Supplementary Information

Methods

1 Single Particles Trajectories analysis

When not stated otherwise, the following analysis was performed using MATLAB version 9.0 (MathWorks).

1.1 Mean Squared Displacement and first moment analysis

The Mean Squared Displacement (MSD) approach has been used to detect deviations from classical free diffusive motion of SPTs. Indeed, the MSD at time lag τ is defined by

$$MSD(\tau) = \langle |\boldsymbol{X}(t+\tau) - \boldsymbol{X}(t)|^2 \rangle, \tag{1}$$

where the average is computed either over realizations or time. This quantity is expected to grow like $MSD(\tau) = A\tau^{\alpha}$ where A is a coefficient and α is called the anomalous exponent. An exponent $\alpha > 1$ is a signature of a super-diffusive regime while $\alpha < 1$ indicates a sub-diffusive regime [1]. This analysis however does not provide any explanation for the mechanisms causing a specific regime because it is usually computed by averaging over time (time-averaged MSD) or space (trajectory-averaged MSD) thus disregarding the possible heterogeneous geometrical organization of STPs.

The motion exhibited by the recorded trajectories analyzed in Fig.2 is neither time nor space homogeneous. Indeed, the motion in nodes and tubules are different and trajectories exhibit high-velocity peaks of random duration (Fig.5d left), separated by random time intervals (Fig.5d right), making the MSD analysis hard to perform on these data. For that reason, we have chosen to conduct here an analysis based on the stochastic equation of motion (15) through which local parameters of motion are extracted from many redundant trajectories. This analysis allows to recover the field of force acting on the particles at each location from the first order moment of the individual trajectory displacements (16).

It is also possible to draw the same conclusions about the observed dynamics by conducting an MSD analysis on specific subset of trajectories.

For this analysis we relied on the time averaged MSD (taMSD) at time lag $\tau = 1 \ldots (M_i - 1)$ defined, for a trajectory X_i composed of M_i points, as:

$$taMSD_i(\tau) = \frac{1}{M_i - \tau} \sum_{k=1}^{M_i - \tau} \left(X_i(t_{k+\tau}) - X_i(t_k)\right)^2 \tag{2}$$

To recover the anomalous exponent from a taMSD curve, we fitted to the first 8 points of this curve in the log-log space the function:

$$\log(A) + \alpha log(t) \tag{3}$$

using the `fit` function from MATLAB version 9.0 (MathWorks). Fig.S2b presents the taMSD curves in log-log space (left) and the corresponding anomalous exponents α (right) obtained by computing the taMSD on each trajectory, longer than 20 points (360ms) and visiting at least two reconstructed network nodes, from the experiment presented in Fig.2 and found a broad distribution of anomalous exponents ($0 \leq \alpha \leq 1.5$) in agreement the proposed two-states dynamics. To investigate the type of dynamics inside the nodes, we determined for each node the ensemble of N sub-trajectories $X_1 \ldots X_N$ located inside the node and obtained the node MSD by averaging:

$$taMSD(\tau) = \frac{1}{N} \sum_{i=1}^{N} taMSD_i(\tau) \tag{4}$$

In Fig.S2c we report for each node of the experiment presented in Fig.2 its taMSD curve in log-log space (left) and the corresponding distribution of anomalous exponents (right) observing exponents $\alpha < 0.8$, indicating a subdiffusive behavior. Finally, to investigate the type of dynamics outside nodes, we synchronized for each nodes, the trajectories on their exit from the node (as presented in section 1.7 and Fig.S2e) and computed an average MSD using eq.4. In Fig.S2d we present three examples of nodes for which this synchronization result in an averaged taMSD characterized by an anomalous exponent $\alpha > 1$ indicating in these cases a super-diffusive behavior.

1.2 Models for the instantaneous velocity distribution

1.2.1 Pure Diffusion model

To analyze the velocity histograms presented in Fig.2c,3a and S3, we use a two-dimensional random walk model:

$$\boldsymbol{X}(t + \Delta t) = \boldsymbol{X}(t) + \sqrt{2D\Delta t}\boldsymbol{\eta}, \tag{5}$$

where $\boldsymbol{\eta} = [\eta_1, \eta_2]$ with $\eta_1, \eta_2 \sim \mathcal{N}(0, 1)$ is a white noise. The distribution of the displacement lengths is given for $l > 0$ by [2]:

$$Pr_{\text{diff}} \left\{ \frac{||\Delta X(t)||}{\Delta t} = l \right\} = \frac{l}{\sigma_d^2} \exp\left(\frac{-l^2}{2\sigma_d^2} \right), \tag{6}$$

where $||.||$ is the Euclidean norm and

$$\sigma_d = \sqrt{\frac{2D}{\Delta t}}. \tag{7}$$

We use eq. 5, with the parameters presented in Table 1, using the apparent diffusion coefficient $D = D_{app}$ defined in section 1.4.2 to generate the distributions shown in Fig.2c, Fig.3a and S3 (solid lines).

1.2.2 Flow-diffusion switching model

To account for the fast (faster than the acquisition time $\Delta t = 18$ms) internode dynamics observed in Fig.2c, we use a jump-diffusion model [3] defined by the following rule

$$\boldsymbol{X}(t + \Delta t) \quad = \boldsymbol{X}(t) + \begin{cases} \boldsymbol{J}\Delta t & \text{w. p. } \lambda\Delta t \\ \\ \sqrt{2D\Delta t}\boldsymbol{\eta} & \text{w. p. } 1 - \lambda\Delta t \end{cases}, \tag{8}$$

where the statistics of the jumps $\boldsymbol{J}$ is approximated from the observed flow velocity distribution, presented in the inset histogram of Fig.2g and in Fig.S3 as $||\boldsymbol{J}|| \sim \mathcal{N}(\mu_{jump}, \sigma_{jump})$. Although the jump angle should follow the ER architecture, for the present model, we draw the angles from a uniform distribution in $[0, 2\pi]$. This simplification holds true as long as we are considering only the norm of the jumps. To estimate the distribution of displacement lengths of process 8, we use Bayes' law and condition the displacement on each state of the process:

$$\begin{aligned} Pr_{\text{switch}} \left\{ \frac{||\Delta \boldsymbol{X}(t)||}{\Delta t} = l \right\} = Pr \left\{ \frac{||\Delta \boldsymbol{X}(t)||}{\Delta t} = l|\text{Jump} \right\} Pr\{\text{Jump}, t\} \\ + Pr \left\{ \frac{||\Delta \boldsymbol{X}(t)||}{\Delta t} = l|\text{Diff} \right\} Pr\{\text{Diff}, t\}. \end{aligned} \tag{9}$$

