Endocytosis by Random Initiation and Stabilization of Clathrin-Coated Pits

Ehrlich, Marcelo; Boll, Werner; van Oijen, Antonius; Hariharan, Ramesh; Chandran, Kartik; Nibert, Max L.; Kirchhausen, Tomas

Published in:
Cell

DOI:
10.1016/j.cell.2004.08.017

IMPORTANT NOTE: You are advised to consult the publisher’s version (publisher’s PDF) if you wish to cite from it. Please check the document version below.

Document Version
Publisher’s PDF, also known as Version of record

Publication date:
2004

Link to publication in University of Groningen/UMCG research database

Citation for published version (APA):

Copyright
Other than for strictly personal use, it is not permitted to download or to forward/distribute the text or part of it without the consent of the author(s) and/or copyright holder(s), unless the work is under an open content license (like Creative Commons).

Take-down policy
If you believe that this document breaches copyright please contact us providing details, and we will remove access to the work immediately and investigate your claim.

Downloaded from the University of Groningen/UMCG research database (Pure): http://www.rug.nl/research/portal. For technical reasons the number of authors shown on this cover page is limited to 10 maximum.
Supplemental Data for:

Ehrlich et al., Cell 118, pp. 591–605

Sampling Statistics
As background fluctuations of fluorescence intensity have an inherent random distribution that could bias our results, it was critical to rule out the possibility of incorrect identification of weak fluorescent signals. This was done by examining the noise level within the tracked data following the analysis of sampling statistics in the images. We determined the standard deviation of the background fluorescence (13.5 ± 0.5 fluorescent units) and then looked for the weakest spots assigned as coated pits. Their peak intensity (16 fluorescence units after background subtraction) results in a signal-to-noise ratio of 16/13.5 = 1.185. Assuming that these fluorescent clusters are Gaussian in shape (size of 3 × 3 pixels) and assuming a full-width half-maximum of 3 pixels, the average fluorescent intensity in each pixel of a cluster is roughly five sixths of that of the center pixel, or 13.33 fluorescence units. Thus, the average signal-to-noise for all pixels within the cluster becomes 13.33/13.5 = 0.988, and, considering a standard normal distribution, the signal-to-noise ratio for a single pixel gives a significance level of 0.322, equivalent to a 67.8% chance that the signal is real. Because the fluorescent cluster contains nine such pixels, the signal-to-noise ratio improves to 2.994 with a significance level of 0.0028 (or 99.72% chance that this signal is real), since it is proportional to the square root of the sampling number. Finally, by imposing the criterion that the fluorescent spot for a coated pit has to remain in the same location in three consecutive time frames, one obtains an increase (proportional to the square root of the number of consecutive images) to 5.2 in the signal-to-noise ratio. This ratio corresponds to a significance level well below 0.0005, indicating that the weak events scored by the program (e.g., those with fluorescent intensities corresponding to 15% of the signal elicited by a complete coated vesicle) have a 99.95% likelihood of being real. Similarly, we estimate that events corresponding to 50% of the fluorescent signal elicited by a coated vesicle have a 99.9999% chance of being real.

Spatial Statistics
In spatial statistics, Ripley’s K function (Ripley, 1976) is a classical tool to analyze spatial point patterns. The definition of the K function is as follows:

\[
K(t) = \frac{E(d < t)}{\lambda} \tag{1}
\]

where \(E(d < t)\) denotes the number of particles within a distance \(t\) of an arbitrary particle, and \(\lambda\) the density of particles (mean number of particles per unit area). The density can be estimated by \(N/A\), where \(N\) is the observed number of points and \(A\) is the area of the field of view. The numerator in Equation 1 can be estimated by

\[
N^{-1} \sum_{i} \sum_{j \neq i} I(d_{ij} < t) \tag{2}
\]

where \(d_{ij}\) is the distance between the \(i\)th and \(j\)th points, and \(I(x)\) is the indicator function, with the value 1 if \(x\) is true and 0 if otherwise.

To prevent a bias in \(K(t)\) values at larger values of \(t\) due to the finite size of the field of view, an edge correction is introduced,

\[
K(t) = \frac{\lambda^{-1} N^{-1} \sum_{i} \sum_{j \neq i} \omega(l_i, l_j) I(d_{ij} < t)}{1 - \frac{A}{\pi t^2}} \tag{3}
\]

with the weight function \(\omega(l_i, l_j)\) providing the edge correction (Ripley, 1976). It has the value 1 when the circle centered at \(l_i\) and passing through the point \(l_j\) (i.e., with radius of \(d_{ij}\)) is completely inside the field of view. If part of the circle falls outside the field of view (i.e., \(d_{ij}\) is larger than the distance from \(l_i\) to at least one boundary), \(\omega(l_i, l_j)\) is the proportion of the circumference of that circle that falls in the field of view.

Ripley’s K function is particularly powerful to test spatial randomness, i.e., whether the spatial distribution of the events is consistent with a homogeneous Poisson process. In that case, \(K(t) = \pi t^2\) for all \(t\). To facilitate
direct comparisons to Poissonian processes, where \( L(t) \) is equal to \( t \), we use the scaled function \( L(t) = (K(t)/\pi)^{1/2} \). The function \( L(t) \) was calculated for every data set, consisting of the \( x, y \) positions of all observed clathrin or \( \sigma_2 \)-adaptin clusters in one particular experiment. To estimate the error for \( L(t) \) of the experimental datasets, we simulated Poissonian distributions of events with identical field and sample size as the experimental data. \( L(t) \) was calculated for a large number of simulations (\( n = 40 \) per dataset), and the standard deviation obtained from the distributions in \( L(t) \) was adopted as an error estimate for the data presented in Figure 2. All analyses and simulations were performed in MATLAB (Mathworks, Natick, MA). The code written in MATLAB to simulate the Ripley’s K function, and the diffusion of LDL is available on request.

**References**

