Supporting Information for

Microvascular architecture and physiological fluctuations constrain the control of cerebral microcirculation

Xiang Ji, Yuchen Zhao, Lu Bai, Kai Wang, David Kleinfeld

David Kleinfeld
E-mail: dk@physics.ucsd.edu
Xiang Ji
E-mail: xij072@ucsd.edu

This PDF file includes:

Contents

1. Network flow response to single vessel modulation

Consider a vascular network with N nodes and E edges. We denote the conductance of edge α between node i and j as g_α and define an incident vector that represents the direction from node i to node j as:

$$(\boldsymbol{b}_\alpha)_k = \delta_{ik} - \delta_{jk}. \quad [1]$$

Stacking all incident vectors gives the incident matrix $B \in \mathcal{R}^{N\times E}$, where the α-th column is the incident vector for edge α. The weighted graph Laplacian matrix is then $L = BGB^T$, where $G_{\alpha\beta} = \delta_{\alpha\beta} g_\alpha$ is the diagonal conductance matrix. Denote the pressure at node i as ϕ_i and the injected flow into node i as $q_i^{(E)}$, conservation of flow then gives:

$$L\boldsymbol{\phi} = \boldsymbol{q}^{(E)}. \quad [2]$$

Under a fixed-flow boundary condition, the weighted graph Laplacian matrix is positive semi-definite. If the entire network is single-connected, then L has one zero eigenvalue and all other eigenvalues are strictly positive. Denoting the Moore-Penrose pseudoinverse of L as L^{-1}, the pressure at each node can be computed as $\phi_i = L^{-1}\boldsymbol{q}^{(E)}$.

Consider external flow I and $-I$, which is injected to nodes i and j, respectively, connected by edge α. Thus $\boldsymbol{q}^{(E)} = I\boldsymbol{b}_\alpha$. The effective network resistance across edge α is then given by

$$R_\alpha = \frac{\phi_i - \phi_j}{I} = \frac{\boldsymbol{b}_\alpha^T (L^{-1}\boldsymbol{q}^{(E)})}{I} = \boldsymbol{b}_\alpha^T L^{-1} \boldsymbol{b}_\alpha \quad [3]$$

and the induced flow in branch β between node m and n, I_β is given by

$$I_\beta = g_\beta \boldsymbol{b}_\beta^T \boldsymbol{\phi} = g_\beta \boldsymbol{b}_\beta^T L^{-1} \boldsymbol{b}_\alpha I \equiv f_\alpha^\beta I, \quad [4]$$

where f_α^β is the fractional flow change in edge β in response to flow injection around edge $\alpha = (i, j)$ and therefore $|f_\alpha^\beta| \in [0, 1]$.

A. Self-modulation. Consider an edge α between two internal nodes (i, j) with baseline flow Q_α given by:

$$Q_\alpha = g_\alpha(\phi_i - \phi_j) = g_\alpha \boldsymbol{b}_\alpha^T L^{-1} \boldsymbol{q}^{(E)}. \quad [5]$$

If the conductance of this edge is increased by ϵ, the weighted graph Laplacian L gets a rank-1 modification, i.e.,

$$L' = L + \epsilon \boldsymbol{b}_\alpha \boldsymbol{b}_\alpha^T \quad [6]$$

and the pseudoinverse of L' can be calculated using the Sherman-Morrison formula:

$$(L')^{-1} = L^{-1} - \epsilon \frac{L^{-1}\boldsymbol{b}_\alpha \boldsymbol{b}_\alpha^T L^{-1}}{1 + \epsilon R_\alpha}. \quad [7]$$

The resulting flow Q'_α is then given by:

$$Q'_\alpha = (g_\alpha + \epsilon)\boldsymbol{b}_\alpha^T (L')^{-1} \boldsymbol{q}^{(E)} = \frac{1 + \hat{\epsilon}_\alpha}{1 + \hat{\epsilon}_\alpha f_\alpha^\alpha} Q_\alpha, \quad [8]$$

where $\hat{\epsilon}_\alpha = \epsilon / g_\alpha \in [-1, \infty)$ is the fractional conductance changes in edge α. The normalized flow change is then

$$\frac{\Delta Q_\alpha}{Q_\alpha} = \frac{1 - f_\alpha^\alpha}{1 + \hat{\epsilon}_\alpha f_\alpha^\alpha} \hat{\epsilon}_\alpha. \quad [9]$$

As the effective network resistance R_α is non-greater than the edge resistance $r_\alpha = g_\alpha^{-1}$, $f_\alpha^\alpha = g_\alpha R_\alpha \in [0, 1]$. Therefore, single vessel dilation always increases its own flow. Noting that $\hat{\epsilon}_\alpha = -1$ corresponds to the case when the segment is completely blocked, we find $\Delta Q'_\alpha = -Q_\alpha$ as expected. However, neither single vessel dilation nor restriction could reverse the flow direction in itself.

If we neglect the dependence of effective viscosity on vessel radius ρ, edge conductance $g \propto \rho^4$ and $\hat{\epsilon}_\alpha = [(\hat{\rho}_\alpha + 1)^4 - 1]$, where $\hat{\rho}_\alpha = \Delta\rho_\alpha / \rho_\alpha$ is the normalized radius change. Equation 9 give:

$$\frac{\Delta Q_\alpha}{Q_\alpha} = \frac{(1 - f_\alpha^\alpha)[(\hat{\rho}_\alpha + 1)^4 - 1]}{1 + f_\alpha^\alpha [(\hat{\rho}_\alpha + 1)^4 - 1]}. \quad [10]$$

For small radius change $\hat{\rho}_\alpha$, we have:

$$\frac{\Delta Q}{Q} = 4(1 - f_\alpha^\alpha)\frac{\Delta\rho_\alpha}{\rho_\alpha} \equiv k_Q \frac{\Delta\rho_\alpha}{\rho_\alpha}. \quad [11]$$

In other words, network constraints reduce the linear response of flow to radius from 4, for a pipe with fixed endpoint pressures, to $4(1 - f_\alpha^\alpha)$, for a pipe in an interconnected network (Figure S1A). The linear response of flow to radius changes is related to f_α^α through

$$f_\alpha^\alpha = 1 - k_Q/4. \quad [12]$$

Similarly, flow speed $v_\alpha = Q_\alpha / (\pi \rho_\alpha^2)$, the speed after radius modulation, v', is related to the original speed through:

$$v'_\alpha = \frac{\sqrt{1 + \hat{\epsilon}_\alpha}}{1 + \hat{\epsilon}_\alpha f_\alpha^\alpha} v_\alpha = \frac{(\hat{\rho}_\alpha + 1)^2}{1 + f_\alpha^\alpha [(\hat{\rho}_\alpha + 1)^4 - 1]} v_\alpha, \quad [13]$$

or

$$\frac{\Delta v_\alpha}{v_\alpha} = \frac{(\hat{\rho}_\alpha + 1)^2}{1 + f_\alpha^\alpha [(\hat{\rho}_\alpha + 1)^4 - 1]} - 1. \quad [14]$$

The right hand side of the equation has roots at $\hat{\rho}_\alpha = 0, -2, -1 \pm \sqrt{-1 + 1/f_\alpha^\alpha}$. For $f_\alpha^\alpha \in [0.5, 1]$, none of the roots are positive, and therefore increasing the radius always decreases the flow speed. For $f_\alpha^\alpha \in [0, 0.5)$, $-1 + \sqrt{-1 + 1/f_\alpha^\alpha} > 0$ and therefore a small dilation $\hat{\rho}_\alpha \in (0, -1 + \sqrt{-1 + 1/f_\alpha^\alpha}]$ could increase flow speed, while a larger dilation decreases the flow speed (Figure S1B). Similarly, for small radius change $\hat{\rho}_\alpha$,

$$\frac{\Delta Q}{Q} = 2(1 - 2f_\alpha^\alpha)\frac{\Delta\rho_\alpha}{\rho_\alpha} \equiv k_v \frac{\Delta\rho_\alpha}{\rho_\alpha}, \quad [15]$$

and

$$f_\alpha^\alpha = \frac{1}{2} - \frac{k_v}{4}. \quad [16]$$

B. Effect on the neighbors. We now consider the effect of dilation of edge α on edge β between internal nodes m and n. Through a similar derivation to that above, we have:

$$Q'_\beta = g_\beta \boldsymbol{b}_\beta^T (L')^{-1} \boldsymbol{q}^{(E)} = \left(1 - \frac{f_\alpha^\beta}{1 + \hat{\epsilon}_\alpha f_\alpha^\alpha} \frac{Q_\alpha}{Q_\beta} \hat{\epsilon}_\alpha\right) Q_\beta, \quad [17]$$

and

$$\Delta Q_\beta = -\frac{\hat{\epsilon}_\alpha f_\alpha^\beta}{1 + \hat{\epsilon}_\alpha f_\alpha^\alpha} Q_\alpha. \quad [18]$$

Therefore, the change of flow in the neighboring vessels is proportional to the basal flow rate in the controlling vessel and depends on the network structure (f_α^β and f_α^α) and the fractional conductance change $\hat{\epsilon}_\alpha$. Therefore, for effective regulation of blood flow, the controlling edge should have a high flow rate. The spatial extent of the significantly affected vessel segments depends on the decay of f_α^β over Euclidean or geodesic distance. The change of flow in edge β is related to flow change in the controlling edge α through

$$\frac{\Delta Q_\beta}{\Delta Q_\alpha} = -\frac{f_\alpha^\beta}{1 - f_\alpha^\alpha}, \quad [19]$$

which is completely determined by the network structure, independent of the magnitude of conductance change $\hat{\epsilon}_\alpha$. As ΔQ_α is distributed across the network, $|\Delta Q_\beta / \Delta Q_\alpha| \leq 1$, and conservation of flow gives $|f_\alpha^\beta| \leq 1 - f_\alpha^\alpha$. Similarly, for two edge β and γ responding to the controlling edge α, their flow responses are related by:

$$\frac{\Delta Q_\beta}{\Delta Q_\gamma} = \frac{f_\alpha^\beta}{f_\alpha^\gamma}, \quad [20]$$

which is also independent of $\hat{\epsilon}_\alpha$.