By definition the steady-state probability of being in a jump or diffusion state are:

$$\begin{aligned} Pr\{\text{Jump}\} &= \lim_{t\to\infty} Pr\{\text{Jump}, t\} = \frac{\lambda\Delta t}{1-\lambda\Delta t+\lambda\Delta t} = \kappa. \\ Pr\{\text{Diff}\} &= 1 - Pr\{\text{Jump}\} = 1-\kappa, \end{aligned} \tag{10}$$

where $\kappa = \lambda\Delta t \in [0,1]$. Approximating eq. 9 by eq. 10 and replacing the displacement length distributions by a normal distribution for the jumps and a Rayleigh distribution (eq. 6) for diffusive displacements, we obtain

$$\begin{aligned} Pr_{\text{switch}}\left\{\frac{||\Delta\boldsymbol{X}(t)||}{\Delta t} = l\right\} &= \kappa Pr\left\{\frac{||\Delta\boldsymbol{X}(t)||}{\Delta t} = l|\text{Jump}\right\} + (1-\kappa)Pr\left\{\frac{||\Delta\boldsymbol{X}(t)||}{\Delta t} = l|\text{Diff}\right\} \\ &= \kappa\frac{1}{\sigma_j\sqrt{2\pi}}\exp\left(\frac{(l-\mu_{jump})^2}{2\sigma_{jump}^2}\right) + (1-\kappa)\frac{l}{\sigma_d^2}\exp\left(\frac{-l^2}{2\sigma_d^2}\right), \end{aligned} \tag{11}$$

where σ_d is given by eq. 7. To estimate the switching probability κ in eq. 11, we use a Maximum-Likelihood Estimation (MLE) approach based on the probability

$$p(l|\kappa) = Pr_{\text{switch}}\left\{\frac{||\Delta X(t)||}{\Delta t} = l\right\}, \tag{12}$$

of observing l given κ. The MLE $\hat{\kappa}$ for N observed displacements $l_1, \ldots, l_N$ is [4]

$$\hat{\kappa} = \arg\max_{\kappa\in[0,1]} \sum_{i=1}^{N} \ln(p(l_i|\kappa)). \tag{13}$$

We compute $\hat{\kappa}$ using the `mle` function of MATLAB version 9.0 (MathWorks) applied to the trajectory displacements extracted from the SPTs described in methods.
To generate the distributions shown in Fig.2c and S3 (dashed lines), we use eq. 8, with the parameters presented in Table 1, using the diffusion coefficient in the nodes $D = D_{node}$ defined in section 1.4.3.

1.3 Reconstruction of the ER Network from SPTs

Detecting nodes (tubule junctions) boundary and inter-junction stretches (tubules) is based on the heterogeneity of the time-integrated particle spatial density presented in Fig.2d and S3. The algorithmic procedure uses the large amount of recorded SPTs described in methods and proceeds as follows. We first construct clusters of points (nodes) defined as regions of aggregation of short displacements (aggregation regions appear to co-localize with short displacements as presented in Fig.2c and S3:

1. Define a threshold V_L (in $\mu m/s$), and discard from the analysis any point $X_i(t)$ such that $\frac{||\Delta X_i(t)||}{\Delta t} \geq v_L$ ($\Delta X_i(t)$ is the displacement as defined in methods).
2. Apply the `dbscan` [5] clustering algorithm to cluster the remaining points (implementation from `scikit-learn` [6] through Python3 provided by the Anaconda Distribution version 4.3.8 (Anaconda Inc.)).
3. Approximate the boundary of each cluster as an ellipse with semi-axes $a > b$ using a principal component analysis. Remove ellipses with an area $\pi ab > 4\mu m^2$ or an eccentricity $\frac{a}{b} > 4$. Merge overlapping ellipses by fitting a new ellipse to the union of their points.
4. Assign each points discarded in step 1 to the cluster corresponding to the ellipse in which they fall, if any.

The `dbscan` algorithm used in step 2 allows to generate clusters based on the local point density and requires two parameters:

1. The maximum distance R (in μm) below which two points are considered to be neighbors.
2. The minimum number of points N at a distance $\leq R$ of a point to start a cluster.

These two parameters define a minimal density $\frac{N}{R}$ of points/μm^2 inside each cluster. The values of R and N depend on the morphology of the imaged ER and the local number of recorded trajectories. For each dataset, these values were determined empirically such that the computed clusters overlap with the ER structure formed by the trajectories.

Once nodes are found, we defined tubules by constructing a connectivity matrix C of size $K \times K$ (K number of detected nodes) where $c_{i,j}$ ($1 \leq i, j \leq K$) contains the number of trajectory displacements starting in node i and arriving in node j. Specifically, we increment the coefficient $c_{i,j}$ by one for each data point $X_k(t_l)$ ($1 \leq k \leq N_t$, $0 \leq l < M_k - 1$) in either of the following cases:

1. $X_k(t_l)$ is located in node i and $X_k(t_{l+1})$ in node j
2. $X_k(t_l)$ is located in node i, $X_k(t_{l+1})$ does not belong to any node and $X_k(t_{l+2})$ is located in node j (in this case $0 \leq l < M_k - 2$).

Finally, we removed from the graph any disconnected node. The different parameters used for reconstructing the graphs presented in the main text and supplementary data are given in Table 2. Graphs from Supplementary Table 1 (Main Text) were constructed using the parameter ranges: $V_L = [9, 19]\mu$m/s (following the histogram of instantaneous velocities), $R = [0.08, 0.26]\mu$m, $N = [10, 45]$ points and $\pi ab \leq [4, 8]\mu\text{m}^2$. Ellipses representing the nodes of the graphs are shown in Fig.2dfg and S3.

1.4 Recovery of the local dynamics in the ER lumen from SPTs

1.4.1 Langevin equation and characteristics of motion

To interpret the individual trajectories described in methods, we use the classical overdamped limit of Langevin's equation [3, 7], where the velocity is the sum of a force or a flow (drift) plus a diffusion term. For a diffusion coefficient D and a field of force $\boldsymbol{F}(\boldsymbol{X})$, the dynamics is given by

$$\dot{\boldsymbol{X}}(t) = \frac{\boldsymbol{F}(\boldsymbol{X}(t))}{\gamma} + \sqrt{2D}\,\dot{\boldsymbol{w}}(t), \tag{14}$$

where $\dot{\boldsymbol{w}}(t)$ is a vector of independent standard δ-correlated Gaussian white noises and γ is the viscosity [3, 9]. The source of the driving noise $\dot{\boldsymbol{w}}(t)$ is the thermal agitation. To interpret trajectories, we coarse-grain eq. 14 in an effective stochastic process [10, 11, 12, 14]

