attachments and incubated for 2 minutes. 3) 10 µl of a mixture of C-tagged

human cardiac sS1 and utrophin at desired concentrations was loaded and incubated for 3 minutes. Prior to mixing myosin and utrophin together, myosin and utrophin dilutions were prepared in AB with 0.1 mg/ml BSA. 4) 20 µl of ABBSA was loaded to wash the channels. 5) Finally, 10 µl of the GO solution [1-5 nM tetramethylrhodamine (TMR)–phalloidin labeled bovine actin, 2 mM ATP, an oxygen scavenging system (0.3–0.4% glucose, 0.25 µg/mL glucose oxidase, 0.45 µg/mL catalase), and an ATP regeneration system (1 mM phosphocreatine, 0.1 mg/mL creatine phosphokinase)] was loaded. Movies were obtained at 23°C at a frame rate of 1 Hz for beta-cardiac sS1 and 3 Hz for alpha cardiac-sS1 using a Nikon Ti-E inverted microscope with Andor iXon+EMCCD camera model DU885. Unloaded and loaded in vitro motility assays for each cardiac sS1 variant were repeated with at least three fresh protein preparations. At each condition at least four movies with a duration of 30 seconds were recorded. At high concentrations of utrophin, we observed significant actin filament severing that increases with myosin concentration. To minimize filament severing due to utrophin, we empirically determined the lowest human cardiac sS1 concentration at which actin gliding velocities were as high as the velocities determined at high myosin concentration. The loaded in vitro motility assay was performed at this low myosin concentration.

Omecamtiv mecarbil titration experiments

K_D for omecamtiv mecarbil binding to cardiac sS1 was determined from fitting the binding isotherm to mean velocity data.

Mean velocity is defined as the linear sum of the omecamtiv mecarbil bound and unbound cardiac sS1 velocities.

$$<MVEL>= f_{bound} V_{bound} + (1 - f_{bound}) V_{unbound} \qquad \text{Eq. S16}$$

f_{bound} is the fraction of sS1 bound to omecamtiv mecarbil, V_{bound} and $V_{unbound}$ are the MVEL of the bound and unbound states respectively.

f_{bound} is determined from the binding isotherm relationship.

$$f_{bound} = \frac{1}{1 + K_D / [OM]} \qquad \text{Eq. S17}$$

K_D and [OM] are the disassociation constant and omecamtiv mecarbil concentrations respectively. Both parameters are in µM.

Replacing f_{bound} in Eq. S16 with expression in Eq. S17, we get the final expression fitted to data.

$$<MVEL>= \frac{1}{1 + K_D / [OM]} V_{bound} + (1 - \frac{1}{1 + K_D / [OM]}) V_{unbound} \qquad \text{Eq. S18}$$

From the fit K_D, V_{bound} and $V_{unbound}$ are determined.

Human cardiac sS1 clean up for the in vitro motility assay

Before starting the in vitro motility assay, cardiac sS1 was cleaned up from partially inactive motors that do not release actin in the presence of excess ATP. For the clean-up, cardiac sS1 was incubated with 20 times more concentrated F-actin (typically ~50 μM) in 4 mM ATP for 10 minutes. To completely precipitate all F-actin as magnesium paracrystals, the myosin F-actin mix was brought to 50 mM $MgCl_2$. Formation of magnesium paracrystals are observed as white precipitates. After 10 minutes of incubation with $MgCl_2$, the sample was sedimented at 95,000 rpm for 20 minutes using a TLA100 rotor. The supernatant was collected and its concentration was measured using the Bradford Assay. A mock clean-up procedure without myosin was also performed simultaneously with the myosin clean up. The final supernatant from the mock clean-up was used as the blank for the Bradford Assay. The quality of the myosin clean-up was assessed by the percent of stuck filaments under unloaded conditions. We repeated the myosin clean-up procedure until the percent stuck dropped below 10%.

Homology modeling of human cardiac sS1 structure

The actin-beta cardiac myosin model is built based on an actomyosin model built from myosin structure fitted to F-actin cryo-EM data (Mendelson and Morris, 1997). Residues missing in the beta cardiac myosin crystal structure (pdb id: 4DB1) are modeled in from the chicken skeletal myosin structure (pdb id: 2MYS).

Bootstrapping for confidence interval estimation

To estimate the 95% confidence interval for parameters of a model fitted to experimental data, the model was iteratively fitted to bootstrapped data. Bootstrapped data was obtained by adding random permutation of residuals from the initial fit to ideal data. Ideal data is the expected data based on the fitted parameters. Bootstrapping iteration was performed for 100 times for each data set. 95% confidence interval for a parameter was estimated to be twice the standard deviation of the fitted parameter distribution. Bootstrapping was used to estimate the confidence intervals for K_S from loaded in vitro motility data and K_D from the OM titration data.

Supplemental References

Bradski, G. Dr. Dobb's Journal of Software Tools.

Finn, R.D., Bateman, A., Clements, J., Coggill, P., Eberhardt, R.Y., Eddy, S.R., Heger, A., Hetherington, K., Holm, L., Mistry, J.*, et al.* (2014). Pfam: the protein families database. Nucleic acids research *42*, D222-230.

Hunter, J.D. (2007). Matplotlib: A 2D graphics environment. Comput Sci Eng *9*, 90-95.

Jones, E., Oliphant, T., Peterson, P., and others (2001-). SciPy: Open source scientific tools for Python.

Larkin, M.A., Blackshields, G., Brown, N.P., Chenna, R., McGettigan, P.A., McWilliam, H., Valentin, F., Wallace, I.M., Wilm, A., Lopez, R.*, et al.* (2007). Clustal W and Clustal X version 2.0. Bioinformatics *23*, 2947-2948.

louwrentius (2011). |P|P|S|S| - (Distributed) Parallel Processing Shell Script.

Mendelson, R., and Morris, E.P. (1997). The structure of the acto-myosin subfragment 1 complex: results of searches using data from electron microscopy and x-ray crystallography. Proceedings of the National Academy of Sciences of the United States of America *94*, 8533-8538.

Otsu, N. (1975). A threshold selection method from gray-level histograms. Automatica *11*, 23-27.

van der Walt, S., Colbert, S.C., and Varoquaux, G. (2011). The NumPy Array: A Structure for Efficient Numerical Computation. Comput Sci Eng *13*, 22-30.

van der Walt, S., Schönberger, J.L., Nunez-Iglesias, J., Boulogne, F., Warner, J.D., Yager, N., Gouillart, E., and Yu, T. (2014). scikit-image: Image processing in Python (PeerJ PrePrints).

Wheeler, T.J., Clements, J., and Finn, R.D. (2014). Skylign: a tool for creating informative, interactive logos representing sequence alignments and profile hidden Markov models. BMC bioinformatics *15*, 7.