We now consider flow control in nearest neighbors. As $f_\alpha^\beta = g_\beta \boldsymbol{b}_\beta^T L^{-1} \boldsymbol{b}_\alpha$, Equation 18 can be expressed as:

$$\Delta Q_\beta = \frac{g_\beta \hat{\epsilon}_\alpha Q_\alpha}{1 + \hat{\epsilon}_\alpha f_\alpha^\alpha} \boldsymbol{b}_\beta^T L^{-1} (-\boldsymbol{b}_\alpha), \quad [21]$$

where $(-\boldsymbol{b}_\alpha)$ can be viewed as a unit-current dipole injection with a negative pole at node i and a positive pole at node j. Denote the pressure induced by this dipole as $\boldsymbol{w} = L^{-1}(-\boldsymbol{b}_\alpha)$. We have

$$\boldsymbol{b}_\beta^T L^{-1} (-\boldsymbol{b}_\alpha) = \boldsymbol{b}_\beta^T \boldsymbol{w} = w_m - w_n, \quad [22]$$

and therefore

$$\Delta Q_\beta = \frac{g_\beta \hat{\epsilon}_\alpha}{1 + \hat{\epsilon}_\alpha f_\alpha^\alpha} (w_m - w_n) Q_\alpha. \quad [23]$$

For vasodilation, $\hat{\epsilon}_\alpha > 0$, and therefore

$$sgn(\Delta Q_\beta) = sgn[(w_m - w_n) Q_\alpha], \quad [24]$$

where $sgn(\cdot)$ is the sign function. Suppose $Q_\alpha > 0$, i.e., the basal flow in edge α is from node i to node j. If edge β is directly connected to edge α at node j, i.e., $m = j$ and $n \neq i$, then

$$sgn(\Delta Q_\beta) = sgn(w_m - w_n). \quad [25]$$

As $\mathbf{w}$ is the pressure induced by a dipole across edge α, it represents the discretized version of the solution to a Poisson equation. By the maximum principle of the Poisson equation, the maximum/minimum value can only appear on the domain boundary, i.e., nodes i and j. Therefore, for any node n other than j, we have $w_j > w_n$ and thus

$$\Delta Q_\beta > 0. \quad [26]$$

Consequently, if edge β is directly downstream of edge α, i.e., $Q_\beta > 0$, then increasing conductance of α always increases flow in β. Conversely, if the basal flow direction in edge β is from node n to node m (or equivalently, node j), i.e., $Q_\beta < 0$, increasing conductance in edge α decreases flow in β. Similarly, increasing conductance in an edge will increase flow in its nearest upstream neighbor and decrease flow in its nearest parallel neighbor.

2. Analysis of tracking-based hemodynamic measurement approach

A. Predictive model enables high-speed dense tracking. The accuracy of hemodynamic measurements critically depends on correctly matching blood cells detected at adjacent time points. These labeled blood cells, imaged using low numerical aperture optics, exhibit limited distinguishable intensity features. Matching these nearly identical particles across frames relays on solving the linear assignment matrix C_{ij} that minimizes the total error between the predicted position $\tilde{x}_{i,t}(t+1)$, according to a motion model, and the observed position in the next time point, $x_{j,t+1}$. Specifically, C_{ij} can be found by solving

$$\arg min_{C_{ij}} \sum_{ij} C_{ij} d[\tilde{x}_{i,t}(t+1), x_{j,t+1}], \quad [27]$$

where $d[\tilde{x}_{i,t}(t+1), x_{j,t+1}]$ measures the distance between two points, and Euclidean distance is commonly used. The validity of this approach depends on an assumption that the prediction error is smaller than the average minimal distance $\langle d \rangle$ between particles:

$$d[\tilde{x}_{i,t}(t+1), x_{j,t+1}] < \langle d \rangle. \quad [28]$$

In 1D, the motion model estimates the velocity $v_{i,t}$ for the cell at $x_{i,t}$ and predicts its position at time point $t+1$

$$\tilde{x}_{i,t}(t+1) = x_{i,t} + \frac{\tilde{v}_{i,t}}{\nu_{acq}}, \quad [29]$$

where ν_{acq} is the acquisition frequency (imaging frame rate) and $\tilde{v}_{i,t}$ is the predicted velocity. Consider cell i passing through a vessel segment E. Its true speed, $v_{i,t}$, through the segment between time t and $t+1$ can be expressed as

$$v_{i,t} = v_E + \delta v_{i,t}, \quad [30]$$

where v_E is the average flow velocity in segment E, and $\delta v_{i,t}$ represents the potential spatial and temporal variations. Let ϵ_n be the normalized prediction error:

$$\epsilon_n = |\frac{\tilde{v}_{i,t} - v_{i,t}}{v_{i,t}}|. \quad [31]$$

Equation 28 then gives the maximum velocity that can be reliably measured by tracking:

$$v_{max} \leq \frac{\nu_{acq}}{\epsilon_n} \langle d \rangle. \quad [32]$$

The widely used Crocker–Grier algorithm was developed for tracking the diffusion of colloidal particles, whose expected displacement between frames is 0. This correspond to $\tilde{v}_{i,t} = 0$ and $\epsilon_n = 1$. In the case of blood flow in microcirculation networks, the motions of the blood cells are directed along the vessel. For fast moving blood cells, the inter-frame displacement could be comparable to the $\langle d \rangle$, and therefore strongly degrade the performance of the Crocker-Grier algorithm. However, for such directed movements, if the average edge velocity v_E is known, by setting $\tilde{v}_{i,t} = v_E$, the normalized prediction error can be reduced to $|\delta v_{i,t}/v_E|$. Therefore, assuming a 33% coefficient of variation in flow velocity, knowing the average flow speed in each edge enables reliable tracking of cells moving at a speed 3-times higher under the same experimental condition.

B. Tracer density variation in a multiscale network. When applying particle tracking to study hemodynamics, tracer density is a key experimental parameter that should be carefully chosen to suit the physiological questions of interest. Consider particles with radius r_0 and number density n circulating in the bloodstream. The average spacing between these particles in free space can be estimated as $d_{fs} = (1/n)^{1/3}$. However, the average minimum distance $\langle$d$\rangle$ between particles in the brain vascular network varies significantly as cells travel from major vessels of a diameter over 100 µm to capillaries with a 4 µm caliber.

B.1. Average minimum distance between non-interacting particles in a cylinder. Consider a cylinder of length L ($L >> R$) and radius R containing N uniformly distributed non-interacting particles. If $d_{fs} >> R$, particles are essentially distributed along a line, and therefore

$$(d_{fs})^3 = \pi R^2 l, \quad [33]$$

where l is the average distance between particles along the axial direction of the cylinder. In this case, the probability of finding a particle near the origin ($x \in [-dx/2, dx/2]$) and $(N-1)$ particles at least distance $|x|$ away from the origin is given by:

$$P_N(x)dx = N\frac{dx}{L}\left(1 - \frac{2|x|}{L}\right)^{N-1} = \frac{dx}{l}\left(1 - \frac{2|x|}{Nl}\right)^{N-1} \simeq \frac{1}{l}\exp\left(-2\frac{|x|}{l}\right)dx, \quad [34]$$

where we have used $l = L/N$ and taken the thermal dynamic limit. The average minimum distance between particles is then

$$\langle d \rangle = \int_{-\infty}^{\infty} \frac{|x|}{l}\exp\left(-2\frac{|x|}{l}\right)\, dx = \frac{l}{2}, \quad [35]$$

and therefore

$$\frac{\langle d \rangle}{d_{fs}} = \frac{2}{\pi}\left(\frac{d_{fs}}{2R}\right)^2. \quad [36]$$

If $d_{fs} << R$, the geometry of the cylinder can be ignored. For non-interacting particles in 3D free space, the average minimum distance $\langle d \rangle$ is given by:

$$\frac{\langle d \rangle}{d_{fs}} = \frac{1}{3}\Gamma\left(\frac{1}{3}\right)\frac{a}{d} = \frac{1}{3}\Gamma\left(\frac{1}{3}\right)\left(\frac{3}{4\pi}\right)^{1/3} \approx 0.556, \quad [37]$$

where a is the Wigner-Seitz radius.