$$\dot{\boldsymbol{X}}(t) = \boldsymbol{b}(\boldsymbol{X}(t)) + \sqrt{2}\boldsymbol{B}_e(\boldsymbol{X}(t))\,\dot{\boldsymbol{w}}(t), \tag{15}$$

where $\boldsymbol{b}(\boldsymbol{X})$ is the empirical drift field, $\boldsymbol{B}_e(\boldsymbol{X})$ the diffusion related matrix and $\boldsymbol{D}_e(\boldsymbol{X}) = \frac{1}{2}\boldsymbol{B}_e(\boldsymbol{X})\boldsymbol{B}_e^T(\boldsymbol{X})$ (T is the transposition) is the effective diffusion tensor. This model is used to construct the empirical estimators for the first and second order moments for the diffusion and drift from trajectories. The conditional moments are computed from the trajectory displacements (see methods) [9],

$$\boldsymbol{b}(\boldsymbol{X}) = \lim_{\Delta t \to 0} \frac{\mathbb{E}[\Delta X(t)\,|\,X(t) = \boldsymbol{X}]}{\Delta t}, \quad D_e(\boldsymbol{X}) = \lim_{\Delta t \to 0} \frac{\mathbb{E}[\Delta X(t)^T \Delta X(t)\,|\,X(t) = \boldsymbol{X}]}{2\Delta t}. \tag{16}$$

Here the notation $\mathbb{E}[\cdot \mid X(t) = \boldsymbol{X}]$ means averaging over all trajectories that are located at point $\boldsymbol{X}$ at time t. The coefficients of eq. 15 can be statistically estimated from a large sample of trajectories in the neighborhood of the point $\boldsymbol{x}$ at time t. On the recorded SPTs, the expectations in eq. 16 are estimated with a time step $\Delta t = 18ms$ according to the image acquisition time.
To compute the empirical drift and diffusion coefficients, we first partition the data into square bins $B(\boldsymbol{x}_k, r)$ of center $\boldsymbol{x}_k$ and side $r = 0.2$ μm [11]. Starting from the N_t acquired projected trajectories described in methods, eq. 16 becomes for a drift vector $\boldsymbol{b}(\boldsymbol{x}) = (b_x(\boldsymbol{x}), b_y(\boldsymbol{x}))$

$$\begin{aligned}
b_x(\boldsymbol{x}_k) &\approx \frac{1}{N_k}\sum_{i=1}^{N_t} \sum_{0\leq j<M_i-1, \boldsymbol{X}_i(t_j)\in B(x_k,r)} \left(\frac{x_i(t_{j+1}) - x_i(t_j)}{\Delta t}\right) \\
b_y(\boldsymbol{x}_k) &\approx \frac{1}{N_k}\sum_{i=1}^{N_t} \sum_{0\leq j<M_i-1, \boldsymbol{X}_i(t_j)\in B(x_k,r)} \left(\frac{y_i(t_{j+1}) - y_i(t_j)}{\Delta t}\right), \qquad (17)
\end{aligned}$$

where N_k is the number of displacements starting in bin $B\left(x_k, r\right)$. Similarly, the components of the effective diffusion tensor $\boldsymbol{D}_e(\boldsymbol{x}_k)$ are approximated by the empirical sums

$$\begin{aligned}
D_{xx}(x_k) &\approx \frac{1}{N_k}\sum_{i=1}^{N_t} \sum_{0\leq j<M_i-1, \boldsymbol{X}_i(t_j)\in B(x_k,r)} \frac{(x_i(t_{j+1}) - x_i(t_j))^2}{2\Delta t} \\
D_{yy}(x_k) &\approx \frac{1}{N_k}\sum_{i=1}^{N_t} \sum_{0\leq j<M_i-1, \boldsymbol{X}_i(t_j)\in B(x_k,r)} \frac{(y_i(t_{j+1}) - y_i(t_j))^2}{2\Delta t} \qquad (18) \\
D_{xy}(x_k) &\approx \frac{1}{N_k}\sum_{i=1}^{N_t} \sum_{0\leq j<M_i-1, \boldsymbol{X}_i(t_j)\in B(x_k,r)} \frac{(y_i(t_{j+1}) - y_i(t_j))(x_i(t_{j+1}) - x_i(t_j))}{2\Delta t}.
\end{aligned}$$

By definition, the moment estimators 17 and 18 are computed by averaging the displacements $\Delta X(t)$, each displacement contributing to the bin into which its initial point $X(t)$ falls. The computation reveals that the diffusion tensor is isotropic (Fig.2f and Fig.S3). Furthermore there is no need for further deconvolution of the SPTs, as the localization noise does not contribute until second order to the diffusion tensor and drift [13]. To obtain stable estimations, we only use bins that contain at least 20 displacements. We use the same grid to compute a density map, providing an estimate of the local density of trajectories, by counting the number of displacements falling into each bin and dividing by the size of the square ($0.04\mu\mathrm{m}^2$). Density and diffusion maps are presented in Fig.2df, Fig.3bc and Fig.S3a,b.

1.4.2 Estimation of the apparent diffusion coefficient

We define the apparent diffusion coefficient D_{app} as the diffusion coefficient estimated using the entire distribution of displacements. Note that this distribution also contains large values, that cannot be attributed to diffusion. We estimated D_{app} by averaging the values obtained using eq. 18 on each bin over the entire map (only for bins containing at least 20 points). The distribution of diffusion coefficients from individual bins of the diffusion map from Fig.2f is presented in Fig.S2a, in Fig.3d for the map of Fig.3c and in Fig.S3a,b for the other datasets.

1.4.3 Estimation of the diffusion coefficients in the nodes

The diffusion coefficient D_{node} inside each node is computed from eq. 16, but in addition we constrained both ends of the displacement be located inside the node. For a node delimited by an ellipse E, we get:

$$D_{node}(\boldsymbol{X}(t)) = \lim_{\Delta t \to 0} \frac{\mathbb{E}[\Delta X(t)^T \Delta X(t) | X(t) = \boldsymbol{X} \in E \text{ and } X(t+\Delta t) \in E]}{2\Delta t}. \quad (19)$$

We use eq. 19 to estimate the diffusion coefficient D_{node} in each node and report their distributions in Fig.2f (inset) for the main text dataset and Fig.S3a,b.

1.4.4 Estimation of the internodes displacements

We define the instantaneous velocity (in $\mu m/s$) between two successive points of the same trajectory $\boldsymbol{X}(t_1)$ and $\boldsymbol{X}(t_2)$ ($t_2 > t_1$) as the ratio of the distance to elapsed time

$$v(\boldsymbol{X}(t_1), \boldsymbol{X}(t_2)) = \frac{||\boldsymbol{X}(t_2) - \boldsymbol{X}(t_1)||}{t_2 - t_1}, \quad (20)$$

where $||.||$ is the Euclidean distance. We define the flow velocity v_{flow} between two nodes A and B, as the average of the instantaneous velocities for the displacements connecting the two nodes:

$$v_{flow}(A,B) = \mathbb{E}\left[v(\boldsymbol{X}(t_1), \boldsymbol{X}(t_2)) \middle| \begin{array}{l} (\boldsymbol{X}(t_1) \in A, \boldsymbol{X}(t_2) \in B) \text{ or} \\ (\boldsymbol{X}(t_1) \in B, \boldsymbol{X}(t_2) \in A) \end{array} \right], \quad (21)$$

where by construction $t_2 - t_1 \in \{\Delta t, 2\Delta t\}$ (see tubule reconstruction from section 1.3). We discretize eq. 21 and obtain the estimator:

$$v_{flow}(A,B) \approx \frac{1}{N} \sum_{i=1}^{N} v(\boldsymbol{X}_i(t_1), \boldsymbol{X}_i(t_2)), \quad (22)$$

where N is the number of trajectory displacements connecting the nodes A, B and $\boldsymbol{X}_i$ is the i^{th} such displacements. The velocity v_{flow} characterizes the jump flow between nodes. As shown in Fig.2c, this internode flow is associated with the thick tail of the velocity distribution. The distributions of jump velocities between each pair of neighbor nodes for the different datasets are presented in Fig.2g and Fig.S3a,b.

1.5 Static ER network analysis

1.5.1 Flow directionality and strongly connected components

To determine whether the ensemble of observed displacements between two neighbor nodes i and j form a uni- or bi-directional flow, we define a uni-directionality score r as the ratio of the number of observed displacements between the two nodes with direction $i \rightarrow j$ divided by the total number of displacements in both directions:

$$r_{i,j} = \frac{c_{i,j}}{c_{i,j} + c_{j,i}}, \quad (23)$$

where C is the connectivity matrix of the graph as defined in section 1.3. Using $r_{i,j}$, we build a binary version C' of C defined as

$$c'_{i,j} = \begin{cases} 1 & \text{when } r_{i,j} \geq 0.25 \\ 0 & \text{otherwise} \end{cases}. \quad (24)$$

C' has the property that $c'_{i,j} = c'_{j,i} = 1$ when $0.25 \leq r_{i,j} \leq 0.75$ (bi-directional flow) and $c'_{i,j} = 1, c'_{j,i} = 0$ when $r_{i,j} > 0.75$ (uni-directional flow). The ratio $r_{i,j}$ (and thus $c'_{i,j}$) is computed only for pairs of nodes connected by at least three displacements ($c_{i,j} + c_{j,i} \geq 3$). To find the connected components of the reconstructed network, we use the classical Tarjan Strongly Connected Components (SCCs) detection algorithm [15] on the binary connectivity matrix C' (`graphconncomp` function from MATLAB version 9.0 (MathWorks)). SCCs are a partition of the ensemble of nodes such that there exists a path (taking into account the directionality) from each node of a subset to any other node of the same subset. The presence in Fig.4b of a SCC ecompassing almost the entire graph shows that the observed flows have the potential to move particles through almost the entire network. In addition Fig.4b also displays the flows directionality on the links as: arrows for uni-directional, solid (no arrow) for bi-directional and dashed for undecided.

1.5.2 Count of Afferent and Efferent branches

We further characterize the structure of the reconstructed ER graph by computing for each node k the number of afferent and efferent branches connected to it. An afferent (resp. efferent) branch is a link $l \to k$ (i.e. $c_{l,k} > 0$) (resp. $k \to l$, $c_{k,l} > 0$) where l is any other node of the graph. Based on the count of afferent and efferent nodes, we define the out and in-degree of node k using the connectivity matrix C as:

$$\mathrm{outdeg}(k) = \sum_{i=1}^{N} \mathbb{1}_{c_{k,i}>0} \quad \text{and} \quad \mathrm{indeg}(k) = \sum_{i=1}^{N} \mathbb{1}_{c_{i,k}>0}, \tag{25}$$

where $\mathbb{1}_{a>b} = \begin{cases} 1 & \text{if } a > b \\ 0 & \text{otherwise} \end{cases}$, and N is the number of nodes in the graph.

In- and out-degrees are computed only for nodes k such that $\sum\limits_{i=1}^{N} c_{i,k} > 2$ and $\sum\limits_{i=1}^{N} c_{k,i} > 2$ respectively. The distribution of in- and out-degrees are presented in Fig.4c.

1.5.3 Fraction of entering and exiting displacements in nodes

To study the passing dynamics of trajectories through nodes we define a retention score for the nodes defined as the ratio of the number of exiting displacements to the total number of entering and exiting displacements for a node k:

$$\phi_k = \frac{\sum_{i=1}^{N} c_{k,i}}{\sum_{i=1}^{N} c_{i,k} + \sum_{i=1}^{N} c_{k,i}}, \tag{26}$$

where C is the connectivity matrix of the graph and N the number of nodes. We have $\varphi \in [0,1]$ such that $\varphi_k \to 0$ indicates that the node retains trajectories, $\varphi_k \to 1$ indicates that trajectories originate from this node and $\varphi_k \approx 0.5$ indicates that trajectories pass through the node. φ_k was computed only for nodes k such that $\sum\limits_{i=1}^{N} c_{k,i} + c_{i,k} > 2$. The distribution of ϕ for the reconstructed ER network is shown in Fig.4d.

1.6 Transient ER network analysis

At steady-state, we identified the uni- and bi-directional flows inside tubules (Fig.4b), we now investigate how the direction of these flows evolve with time. To this end, we determine the distribution of durations of uni-directionality periods of the flow between two nodes as presented in Fig.5a. The analysis is performed on each pair or neighbouring (directly linked by trajectories) nodes, linked by at least 20 displacements registering one of the two possible directionalities (either node $A \to B$ or $B \to A$). The following algorithm groups the successive displacement events as a function of their directionality and determine the duration of these groups:

1. Collect the next displacement event observed at time t_{first}, identify its direction say $A \to B$ and form a new group containing this event.
2. Accumulate in the group created in step.1 the following jump events with direction $A \to B$ and stop when there are no more event or after encountering two events with direction $B \to A$ (in this case the event or the two successive $B \to A$ events when it occurs, are not collected). The last event considered in the group occurs at time t_{end} and always has the same direction as the group (see Fig.5a).
3. Compute the duration of the group as $\tau = t_{\text{end}} - t_{\text{first}}$.
4. Return to step 1.