B.2. Monte Carlo simulation of hard spheres in a cylinder. To characterize this variation, we performed Monte Carlo simulations of hard sphere packing in a cylinder. These simulations allowed us to compute how $\langle d \rangle$ depends on vessel geometry and tracer density. For $d_{fs} \geq 2R$, where R is the vessel radius, the scaling relation closely follows the 1D analytical model for non-interacting particles (Equation 36). For $d_{fs} < 2R$, the normalized spacing $\langle d \rangle / d_{fs}$ is bounded between 0.56 (Equation 37) and about 0.7 (Figure S2A).

B.3. Estimates of average cell spacing $\langle d \rangle$**.** Let H be the hematocrit, γ be the blood cell labeling ratio, and r_0 be the blood cell radius. The average distance between cells in 3D free space is

$$d_{fs} = \left(\frac{4\pi}{3\gamma H}\right)^{1/3} r_0. \quad [38]$$

For $H = 0.45$, $r = 2\ \mu m$ and a labeling fraction γ of 10%, d_{fs} is 9.1 µm. Labeling 0.1% of the cell increase d_{fs} to 42.1 µm. These estimations, combined with Equation 36 and Equation 37, suggest that as RBCs traverse the vascular network, their average minimum distance $\langle d \rangle$ can vary by orders of magnitude. The narrowest spacing occurs in the fastest-flowing segments (Figure S2B). As a result, at a 70 Hz acquisition rate and 10% labeling fraction, the conventional tracking algorithm fails to reliably measure blood flow velocity in most arteries and arterioles, where flow speeds readily exceed 1 mm/s (Figure S2C). This highlights the need to incorporate motion-model-based approaches for blood cell tracking.

C. Tracer density limits measurements frequency. While large spacing between tracers simplifies computational analysis, it also reduces particle detection frequency. The measurement frequency, ν_m, is given by:

$$\nu_m = \frac{v}{\langle d \rangle}, \quad [39]$$

where v is the flow speed in the vessel. Low detection frequency can hinder systematic measurement of transient hemodynamics in the microvessels. Importantly, this frequency is limited by the minimum flow speed in the vessels of interest and is independent of the imaging acquisition rate (Figure S2D). Thus, the maximum target flow velocity v_{max} and the minimum target measurement frequency ν_d collectively determine the tracer density in the experiment.

3. Computational methods

A. Cortical vascular network reconstruction. We utilized a subset of whole mouse brain vascular connectomes in our analysis. These connectomes are spatial graph representations of the entire mouse brain vascular network, containing vessel centerlines at 1 µm resolution, microvessel radius estimated at 0.25 µm resolution, and the connectivity data between over six million branches. The reconstruction procedure is detailed in (1). For this study, we selected a $(1072\ \mu m)^3$ volume of the vascular network near the somatosensory cortex and manually proofread vessel connectivity, with particular attention to pial vessels. We then traced the large pial vessels back to major surface vessels with known anatomical identities in multiscale downsampled whole brain image volumes and labeled them as arterioles or venules. These labels were computationally propagated to connected branches with radii greater than 3.5 µm, while branches with radii smaller than 3.5 µm were defined as capillaries. Using this semi-automated procedure, we classified 18,973 vessel branches into capillaries, arterioles, and venules. A three-dimensional reconstruction at 1 µm resolution of this network is shown in Figure S5. For the resistor network analysis, we constructed an undirected graph from the spatial graph after pruning all microvessel branches with unconnected endpoints. The resulting graph contains 14,091 capillary branches, 162 arterial branches, and 657 venous branches.

B. Confocal light field datasets. Our analyses used five datasets from the red blood cell tracking experiments previously described(2). Each dataset consisted of 1,258 3D volumetric image stacks acquired at 70 Hz, with each stack containing $512 \times 512 \times 100$ voxels at a voxel resolution of $2 \times 2 \times 2.5\ \mu m^3$.

C. Spatial graph reconstruction. We computed the structural image of the vascular in each experiment by taking the maximum value of each voxel over all 1,258 volumes. The resulting image volumes were cropped and masked to clean up the out-of-focus light field reconstruction artifacts. We then removed shot noise using a 3D median filter, stretched image contrast, resized the image to 2 µm isotropic voxel size, and computed vessel segmentation using a combination of filters and adaptive thresholding(1). We then stitched both the image and vessel segmentation using the transnational displacement vector computed from pair-wise intensity-based 3D registration. Using *itk-snap*(3), we manually refined image segmentation and further labeled arterioles and venules based on pial vascular anatomy and the recorded flow directions. This led to a labeled array where arterioles, venules, and capillaries are presented with different integers. Using this labeled array, we constructed two masks, one for arterioles and capillaries, and the other for venules and capillaries. For each mask, we filled the small holes and removed small connected components before computing the skeleton and the distance transform field. Using the distance transform field, we refined the positions of skeleton voxels. The skeletons from both masks were merged to construct the spatial graph and further iteratively refined by removing short unconnected branches and small loops(1).

D. Cell detection. To remove low-frequency, out-of-focus light field reconstruction artifacts, we apply a Difference of Gaussian (DoG) filter to each 3D volume. Since brain vasculature typically only accounts for less than 5% of the volume and only about 10% of the RBCs were labeled in the experiment, we assume that 95% of the voxels represent background and computed their mean μ_{bg} and standard deviation σ_{bg}. Local intensity maxima are then detected across the filtered volume with a minimum inter-peak distance of 6 µm and intensity thresholds set to at least 5 σ_{bg} above μ_{bg}. To further eliminate noise-induced detections, the identified peaks are sorted by intensity in descending order, and the intensity differences between the consecutive peaks are computed. This difference sequence is smoothed using a moving average filter with a window size of 10. We then retain only peaks for which the smoothed intensity difference remains above a threshold of 1, based on the assumption that noise-induced peaks would cluster around similar intensity levels. For each remaining valid peak, we compute a set of intensity features from the original (unfiltered volume). These features include the peak intensity, signal-to-noise ratio based on local neighborhood statistics, and intensity-weighted center-of-mass position. Additionally, the local spatial intensity profile around each peak is characterized by computing the covariance matrix of voxel intensity within the neighborhood, followed by eigenvalue decomposition to extract the shape and orientation.

E. Network flow tracker (NFT). For each dataset, we first reconstructed the local vascular network within the imaged volume. Cell detections located far from the reconstructed mask were removed, and the remaining detections were assigned to their nearest skeleton voxel. Cells appearing stationary were identified and labeled as "tracked". The remaining detections were used to generate detection maps for each edge in the spatial graph. An iterative, spatiotemporal-correlation-based algorithm was applied to estimate the average flow velocity along each edge. In parallel, a graph-constrained tracking algorithm estimated velocity in microvessels where the flow is slow. These two independent velocity estimates were then combined to initialize the graph-constrained predictive tracking algorithm, which linked cells across frames by minimizing position prediction error and intensity feature discrepancy. The resulting trajectories were used to iteratively update the motion model and improve tracking accuracy. A detailed description of each step is provided in the following sections.

E.1. Voxel–to–Skeleton Mapping. We implemented a custom nearest–skeleton assignment procedure to map every voxel in the imaged volume to its closest skeleton voxel while minimizing erroneous arteriole/venule cross assignments in regions where large vessels run in close apposition. To avoid a single global distance transform that can bias the assignment of small vessels bordering a large vessel of another class, we computed two 3D Euclidean distance transforms (EDTs) with index returns: (i) over the union of arterioles and capillaries and (ii) over the union of venules and capillaries. Each transform yields, for every voxel inside the respective mask, the index of the nearest skeleton voxel belonging to that mask subset. These partial nearest–index fields were then merged: arterial/capillary voxels receive assignments from the first transform and venous/capillary voxels from

the second, ensuring that an arteriole (venule) is never "pulled" toward a nearby venule (arteriole) solely by absolute distance when both are present.

For voxels outside the vascular mask (background), we first computed their nearest mask voxel via an EDT of the background; each such mask voxel already had a skeleton assignment from the merged arterial/venous fields, which we propagated back to the background voxel. We then converted all stored nearest skeleton indices into per-voxel Euclidean distances between each voxel coordinate and its assigned skeleton coordinate, producing a dense distance map and corresponding nearest-skeleton coordinate arrays. The resulting data structure supports efficient repeated queries (index or coordinate to nearest skeleton index, class label, or distance) required for downstream graph and path length analyses.

E.2. Efficient shortest-path search on large spatial graphs. We developed a data structure for efficient shortest-path search between arbitrary skeleton voxel indices in the spatial graph. For each branch in the voxel-level spatial graph, we ordered skeleton voxels from one end to another and computed the geodesic distance between the centroids of the connected nodes. These lengths become edge weights in a *NetworkX* (4) undirected multigraph. A distance query between two voxel indices first tests whether both voxels lie on the same edge; if so, the path length is the accumulated Euclidean distance along the pre-ordered coordinates between their positions (direction recorded as ± 1). Otherwise, each voxel is mapped either to an existing node or, if it lies inside an edge, that edge is virtually split at the voxel by introducing a transient "dummy" node: we remove the original edge and insert two new weighted sub-edges whose lengths are computed from the cumulative inter-voxel distances. With both query voxels thus associated with nodes, we compute the weighted shortest path (Dijkstra) and then restore any modified edges, removing dummy nodes. Optional caching stores previously computed path lengths and full node paths for reuse. Additional routines reconstruct the voxel-level path (and per-voxel direction signs) from a node path, derive edge traversals, and report path direction relative to a chosen start node. Compared to performing each query by constructing a sparse voxel adjacency and running voxel-level Dijkstra, this data structure enabled > 100 times acceleration in shortest-path queries in our benchmarks, thereby making feasible spatial-graph–based particle tracking workflows that require millions of neighborhood searches and distance comparisons.