We discarded groups formed by less than 3 displacements with the same direction. The distribution of the uni-directional flow durations τ is presented in Fig.5b.

To interpret the mean of this distribution, we recall that this statistics is partly contaminated by the fraction of activated particles located in nodes and the track lengths. Indeed, only photo-activated particles appear in the statistics. To recover the uni-directional duration of the fluxes from the empirical distribution, we use a model taking into account these two characteristics. The probability that the flux between nodes A and B lasts a duration $\tau_{Uni} = t$ is computed by conditioning on having k events (k particles flowing in the same direction) during that time

$$Pr\{\tau_{Uni} = t\} = \sum_{k=1}^{\infty} Pr\{\tau_1 + \ldots + \tau_k = t|k\} Pr_{Uni}\{k\}, \tag{27}$$

where the probability that k event occurs in the same direction is by symmetry,

$$Pr_{Uni}\{k\} = \frac{1}{2^k} \tag{28}$$

and τ_k is the arrival time of the k^{th} event after the first one. We consider that the distributions of arrival times are Poissonian with same rate λ, independent of the initial node A or B. We conclude that

$$Pr\{\tau_1 + \ldots + \tau_k = t|k\} = \lambda \frac{(\lambda t)^{k-1}}{(k-1)!} \exp(-\lambda t). \tag{29}$$

Computing the sum from eq. 27, we obtain for at least 2 events

$$Pr\{\tau_{Uni} = t\} = \frac{\lambda}{2} (\exp\left(-\frac{\lambda t}{2}\right) - \exp(-\lambda t)). \tag{30}$$

Eq. 30 is the probability density function when all particles are labeled and its average is $\frac{3}{2\lambda}$. When a particle is activated with probability p, which represents the steady-state fraction of labeled particles, the statistics of uni-directional flow is still given by formula 30 but with a rate $\tilde{\lambda}$. Using Bayes' law, the probability of observing a displacement event is

$$\begin{aligned} Pr\{\tau_{Uni} = t\} &= Pr\{\tau_{Uni} = t, activated\} Pr\{activated\} \\ &\quad + Pr\{\tau_{Uni} = t, notactivated\} Pr\{notactivated\}, \end{aligned} \tag{31}$$

where $Pr\{activated\} = p$ is the fraction of activated molecules. The second term is zero because we cannot see displacement events from non-activated molecules. Thus the mean unidirectional flow duration becomes

$$\langle \tau \rangle = p\bar{\tau}, \tag{32}$$

where the rates are related by $\lambda = p\tilde{\lambda}$. Considering a fraction $p = 1\%$ of activated molecules, we predict that the mean undirectional time should be around $\langle \tau \rangle = 38ms$ instead of the observed $\bar{\tau} = 3.89$s.

1.7 Instantaneous velocities along individual trajectories

To obtain statistics of the velocity fluctuations of trajectories (recorded at different times over a period of seconds) and possibly located either in tubules or nodes, we introduced a 'synchronization' procedure: the velocity fluctuations of individual trajectories were monitored starting from the last time point where the particle was in a given node prior to its exit. The velocity fluctuations following the node exit was plotted as a function of time elapsed since the synchronization event (a universal time scale for all trajectories regardless when they were detected). In details, the synchronisation event is defined for a trajectory X_i, its last recorded point inside some node A,

$$t_i^* = \max_{0 \le j < M_i - 1} \{\boldsymbol{X}_i(t_j) \in A \text{ and } \boldsymbol{X}_i(t_{j+1}) \notin A\}. \tag{33}$$

Considering the n_A trajectories going through node A, the ensemble of last points inside A is

$$X_A = \{\boldsymbol{X}_i(t_i^*) | i = 1..n_A\}. \tag{34}$$

We now divide displacements along individual trajectories into two subsets based on the starting node A and appearing at a time $\tau > 0$ after exit: $X_{A,node}(\tau)$ containing displacements inside nodes and $X_{A,tubule}(\tau)$ containing displacements connecting two nodes:

$$\begin{aligned} X_{A,node}(\tau) &= \{X_i(t_i^* + \tau) \mid X_i(t_i^*) \in X_A \text{ and } X_i(t_i^* + \tau - \Delta t) \in N \text{ and } X_i(t_i^* + \tau) \in N\} \\ X_{A,tubule}(\tau) &= \{X_i(t_i^* + \tau) \mid X_i(t_i^*) \in X_A \text{ and } X_i(t_i^* + \tau - \Delta t) \in N_1 \text{ and } X_i(t_i^* + \tau) \in N_2\}, \end{aligned}$$

where $N, N_1, N_2 \in \mathcal{N}$ the ensemble of nodes and $N_1 \neq N_2$ (N_1 or N_2 can be not a node). We computed the average velocities $v_{Inter}(\tau)$ (resp. $v_{Intra}(\tau)$) at time $\tau \geq \Delta t$ for each subset:

$$v_{Intra}(\tau) = \frac{1}{n_{A,intra}(\tau)} \sum_{\boldsymbol{X}_i(t_i^*+\tau) \in X_{A,node}(\tau)} v(\boldsymbol{X}_i(t_i^* + \tau - \Delta t), \boldsymbol{X}_i(t_i^* + \tau)) \quad (35)$$

and

$$v_{Inter}(\tau) = \frac{1}{n_{A,inter}(\tau)} \sum_{\boldsymbol{X}_i(t_i^*+\tau) \in X_{A,tubule}(\tau)} v(\boldsymbol{X}_i(t_i^* + \tau - \Delta t), \boldsymbol{X}_i(t_i^* + \tau)), \quad (36)$$

where $n_{A,intra}(\tau)$ (resp. $n_{A,inter}(\tau)$) is the number of displacements from synchronized trajectories that fall inside (resp. between two nodes) at time τ and $v(.,.)$ is the instantaneous velocity (eq. 20). The synchronization process and the intra and inter node velocities along synchronized trajectories are presented schematically in Fig.S2e. In this figure, schematic trajectories synchronized on their exit of a node `A` are presented on the left panel and their associated instantaneous velocity as a function of the time since exit τ from A are presented on the right panel. Fig.5c presents the average (line) and standard deviation (shade) of the instantaneous velocity for v_{intra} (black) and v_{inter} (red) for a selected node `A` of the main-text network for $\tau \leq 0.2$s.

1.8 Instantaneous velocity peaks duration and inter-peaks period

To further analyze the fluctuations of the instantaneous velocities along individual trajectories we proceed as in section 1.7 and synchronize the trajectories leaving a given node A. We then estimated the time spent by individual trajectories in a high velocity regime (above a threshold $v_{\text{high}} = 19\mu m/s$) as the number of successive time steps $\tau = j\Delta t$ ($j = 0\ldots$) for which

$$v(X_i(t_i^* + \tau + j\Delta t), X_i(t_i^* + \tau + (j+1)\Delta t) > v_{\text{high}}, \quad (37)$$

where X_i is a synchronized trajectory and $v(.,.)$ is the instantaneous velocity (eq. 20). The distribution of the high velocity regime durations is shown in Fig.5d and the distribution of periods between two high velocity regimes in Fig.5e, computed for trajectories synchronized for each node of the ER-network. Note that we considered only trajectories that visited at least three different network nodes.

1.9 Dynamics of tubular junctions

1.9.1 Extraction of junctions from SIM images

We extracted the features of tubular junctions from a SIM images stack (50 images) through a procedure similar to [16]: first, the contrast of the entire image stack was manually modified to make the ER network more apparent. Then each image was skeletonized, junctions were extracted from the skeleton [17] (AnalyzeSkeleton plugin, ImageJ) and only junctions covering more than 3 pixels were kept. For each image of the stack, a new grayscale image was generated where only the pixels belonging to selected junctions appear white and afterward a Gaussian blur with $\sigma = 1$px was applied to produce a single-particle like image. On this stack of grayscale images, we applied a particle detection and tracking algorithm [8] to follow junctions through successive images (Spot Detection and Tracking plugins, Icy). This procedure produces an ensemble of N trajectories $X_1 \dots X_N$ such that $X_i = X_i(t_0) \dots X_i(t_{M_i})$ and $\Delta t = t_i - t_{i-1} = 90$ms. From this ensemble, we keep only trajectories possessing at least 40 points. These trajectories are displayed in Fig.S4ab overlaid on top of the average stack image.

1.9.2 Mean Squared Displacement (MSD) analysis

We first characterize the type of diffusive motion exhibited by trajectories using a Mean Squared Displacement (MSD) analysis. For an ensemble of T trajectories the MSD at time t_k is defined as:

$$MSD(t_k) = \frac{1}{T} \sum_{i=1}^{T} ||X_i(t_k) - X_i(t_0))||^2 \approx A(t_k - t_0)^{\alpha} \tag{38}$$

where $||.||$ is the Euclidean distance, A is a coefficient and $\alpha > 0$ is the anomalous exponent characterizing the type of diffusive motion. To estimate α, we fit the first 20 points of the MSD curve to the line $y(t_k) = log(A) + \alpha log(t_k - t_0)$ in the log-log space using a MATLAB version 9.0 (MathWorks) script. Fig.S4c shows the distribution of α obtained applying this procedure to 100 independent samples of $T = 20$ randomly selected trajectories from the ensemble of trajectories possessing at least 40 points and keeping only the fits for which the coefficient of determination $R^2 > 0.75$. We found that $\alpha = 0.60 \pm 0.24$ suggesting a sub-diffusive behavior or diffusion in the presence of confinement forces [19].

1.9.3 Confined motion of tubular junctions

The distribution of instantaneous velocities (Fig.S4d) suggests to model junction's dynamics as a diffusion process confined by active forces generated by a parabolic potential well. The corresponding equation of motion is expressed as an Ornstein-Ulhenbeck stochastic process

$$\dot{X} = \kappa(X - \mu) + \sqrt{2D}\dot{\boldsymbol{w}}, \tag{39}$$

where μ is the center of the potential well, κ in $1/s$ the spring coefficient, D in $\mu\mathrm{m}^2/\mathrm{s}$ the diffusion coefficient and $\dot{\boldsymbol{w}}(t)$ is a vector of independent standard δ-correlated Gaussian white noises. We estimate for each recorded trajectory X_i the three parameters: $\hat{\mu}_i$, $\hat{\kappa}_i$ and $\hat{D}_i$. The center of the well μ_i is approximated by the center of mass of the trajectory:

$$\hat{\mu_i} = \frac{1}{M_i} \sum_{j=0}^{M_i - 1} X_i(t_j). \tag{40}$$

The parameters $\hat{\kappa}_i$ and $\hat{D}_i$ are estimated using the maximum-likelihood estimators [18]. For a trajectory $X_i(t_0) \dots X_i(t_n)$ we compute for each dimension $X_i(t) = (x_i^1(t), x_i^2(t))$, $d = 1, 2$

$$\hat{\beta}_1^d = \frac{n^{-1} \sum\limits_{k=1}^{n} x_i^d(t_k) x_i^d(t_{k-1}) - n^{-2} \sum\limits_{k=1}^{n} x_i^d(t_k) \sum\limits_{k=1}^{n} x_i^d(t_{k-1})}{n^{-1} \sum\limits_{i=1}^{n} x_i^d(t_{k-1})^2 - n^{-2} (\sum\limits_{i=1}^{n} x_i^d(t_{k-1}))^2} + \frac{4}{n}, \tag{41}$$

$$\hat{\beta}_2^d = \frac{n^{-1} \sum\limits_{k=1}^{n} (x_i^d(t_k) - \hat{\beta}_1^d x_i^d(t_{k-1}))}{1 - \hat{\beta}_1^d}, \tag{42}$$

$$\hat{\beta}_3^d = n^{-1} \sum_{k=1}^{n} (x_i^d(t_k) - \hat{\beta}_1^d x_i^d(t_{k-1}) - \hat{\beta}_2^d (1 - \hat{\beta}_1^d))^2, \tag{43}$$

from which we obtain the estimators:

$$\hat{\kappa}_i^d = \frac{\log(\hat{\beta}_1^d)}{\Delta t} \quad \text{and} \quad \hat{D}_i^d = \frac{\hat{\kappa} \hat{\beta}_3^d}{1 - (\hat{\beta}_1^d)^2} \tag{44}$$

where $\Delta t = t_k - t_{k-1}$ is the time-step and the term $\frac{4}{n}$ in $\hat{\beta}_1^d$ is a correction for the low number of points. In practice, we obtain a symmetric tensor and force field computed as the averages:

$$\hat{\kappa}_i = \frac{\hat{\kappa}_i^1 + \hat{\kappa}_i^2}{2} \quad \text{and} \quad \hat{D}_i = \frac{\hat{D}_i^1 + \hat{D}_i^2}{2}. \tag{45}$$

We apply this estimation only on trajectories possessing at least 40 points. Fig.S4e shows the distribution of the estimated spring constants $\hat{\kappa}$ and Fig.S4f the distribution of the estimated diffusion coefficients $\hat{D}$.