E.3. Iterative spatiotemporal-correlation-based flow velocity estimation. For each branch, we constructed a detection map and used an adaptive spatiotemporal correlation procedure to estimate average velocity. The detection map is a T-by-N matrix whose elements (t, i) indicate if a cell is detected at time t near the i-th skeleton voxel of the branch, where T is the number of frames and N is the number of skeleton voxels in the branch. We used an initial spatial bin size dx and temporal shift dt to compute root-mean-square-normalized, shifted pairwise correlations between binned intensity traces. Directional flow was estimated from correlations whose index offsets share a dominant sign, while their weighted offset gave the average speed. If the velocity appeared too small yet the correlation dominance was strong, the temporal shift was successively doubled to capture slow motion. If correlations are indeterminate, the bin size was doubled to improve signal to noise level. The process stopped when a valid, finite velocity is obtained or when any limits were reached.

E.4. Stationary cell detections. We identified stationary particles through a multi-stage spatiotemporal filtering and tracking procedure. First, we counted the number of times cells were detected at each voxel. Voxels with counts exceeding a minimum voxel-detection threshold of 70 were marked "high-count." A 3-D binary mask of these voxels was dilated with a spherical structuring element of radius 1 to join immediately adjacent high-count sites into contiguous connected components (CCs). Only detections whose voxel indices lay inside this dilated mask were retained and temporally ordered, and their positions were linked into preliminary trajectories using the *trackpy* package (5), a frame-to-frame particle tracking algorithm constrained by a maximal allowed displacement. Resulting trajectories with fewer than 5 time points were discarded. Each remaining trajectory was then assigned to exactly one high-count CC, and CCs lacking any trajectory whose maximal span reached at least 10 frames were removed. Within each remaining CC, trajectories were temporally merged when the inter-trajectory temporal gap and spatial separation at the junction were less than 5 frames, producing longer composite particle objects. Finally, we eliminated residual short or fragmented tracks whose total temporal support did not exceed a stricter final threshold of 35 frames.

E.5. Spatial-graph-based predictive cell tracking. We modified and augmented the *trackpy* framework(5) to perform model-based particle tracking constrained to a pre-computed spatial graph, enabling accurate association of detections across frames in tortuous 3D networks.

To incorporate network constraints, we developed a data structure that maintains (1) a nearest skeleton map mapping arbitrary coordinates to the closest skeleton voxel and (2) an estimated velocity for each edge. For each particle at time t, we projected its position onto the skeleton and predicted one or more plausible next frame positions by propagating along the edge based on the estimated edge velocity. When a particle was near or passed a node, we predicted all possible downstream edges whose flow direction was consistent with local flow sign; if the particle would traverse beyond an endpoint within the time step, we recorded the endpoint position plus a signed "extra travel" distance indicating likely exit from the network.

Candidate selection was based on a nearest neighbor search near the predicted positions. For each prediction, we obtained up to k spatially proximate detections in frame $t+1$. We computed the geodesic distance along the spatial network between the prediction positions and their nearby candidate detections. For a particle moving along an edge with velocity v, we estimated the search range as:

$$\delta v = \min\{\Delta V,\ \max\{f_v v,\ \Delta V_{min}\}\}, \qquad [40]$$

where ΔV and ΔV_{min} are the maximum and minimum search range, respectively, and $f_v > 0$ was used to estimate the search range based on edge velocity. Candidates whose geodesic distance exceeded the search range δv were rejected. If a predicted position lay just outside the network, its remaining distance to the exit endpoint contributed to an extra cost term, while overly distant exit projections were discarded.

In addition, we integrated feature similarity cost for detection linking. For user specified features (e.g., peak intensity), we computed a probabilistic similarity score s for each candidate pair, then compute the feature similarity cost as $\lambda(1-s)$, where λ is the maximum cost due to feature discrepancy. We added this cost to the geodesic distance before solving the linear assignment problem.

F. RBC partition probability at diverging nodes. For each diverging node, we identified tracked cells that moved from the inflow branch into one of the outflow branches. For each such cell, we recorded the last time it appeared in the inflow branch and the branches it subsequently entered. When flow speed was high and the immediate downstream branch was short, a cell might traverse the entire downstream branch and reach the next-nearest branch. In these cases, we determined the immediate downstream branch by finding the shortest path along the skeleton between the two tracked detection positions.

For the inflow branch, we constructed a time series $N(t)$, where the t-th element represents the number of cells that left the inflow branch at time t. Similarly, for each outflow branch i, we defined a time series $n_i(t)$ representing the number of cells that entered branch i at time t. Assuming a multinomial distribution, the maximum likelihood estimation of the overall entrance probably into branch i over the recording period is

$$p_i = \frac{\sum_t n_i(t)}{\sum_t N(t)} \quad [41]$$

with the estimated standard deviation of the mean given by

$$\sqrt{\frac{p_i(1-p_i)}{\sum_t N(t)}} \quad [42]$$

G. Phase separation fitting. To robustly estimate RBC partition probabilities, we selected degree-3 diverging nodes with at least 25 tracked cells that moved from the inflow branch into one of the outflow branches. Median RBC traversing speed was used for computing the normalized flow speed ratio. Denote the entrance probability and the flow speed in the i-th downstream branch as p_i and v_i, respectively. The preferred entrance probability in Figure 7A is defined as $\max(p_1, p_2)$ and the normalized flow speed ratio in branch i is $\kappa_i = |v_i|/(|v_1| + |v_2|)$. The flow speed bias is defined as $2(max(\kappa_1, \kappa_2) - 0.5)$. Phase separation nonlinearity, n, was determined by nonlinear least-squares fitting using the Trust Region Reflective algorithm, initialized with a value of 1.

Considering the noise level when combining data across all experiments, we applied an iterative outlier rejection procedure. After an initial fit, outliers were identified based on the percentile range of the residuals: data points with residuals exceeding 2.224 times the inter-percentile range (equivalent to about 3 standard deviations for a Gaussian distribution) were removed, and the formula was refit to the remaining data. This process was repeated until no new outliers were detected, resulting in the rejection of 10 nodes out of 416.

4. Tracking algorithms performance comparison

A. Particles in a microfluidic tube. We fabricated a microfluidic device and imaged 200-nm diameter fluorescent particles flowing through the channel driven by a syringe pump using a confocal light field microscope at 2 µm isotropic voxel size and 400 Hz for 5 s. The microfluidic devices comprised 20-µm-high, 50-µm-wide straight channels (10 mm long) molded in 5-mm-thick Polydimethylsiloxane (PDMS) and plasma-bonded to 0.17-mm glass slides. Custom PDMS chips were fabricated by Wenhao Microfluidics (Suzhou, China) using standard soft-lithography procedures (6).

We adapted the computational method described above for particle detection, localization, and feature extraction, and selected 287,782 high-signal-to-noise detections for tracking. Using the Crocker-Grier algorithm implemented in *trackpy*, we tracked particles with a maximum search range of 10 µm. Of the 91.3% detections linked across adjacent frames, 89.5% were unique matches; that is, only one particle in the next frame laid within 10 µm. The median inter-frame displacement was 1.1 µm, while the 99 percentile is 5.6 µm (corresponding to 2.2 mm/s).

The slow flow in the microfluidic tube can be well approximated as laminar from end to end, so the angle between a particle's velocity in adjacent frames should be about 0°. We therefore selected trajectories that (1) spanned at least 10 frames and (2) had a maximal angle between velocity vectors in consecutive frames $< 60°$. Because the predictive tracking algorithm assumed a constant velocity along the centerline, we further selected trajectories with $> 50\%$ of particle positions within 10 µm from the tube centerline. These criteria yielded 792 trajectories comprising 25,823 detections. For each frame, we computed the nearest neighbor distance for each particle and then took the median. Across 2000 frames, 50% of the frames had this median distance greater than 22.1 µm, and the minimal observed median is 11.4 µm. The average particle velocity is 1.0 mm/s, with a standard deviation of 0.35 mm/s, and 99.91% of the linked detections have instantaneous speed less than 1.6 mm/s. These trajectories served as the "ground truth" for benchmarking tracking algorithms.

As analyzed previously, the performance of the algorithm degrades when the prediction error becomes comparable to the inter-particle spacing. In our experimental setup, the inter-particle spacing was relatively stable. We therefore subsampled the trajectory at fixed temporal interval, applied both algorithms to the same subsampled detections with the maximum search range of 1.6 mm/s and evaluated the performances in terms of precision, recall, accuracy, and F1 score (Figure S3).

Consistent with the theoretical analysis, the Crocker-Grier algorithm started to degrade significantly as the sampling frequency dropped below (1000 µm/s) / 11.4 µm = 88 Hz. By contrast, Network Flow Tracker (NFT) remained relatively stable for trajectories downsampled to 25 - 100 Hz. NFT achieved 79% accuracy at 25 Hz, whereas attaining 81% accuracy with Crocker-Grier algorithm requires a sampling frequency of 67 Hz. The ratio of these frequencies, 25 / 67 = 0.37, is comparable to the coefficient of variation of the speed distribution, 0.35, again in line with the theoretical analysis.

B. RBC tracking in a reconstruct network. To further evaluate algorithm performance on realistic data, we simulated RBC trajectories using the vascular network structure and flow measurements measured between 100 and 250 µm below the cortical surface. Among the 945 interconnected branches, we identified 162 "in-flow" branches in which RBCs moved from an unconnected endpoint toward a connected endpoint. In each frame, the average number of RBCs entering the network through the in-flow branch i is given by:

$$r_i = \frac{v_i N_i}{l_i T} \qquad [43]$$

where v_i is the branch's average flow velocity, N_i is the total number of RBCs detected in branch i over the recording period T, and l_i is the branch length. For each in-flow branch, we modeled RBC entrances as a Poisson Process with mean r_i, and sampled each RBC's initial position uniformly along an interval on the vessel centerline, which started at the unconnected endpoint and had a length determined by the flow speed and frame rate. For each RBC, we assumed a constant intensity across the network, which was drawn from a Gaussian distribution with a mean of 10,000 and a standard deviation of 2,000.

In each simulation step, we updated particle positions to the next frame. RBC velocities in branch i were sampled from a Gaussian distribution with a mean of v_i and a standard deviation of $0.3v_i$, matching the *in vivo* coefficient of variation reported in the main text. At bifurcations where an RBC could enter multiple downstream branches, the next branch was chosen uniformly at random. We then identified particles that exited the network and removed them in the subsequent steps. Finally, we initialized new RBCs in the in-flow branches. Using these procedures, we simulated 179,424 detections across 1,500 frames.

We quantified algorithm performance on these simulated data, where each RBC's identity and trajectories were known. Beyond overall statistics, we assessed accuracy within individual branches. As CG used a single global maximum search range, its accuracy dropped sharply as the inter-frame displacement in a branch approaches this limit (Figure S4A, C). In contrast, NFT was able to accurately track cells across a wider range of velocity (Figure S4B, D). We further perturbed tracking parameters used in NFT and found the overall performances to be very stable with respect to a wide range of parameters (Table S1).

C. Appendix: Algorithm performance evaluation metrics. We evaluated the performance of algorithms by comparing the tracking results with the ground truths from either speed measurements or simulations. For each particle in frame t, the ground-truth and predicted link targets were stored as one-dimensional integer vectors

$$\mathbf{y}^{\text{pred}}, \mathbf{y}^{\text{gt}} \in \mathbb{Z}^N,$$

where the i-th element encodes the target of particle i in frame $t+1$ as follows: (i) a value ≥ 0 denotes the index of the associated particle in frame $t+1$ (continuation); (ii) -2 indicates a normal termination, i.e. the particle is predicted to exit the system; (iii) -3 indicates an abnormal termination, i.e., the particle is lost.

We then defined the following mutually exclusive outcomes:

$$\begin{aligned}
\text{TP:} &\quad y_i^{\text{pred}} \geq 0,\ y_i^{\text{gt}} \geq 0,\ y_i^{\text{pred}} = y_i^{\text{gt}}, \\
\text{False connection:} &\quad y_i^{\text{pred}} \geq 0,\ y_i^{\text{gt}} \geq 0,\ y_i^{\text{pred}} \neq y_i^{\text{gt}}, \\
\text{False positive connection:} &\quad y_i^{\text{pred}} \geq 0,\ y_i^{\text{gt}} < 0, \\
\text{FN:} &\quad \text{False connection} \,||\, \text{False positive connection} \\
\text{TN:} &\quad y_i^{\text{pred}} < 0,\ y_i^{\text{gt}} < 0, \\
\text{FN:} &\quad y_i^{\text{pred}} < 0,\ y_i^{\text{gt}} \geq 0.
\end{aligned}$$

In addition, we tracked correctness of exit events via indicators for true/false normal exits ($y_i^{\text{pred}} = y_i^{\text{gt}} = -2$ or $y_i^{\text{pred}} = -2 \neq y_i^{\text{gt}}$) and true/false abnormal exits ($y_i^{\text{pred}} = y_i^{\text{gt}} = -3$ or $y_i^{\text{pred}} = -3 \neq y_i^{\text{gt}}$).

Let $p_{\text{TP}}, p_{\text{FP}}, p_{\text{TN}}, p_{\text{FN}}$ denote the empirical frequencies (means of the corresponding indicator vectors) over all evaluated samples. We computed the following summary statistics:

$$\begin{aligned}
\text{Precision} &= \frac{p_{\text{TP}}}{p_{\text{TP}} + p_{\text{FP}}}, \\
\text{Recall} &= \frac{p_{\text{TP}}}{p_{\text{TP}} + p_{\text{FN}}}, \\
\text{Accuracy} &= \frac{p_{\text{TP}} + p_{\text{TN}}}{p_{\text{TP}} + p_{\text{TN}} + p_{\text{FP}} + p_{\text{FN}}}, \\
\text{F1} &= \frac{2 \cdot \text{Precision} \cdot \text{Recall}}{\text{Precision} + \text{Recall}},
\end{aligned}$$

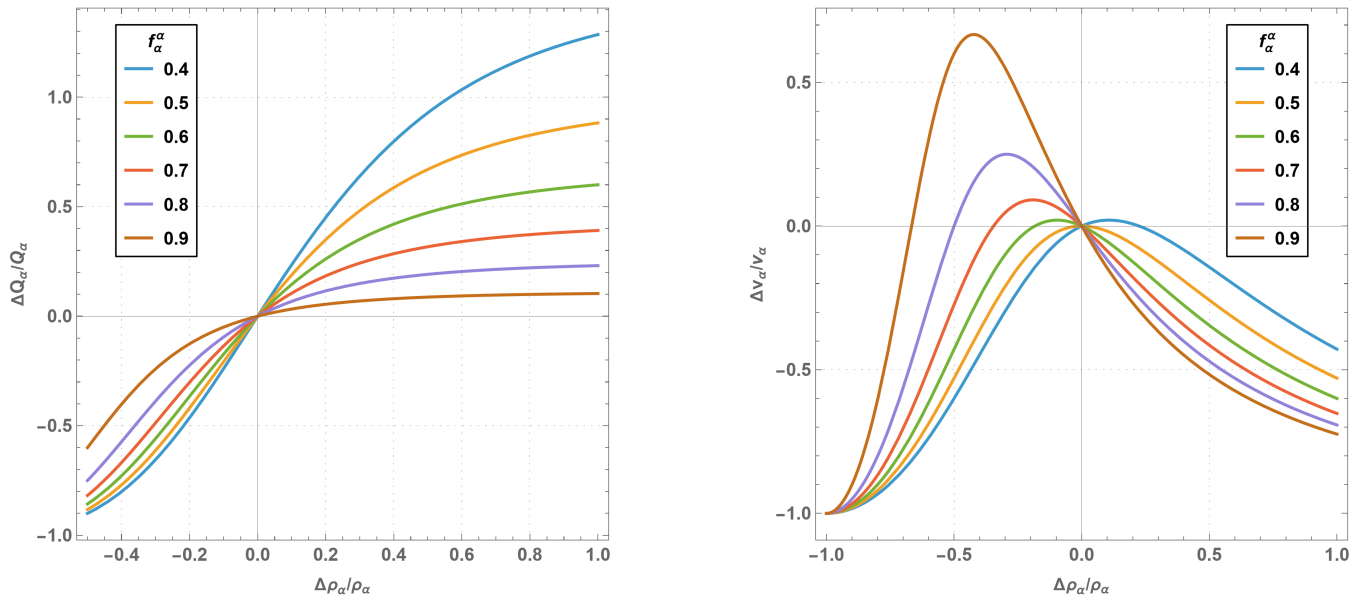

Fig. S1. Normalized flux and flow speed response to a change in radius. (A) Normalized flux change in a vessel $\Delta Q_\alpha/Q_\alpha$ in response to its own normalized radius change $\Delta v_\alpha/v_\alpha$. (B) Normalized flow speed change in a vessel $\Delta v_\alpha/v_\alpha$ in response to its own normalized radius change $\Delta\rho_\alpha/\rho_\alpha$.