1.9.4 Estimating the area of confinement of tubular junctions

In this section, we define and compute the area of confinement of each junction based on the statistics of the trajectories. For each trajectory X_i we compute the 95% confidence ellipse $e_i = (c_i, a_i, b_i, \varphi_i)$ of center c_i, largest (resp. smallest) semi-axis a_i (resp. b_i) and angle (with x-axis) φ_i, of the spatial spreading of its points, considering this distribution as normal. We obtained the ellipse as follows [20]: first, we collected all points $X_i(t)$ of the trajectory into a $2 \times n$ matrix O^i, then applied a Singular Value Decomposition algorithm to the covariance matrix of O^i: $U^i \Sigma^i (V^i)^* = \operatorname{cov}(O^i)$ and finally recovered the ellipse as:

$$c_i = \frac{1}{M_i} \sum_{j=0}^{M_i - 1} X_i(t_j), \ a_i = \sqrt{5.991 \sigma_{1,1}^i}, \ b_i = \sqrt{5.991 \sigma_{2,2}^i}, \ \varphi_i = \arctan(\frac{u_{2,1}^i}{u_{1,1}^i}), \tag{46}$$

where $\sigma_{1,1}^i$ and $\sigma_{2,2}^i$ are the two eigenvalue of the matrix $\operatorname{cov}(O^i)$. We define the confinement area A_i for a trajectory X_i as the area of the estimated ellipse: $A_i = \pi a_i b_i$. The distribution of observed confinement areas is presented in Fig.S4g.

We now compare this distribution, with the expected distribution for a particle moving in a potential well. To this end for each trajectory X_i, we simulated eq. 39 with the estimated parameters $\hat{\mu}_i, \hat{\kappa}_i, \hat{D}_i$ using Euler's scheme to obtain a trajectory Y_i:

$$Y_i(t_k) = Y_i(t_{k-1}) + \hat{\kappa}_i (Y_i(t_{k-1}) - \mu_i) \delta t + \sqrt{2 \hat{D}_i \delta t \boldsymbol{\eta}}, \tag{47}$$

where $\delta t = 0.0001$s is the simulation time-step, $\boldsymbol{\eta} = [\eta_1, \eta_2]$ with $\eta_1, \eta_2 \sim \mathcal{N}(0, 1)$ is a white noise and $Y_i(t_0) = \mu_i$. To prevent the choice of the first point to influence the statistics, we run the simulation for 45000 time steps before recording the trajectory. We then sub-sampled Y_i by keeping one every 900 points to match the experimental acquisition time $\Delta t = 0.09$s and computed the confinement area of Y_i using the same procedure as for X_i. This procedure was only applied to junctions for which $\hat{\kappa}_i > 0$. The distribution of confinement areas obtained from simulated trajectories is shown in Fig.S4g.

2 Characterization of ER tubule contractions

2.1 Extraction of tubule contraction statistics

Fast SIM images of ER in live cells, acquired and reconstructed as described in methods, were rendered using the Edges look-up-table of the `Fiji` software, with contrast settings to visualise one-pixel wide boundaries of the tubules. Contraction sites were identified as such if the tubule edges bended to merge more than once at the same position. In Fig.5g we report the distributions of three observable characteristics of contraction events extracted from SIM images. The duration of a contraction event (Fig.5g left) is computed as the difference between the last and first frames for which the event is detected; The time interval between successive contractions (Fig.5g middle) is computed for each individual tubule and pair of successive contraction events as the difference between the first frame of the second contraction and the last frame of the first contraction; Finally the length of a contraction (Fig.5g right) is computed by counting the number of pixels along the tubule axis involved in the contraction.

2.2 Elementary model of tubule contraction

The relation between the ER constrictions inside the tubules and the flow is suggested rather than correlative. To increase the evidences of this correlation, we propose an elementary computation to link the contractions and flow. Considering an incompressible ER luminal fluid, the conservation of the mass is

$$\frac{\partial \rho}{\partial t} = div(\boldsymbol{v}\rho), \tag{48}$$

where ρ is the fluid density and $\boldsymbol{v}$ the velocity at position $\boldsymbol{x}$. When a constriction occurs, we suppose that it leads to a decreased volume V_c, that generates a local flow. This flow can be obtained by integrating eq. 48 inside the tubule and we get:

$$\frac{dV}{dt} = S_2 v_2 - S_1 v_1, \tag{49}$$

where S_2, S_1 are the cross-sections at the right and left of the constriction associated with a constant velocity v_1 (resp. v_2) on the surface (note that we assume that there is no flow through the lateral surface of the tubule). When $S_2 = S_1$, we obtain since $v_2 = -v_1$ (by symmetry), so that the initial flow is given during the constriction phase by

$$v = \frac{1}{2S_1}\frac{dV}{dt}. \tag{50}$$

Thus a constriction occurring in a cylinder of constant section πr^2 along a segment of size $L = 300$ nm during $15ms$, leads to a velocity $V = 0.3/0.03 = 10\mu m/s$. To recover the velocity at the junction, we need now to model how this change in the tubule shape contributes to the velocity of ejection v_{expelled}.
A possible model is that the flow enters the node through a smaller section than the radius of the tubule. Suppose that the size is $r = 0.5 r_{tub}$, then using the mass conservation with this ratio of surfaces, we obtain a velocity at the entrance of a junction of

$$v_{\text{entrance}} = \frac{S_1}{S_{\text{entrance}}} v, \tag{51}$$

leading to a factor 4. Thus we obtain a velocity that could reach $v_{\text{entrance}} = 40\mu m/s$, compatible with the maximum velocity we find for the ejection of trajectories (Fig.2).
If the cross-section in the node at the two opposite tubules from the one where the flow is generated are identical to the one receiving the flow, we finally get the relation

$$v_{\text{expelled}} = \frac{v_{\text{entrance}}}{2} \approx 20\mu m/s. \tag{52}$$

Fig.5h illustrates the consequence of a constriction: a local constriction generates a flow v in both directions. The flow leads to an acceleration at the entrance of a tubule, if the entrance has a small surface. Due to the flow conservation, the velocity of the expelled trajectory is of the order $20\mu m/s$. Considering a pinch length of $L = 100$nm we obtain $v \approx 3\mu m/s$, $v_{\text{entrance}} = 13\mu m/s$, $v_{\text{expelled}} = 7\mu m/s$ while for $L = 400nm$ we obtain $v \approx 13\mu m/s$, $v_{\text{entrance}} \approx 53\mu m/s$ and $v_{\text{expelled}} = 27\mu m/s$.