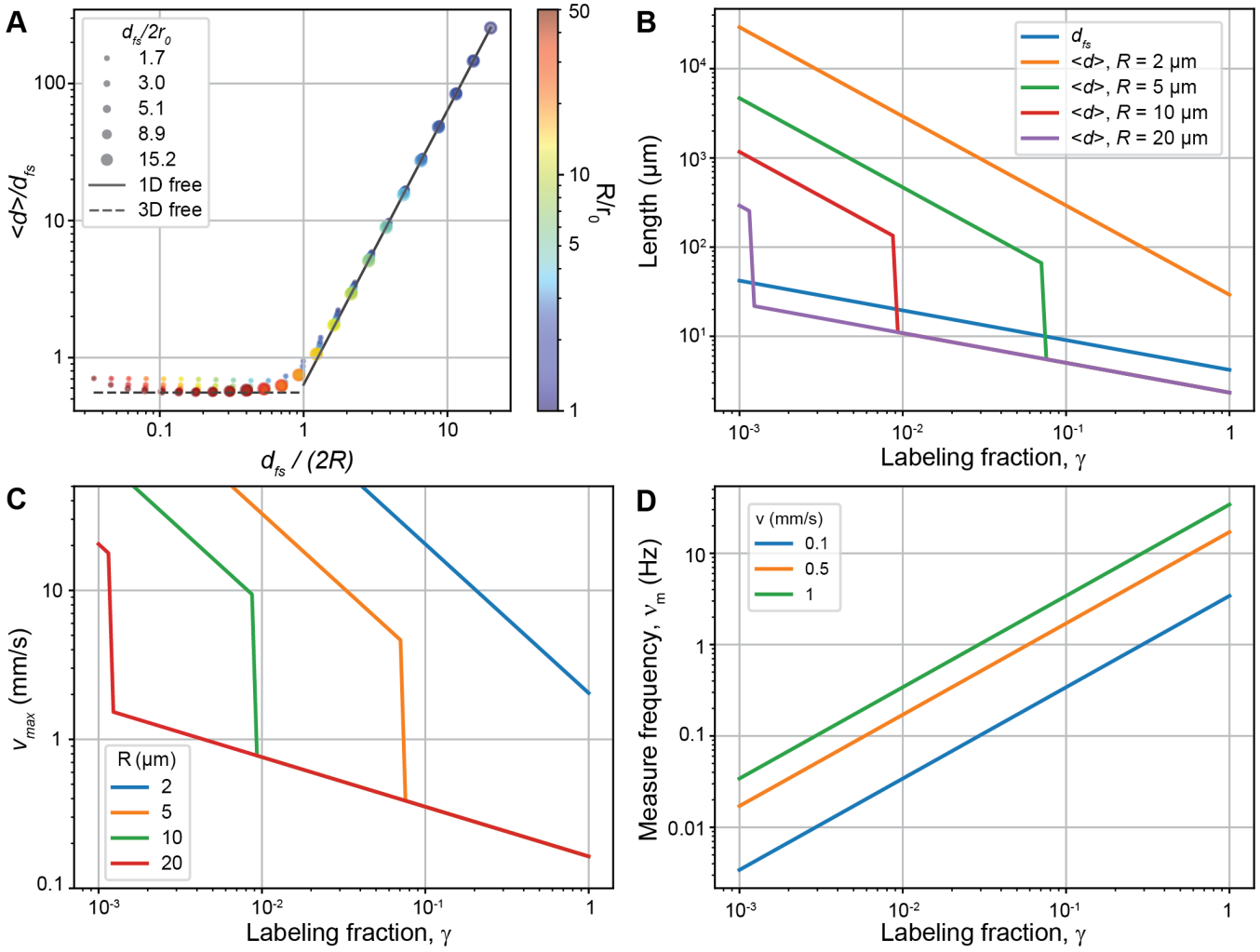

Fig. S2. Performance limits of the tracking-based approach for measuring hemodynamics in a multiscale vascular network. (A) Average normalized minimum distance between cells ($\langle d \rangle / d_{fs}$) in a cylinder versus average spacing between cells in free 3D space ($d_{fs}/(2R)$). Dots are Monte Carlo simulation results for hard spherical particles of radius r_0 in a sufficiently long cylinder, color-coded by the ratio between cylinder radius R and particle radius r_0. The solid black line represents the scaling relation for non-interacting particles in 1D. The dashed black line represents the value for non-interacting particles in 3D. (B) Lengths of d_{fs} and $\langle d \rangle$ in vessels of different diameters for different labeling fractions, γ. (C) Estimated maximum cell speed that can be reliably tracked by the conventional algorithm ($\epsilon_n = 1$) versus the labeling fraction γ in vessels of different radius R. Estimations are based on an acquisition frequency of 70 Hz. (D) Estimated measurement frequency ν_m versus labeling fraction γ in a microvessel of different flow speed.

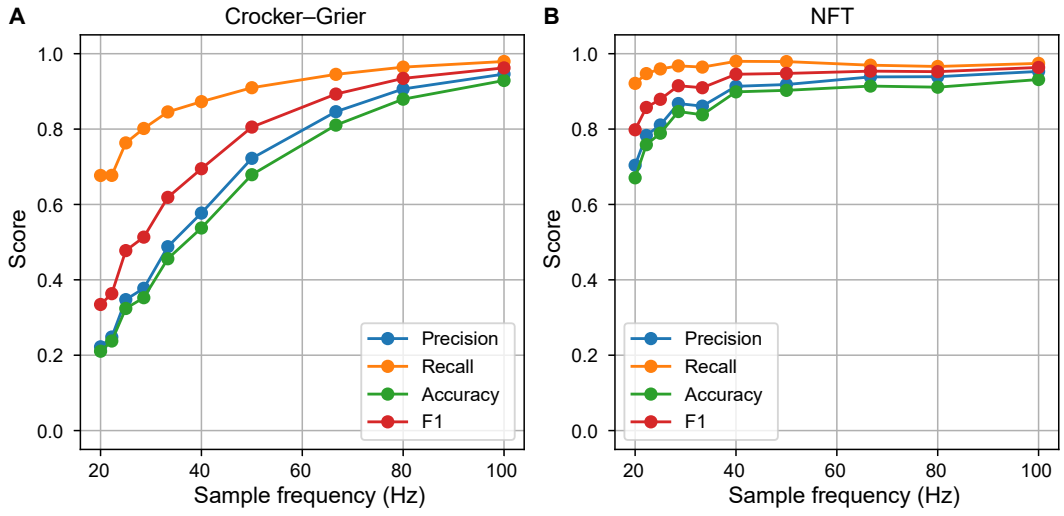

Fig. S3. Quantification of particle-tracking performance in a microfluidic channel. Particles were tracked at 400 Hz using the Crocker-Grier algorithm. The resulting trajectories were uniformly subsampled at varies frequencies and used as ground truths to evaluate tracking performance for the Crocker-Grier algorithm (A) and the Network Flow Tracker (B).

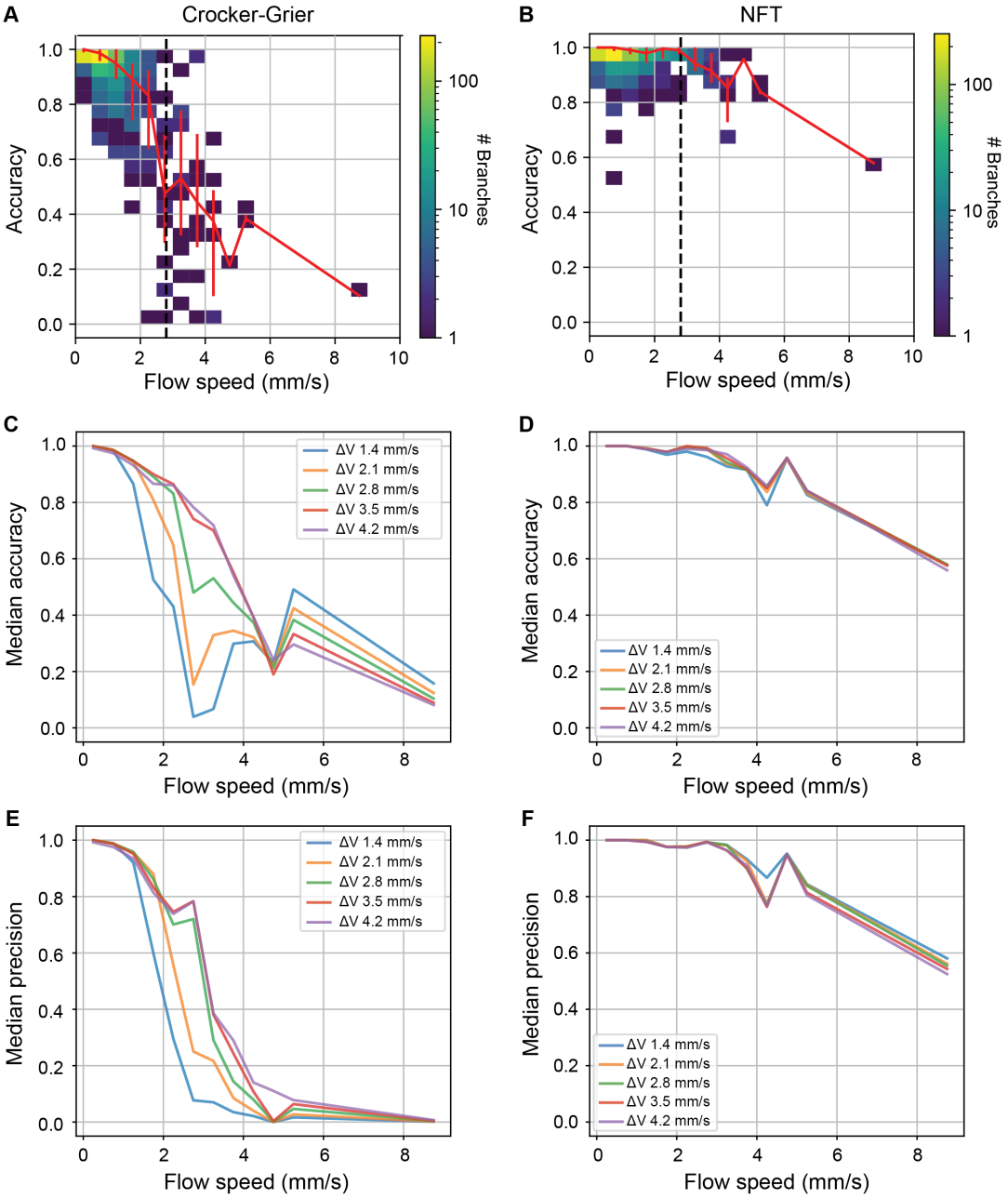

Fig. S4. Quantification of tracking accuracy and precision in vessel branches across a range of flow speeds using a simulated dataset. (A) Joint distribution of tracking accuracy and flow speed for the Crocker-Grier algorithm (CG). The dotted vertical line at 2.8 mm/s indicates the maximum search range ΔV used by the tracking algorithm. Red curve represents the median accuracy within each flow-speed bin; error bars indicate the interquartile range. Overall accuracy: 0.817. (B) Same as (A), but for the Network Flow Tracker (NFT). Overall accuracy: 0.952. (C, D) Median accuracy of CG and NFT using different search ranges. (E, F) Median precision of CG and NFT using different search ranges.

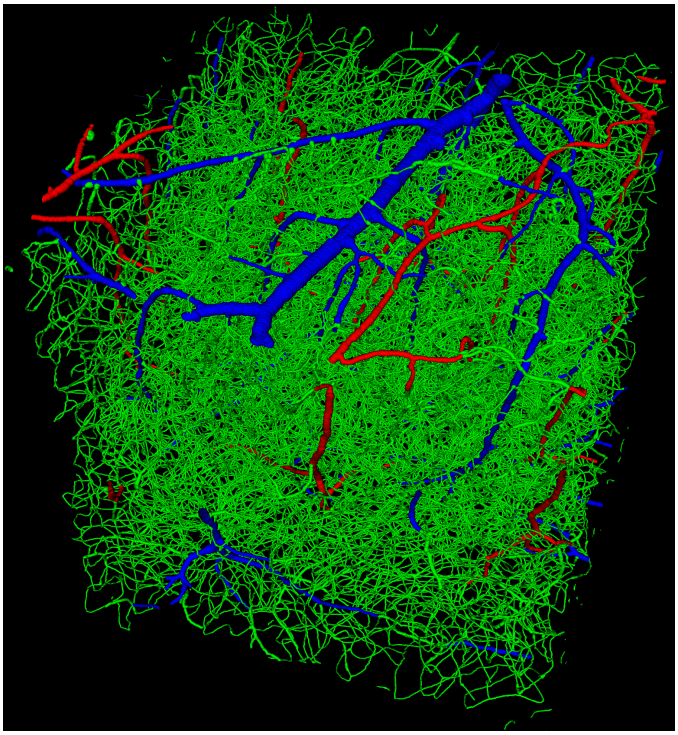

Fig. S5. Three-dimensional rendering of the reconstructed mouse brain cortical vascular network in a $(1,072\ \mu m)^3$ volume. Red: arterioles; blue: venules; green: capillaries.

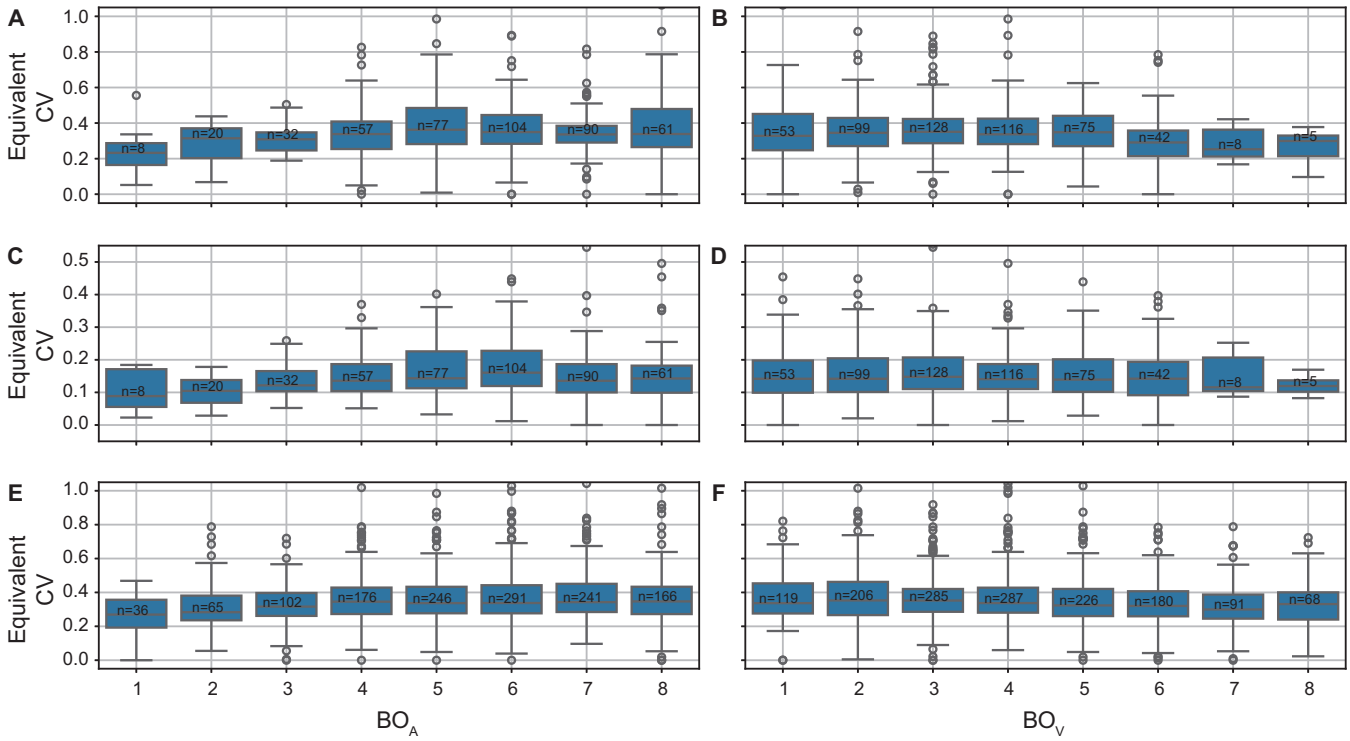

Fig. S6. Equivalent coefficient of variation (ECV) of blood flow in microvessels grouped by branch orders. (A) ECV of blood flow measured at 70 Hz in microvessels located 100 - 250 µm below the cortical surface, grouped by geodesic distance (branch order) to the nearest penetrating arteriole (BO_A). The number above the median line indicates the number of microvessels in each group. (B) Same as A, but microvessels are grouped by geodesic distance to the nearest penetrating venule (BO_V). (C) Same as A, but showing ECV of blood flow fluctuations below 2 Hz. (D) Same as B, but showing ECV of blood flow fluctuations below 2 Hz. (E) Same as A, but includes all microvessels up to 550 µm below the cortical surface. (F) Same as B, but includes all microvessels up to 550 µm below the cortical surface.

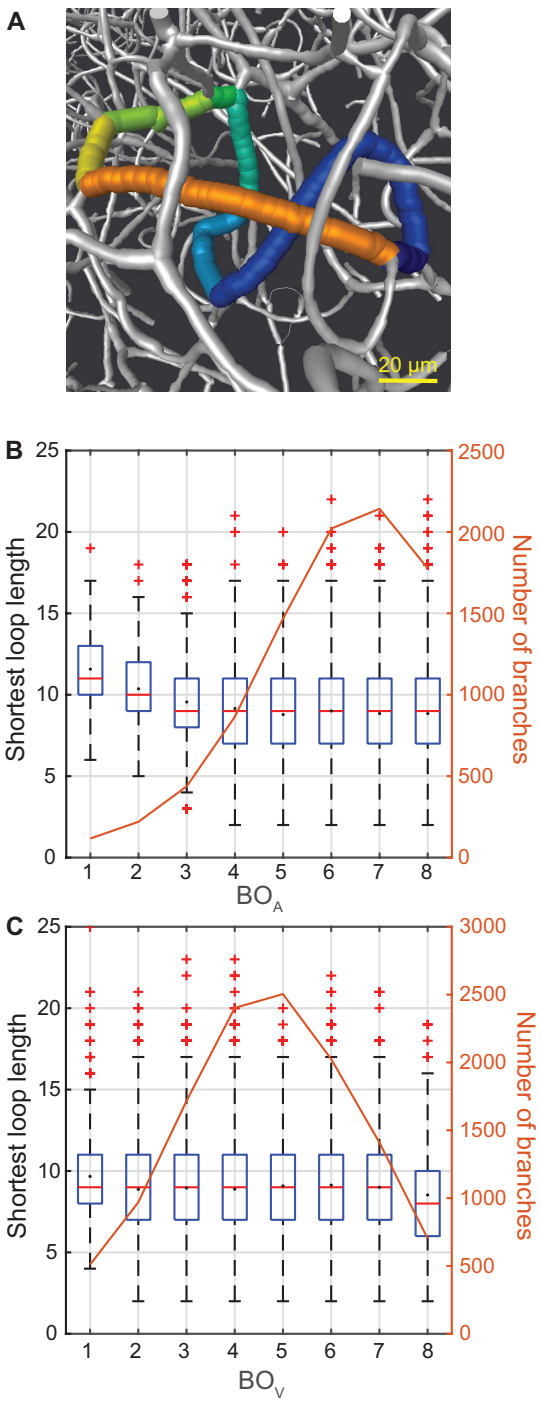

Fig. S7. Statistics of the shortest loop length for microvascular branches across different branch orders from penetrating arterioles and venules computed from the vascular network shown in Figure S5. For each branch, the shortest loop length is defined as the minimum number of branches forming a loop within the local network that contains the given branch. (A) A loop that consists of 8 branches in a reconstructed vascular network described in (7). (B) Left: Distributions for 9,049 microvessel branches grouped by branch order from arterioles (BO_A). Right: the number of branches in each BO_A group. Each whisker plot shows the minimum, 25th percentile, median (red horizontal bar), mean (black dot), 75th percentile, and maximum. Red "+" symbols indicate outliers. (C) Same as panel (B), but for 12,238 microvessel branches within 8 branch orders from venules (BO_V).