2.3 Simultaneous contractions statistics

Under the assumption that contractions are spatio-temporally independent events, the probability of a contraction follows a Poissonian distribution of rate λ such that

$$P\{\text{one contraction in } [t, t+\Delta t]\} = \lambda \Delta t \tag{53}$$

Thus the probability of n contractions during that time interval is a rare event which probability is given by

$$\begin{aligned} P\{n \text{ contractions in } [t, t+\Delta t]\} =& (P\{\text{one contraction in } [t, t+\Delta t]\})^n \\ =& (\lambda \Delta t)^n. \end{aligned}$$

Considering $\lambda = 1/f_{contraction}$ where $f_{contraction} = 1.5$Hz as given in Fig.5f during a time step of $\Delta t = 100$ms (the acquisition time of SIM images), then $P\{\text{one contraction in } [t, t+\Delta t]\} = 0.15$ and $P\{\text{two contractions in } [t, t+\Delta t]\} = 0.0225$ and thus can be neglected compared to one contraction.

3 Tables

	Pure Diffusion Model	Flow-Diffusion Model
Dataset 1 (Main text)	$D_{app} = 1.13\ \mu\text{m}^2/\text{s}$	$D_{node} = 0.19\ \mu\text{m}^2/\text{s}$ $\mu_j = 22.9\ \mu\text{m/s}$ $\sigma_j = 6.92\ \mu\text{m/s}$ $\kappa = 0.33$
Dataset 2 (Fig.S3a)	$D_{app} = 1.02\ \mu\text{m}^2/\text{s}$	$D_{node} = 0.35\ \mu\text{m}^2/\text{s}$ $\mu_j = 23.19\ \mu\text{m/s}$ $\sigma_j = 3.85\ \mu\text{m/s}$ $\kappa = 0.31$
Dataset 3 (Fig.S3b)	$D_{app} = 4.10\ \mu\text{m}^2/\text{s}$	$D_{node} = 0.57\ \mu\text{m}^2/\text{s}$ $\mu_j = 45.01\ \mu\text{m/s}$ $\sigma_j = 12.75\ \mu\text{m/s}$ $\kappa = 0.41$

Table 1: Estimated motion parameters of the pure diffusion and flow-diffusion models for the datasets presented in the main text and supplementary figures.

		Datasets		
Symbol	Description	Main text	Fig.S3a	Fig.S3b
V_L	Max. inst. vel. (μm/s)	9.5	10	19
R	Max. neighbor distance (μm)	0.1	0.12	0.18
N	Min. num. points in cluster	25	35	30

Table 2: Parameters used for fitting the network on the presented datasets.

References

[1] Metzler R. and Klafter J. (2000). ”The random walk's guide to anomalous diffusion: a fractional dynamics approach”, Physics reports, 339(1), 1-77.

[2] Michalet X. (2010). ”Mean Square Displacement Analysis of Single-Particle Trajectories with Localization Error: Brownian Motion in Isotropic Medium”, Phys. Rev. E, 82.4: 041914.

[3] Z. Schuss (2010). ”Diffusion and Stochastic Processes: an Analytical Approach”, Springer, New York,.

[4] Kendall M. and Stuart A. (1961). ”The advanced theory of statistics: Inference and relationship”, Hafner Publishing, New York

[5] Ester M., Kriegel H. P., Sander J. and Xu X. (1996). ”A density-based algorithm for discovering clusters in large spatial databases with noise”, Kdd Vol. 96, No. 34, pp. 226-231.

[6] Pedregosa F., Varoquaux G., Gramfort A., Michel V., Thirion B., Grisel O., Blondel M., Prettenhofer P., Weiss R., Dubourg V., Vanderplas J., Passos A., Cournapeau D., Brucher M., Perrot M. and Duchesnay E. (2011). ”Scikit-learn: Machine Learning in Python”, Journal of Machine Learning Research.

[7] Langevin P. (1908). ”Sur la théorie du mouvement Brownien”, C.R. Paris, 146, 530–533.

[8] Chenouard N., Bloch I. and Olivo-Marin, J.C. (2013). ”Multiple hypothesis tracking for cluttered biological image sequences”, IEEE transactions on pattern analysis and machine intelligence, 35(11), 2736-3750.

[9] Hoze N. and Holcman D. (2017). ”Statistical Methods for Large Ensembles of Super-Resolution Stochastic Single Particle Trajectories in Cell Biology”, Annual Review of Statistics and Its Application.

[10] Holcman D., Hoze N. and Schuss Z. (2015). ”Analysis and interpretation of superresolution single-particle trajectories”, Biophysical journal, 109 (9), 1761-1771.

[11] Hoze N., Nair D., Hosy E., Sieben C., Manley S., Herrmann A., Sibarita J.B., Choquet D. and Holcman D. (2012). "Heterogeneity of receptor trafficking and molecular interactions revealed by superresolution analysis of live cell imaging", Proc. Natl. Acad. Sci. USA, 109: 17052–17057.

[12] Hoze N. and Holcman D. (2014). "Residence times of receptors in dendritic spines analyzed by stochastic simulations in empirical domains", Biophysical journal 107 (12), 3008-3017

[13] Hoze N. and Holcman D. (2015). "Recovering a stochastic process from super-resolution noisy ensembles of single-particle trajectories", Physical Review E 92 (5), 052109 2015.

[14] Hoze N. and Holcman D. (2017)."Statistical Methods for Large Ensembles of Super-Resolution Stochastic Single Particle Trajectories in Cell Biology", Annual Review of Statistics and Its Application, (4):189-223.

[15] Tarjan R. (1972). "Depth-first search and linear graph algorithms", SIAM vol. 1, no 2, p. 146-160.

[16] Nixon-Abell J., Obara J. C., Weigel V. A., Li D., Legant R. W., Shan Xu C., Pasolli A., Harvey K, Hess F.H., Betzig E., Blackstone C. and Lippincott-Schwartz J. (2016). "Increased spatiotemporal resolution reveals highly dynamic dense tubular matrices in the peripheral ER", Science, 354.

[17] Arganda-Carreras I., Fernandez-Gonzalez R., Munoz-Barrutia A. and Ortiz-De-Solorzano C. (2010), "3D reconstruction of histological sections: Application to mammary gland tissue", Microscopy Research and Technique, Volume 73, Issue 11, pages 10191029.

[18] Tang C.Y. and Chen S.X. (2009). "Parameter estimation and bias correction for diffusion processes", Journal of Econometrics, vol. 149, p. 65-81.

[19] Amitai A., Seeber A., Gasser S.M. and Holcman D. (2017). "Visualization of chromatin decompaction and break site extrusion as predicted by statistical polymer modeling of single-locus trajectories.", Cell Reports, 18(5), 1200-1214.

[20] Bishop C.M. (2006), "Pattern recognition and machine learning", Springer.

[21] McKinney S.A., Murphy C.S., Hazelwood K.L., Davidson M.W., Looger L.L. (2009), "A bright and photostable photoconvertible fluorescent protein", Nature Methods, 6(2), 131.