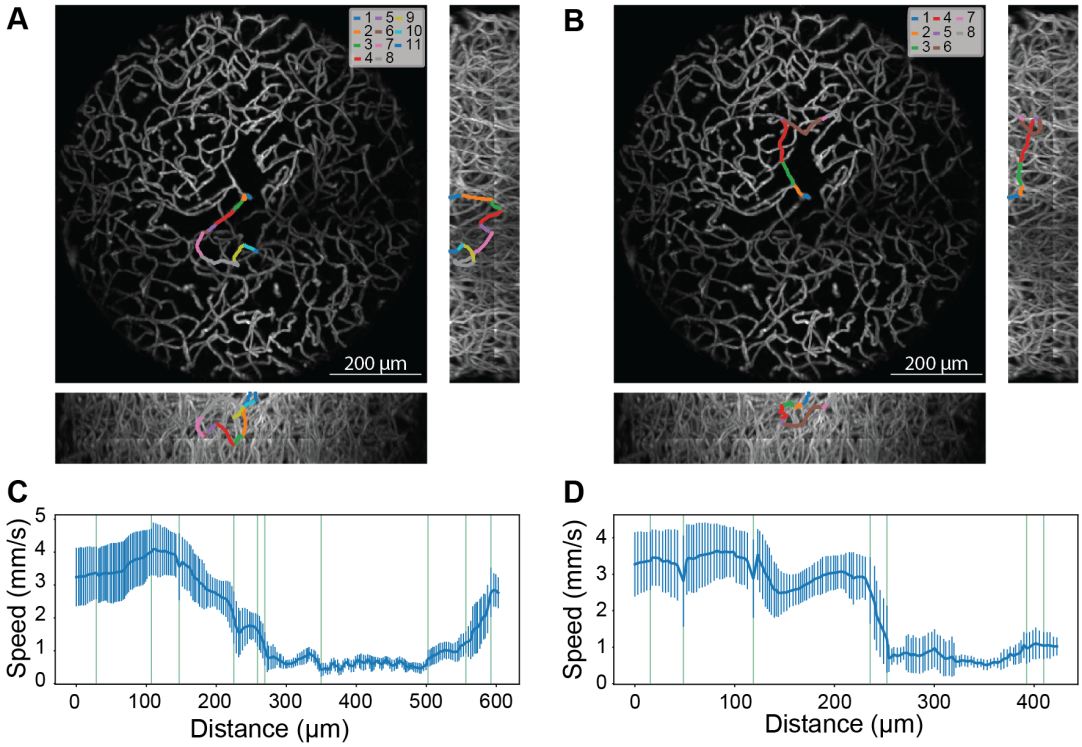

Fig. S8. Spatial flow speed variation along cell trajectories. (A) A vessel path connected a penetrating arteriole to a penetrating venule, overlaid on the maximum intensity projections of the vascular image. Individual branches are color-coded. (B) Same as (A), but for a path toward the capillaries in the deeper layer. (C) Flow speed (mean ± SD) along the path visualized in panel (A). Green vertical lines represent the branch points. (D) Same as (C), but for the path in (B).

Table S1. Tracking algorithm performance comparison. Detailed performance quantification of the Network Flow Tracker (NFT) and the Crocker-Grier algorithm (CG) applied to a simulated dataset. ΔV is the maximum search range. For NFT, the actual search range for each particle was computed dynamically based on ΔV, ΔV_{min}, f_v, and the estimated speeds in the branches. λ is the maximum cost for feature discrepancy. The set of parameters in blue was used for the results described in the main text.

Algorithm	# Detections	Tracking parameters					True positive fraction	False connection fraction	False positive connection fraction	False positive fraction	True negative fraction	False negative fraction	True normal exit fraction	False normal exit fraction	True abnormal exit fraction	False abnormal exit fraction	Precision	Recall	Accuracy	F1
		ΔV	fv	ΔVmin	Features	λ														
NFT	179424	10	0.75	10	peak_int	10	0.836	0.017	0.005	0.023	0.113	0.028	0.098	0.022	0.014	0.007	0.974	0.967	0.949	0.970
		15	0.75	10	peak_int	10	0.841	0.018	0.008	0.026	0.111	0.022	0.097	0.021	0.013	0.002	0.970	0.974	0.952	0.972
		20	0.25	10	peak_int	10	0.841	0.018	0.009	0.027	0.109	0.023	0.096	0.02	0.013	0.003	0.969	0.974	0.950	0.971
		20	0.5	10	peak_int	10	0.841	0.018	0.009	0.028	0.109	0.022	0.096	0.02	0.012	0.002	0.968	0.974	0.950	0.971
		20	0.75	5	peak_int	10	0.841	0.019	0.01	0.029	0.108	0.021	0.096	0.02	0.012	0.002	0.967	0.975	0.950	0.971
		20	0.75	10	N.A.	10	0.824	0.036	0.011	0.046	0.107	0.022	0.095	0.021	0.012	0.001	0.947	0.974	0.931	0.960
		20	0.75	10	peak_int	10	0.842	0.019	0.01	0.029	0.108	0.022	0.096	0.021	0.012	0.001	0.967	0.975	0.950	0.971
		20	0.75	15	peak_int	10	0.841	0.018	0.01	0.028	0.108	0.022	0.096	0.021	0.012	0.001	0.967	0.975	0.950	0.971
		20	1.00	10	peak_int	10	0.841	0.019	0.01	0.029	0.108	0.021	0.096	0.021	0.012	0.001	0.967	0.975	0.950	0.971
		25	0.75	10	peak_int	10	0.841	0.019	0.012	0.031	0.106	0.022	0.095	0.021	0.011	0.001	0.965	0.975	0.948	0.970
		30	0.75	10	peak_int	10	0.841	0.019	0.014	0.033	0.105	0.021	0.094	0.021	0.01	0.001	0.962	0.975	0.946	0.969
		20	0.75	10	peak_int	5	0.839	0.021	0.01	0.032	0.108	0.022	0.096	0.021	0.012	0.001	0.964	0.975	0.947	0.969
		20	0.75	10	peak_int	15	0.843	0.017	0.01	0.027	0.109	0.021	0.096	0.021	0.012	0.001	0.969	0.975	0.952	0.972
CG		10	N.A.	N.A.	N.A.	N.A.	0.653	0.094	0.032	0.125	0.086	0.136	N.A.	N.A.	0.013	0.209	0.839	0.828	0.739	0.833
		15					0.713	0.1	0.039	0.139	0.079	0.069			0.012	0.136	0.837	0.912	0.792	0.873
		20					0.738	0.104	0.044	0.148	0.074	0.04			0.011	0.104	0.833	0.948	0.811	0.887
		25					0.747	0.108	0.05	0.159	0.068	0.026			0.009	0.085	0.825	0.966	0.815	0.890
		30					0.746	0.116	0.056	0.172	0.062	0.02			0.008	0.073	0.813	0.974	0.808	0.886

References

1. X Ji, et al., Brain microvasculature has a common topology with local differences in geometry that match metabolic load. *Neuron* **109**, 1168–1187.e13 (2021).
2. Z Zhang, et al., Imaging volumetric dynamics at high speed in mouse and zebrafish brain with confocal light field microscopy. *Nat. Biotechnol.* **39**, 74–83 (2021).
3. PA Yushkevich, et al., User-guided 3D active contour segmentation of anatomical structures: Significantly improved efficiency and reliability. *NeuroImage* **31**, 1116–1128 (2006).
4. AA Hagberg, DA Schult, PJ Swart, Exploring network structure, dynamics, and function using NetworkX in *Proceedings of the 7th Python in Science Conference*, eds. G Varoquaux, T Vaught, J Millman. (Pasadena, CA USA), pp. 11–15 (2008).
5. DB Allan, T Caswell, NC Keim, CM van der Wel, RW Verweij, Soft-matter/trackpy: V0.6.4 (Zenodo) (2024).
6. Y Xia, GM Whitesides, Soft Lithography. *Annu. Rev. Mater. Res.* **28**, 153–184 (1998).
7. P Blinder, et al., The cortical angiome: An interconnected vascular network with noncolumnar patterns of blood flow. *Nat. Neurosci.* **16**, 889–897 (2013).