Supplementary information:

Solving single molecules: filtering noisy discrete data made of photons and other type of observables

Ophir Flomenbom, Flomenbom-BPS Ltd, 19 Louis Marshal St., Tel Aviv, Israel 66268

In the supplementary file, I show: the filter presented in the main text solves cases where binning the data doesn't work: the obtained binned trajectory is "too" noisy and all the methods that are used on the binned data will not help. (The methods in [50] do not filter the noise: these calculate correlation functions from the raw data (and from slightly smoothed data) but this does not clean the data and the extracted information is rather poor in content).

In the figures we simply show that when the noise increases there are cases where binning results in trajectories where an *off* event might contain a peak that is identical with *on* events and that there are *on* events that contain "too" few photons, identical with an *off* event. Thus any method that is applied on the binned data will not filter such cases.

We highlight the point that here the noise is since also in the *off* state photons are recorded with an exponential rate. Fourier transformed the binned data will not help much here, in particular in the cases specified in the next figures.

Just an algorithm that study every photon and using the local information of consecutive photons and global information (thresholds extracted from the entire data) can help. The algorithm of the main text constituting the best way that enabling solving the data photon after photon.

A simple case.

In figure 1 in the supplementary material, we present data that is generated from the mechanism in figure 2A in the main text. The rates are: λ_{on}=1/10,γ_{on}=1, λ_{off}=1/99, γ_{off}=1/10 (all units scaled). This figure is with various panels: the upper panel is the clean data and the binned data: number of photons in a bin of size 1.99. The other (lower) panel is the photon durations in the order recorded and we also plot the real identification: the larger value is the *about* $1/\gamma_{off}$ and the smaller value is the about $1/\gamma_{on}$

This set up generating very clear two state data: we see the two states also in the binned data (middle of the figure, blue) and the binned data & the clean data (red curve) coincide. Thus cleaning the noise from the binned data is simple. We show with arrows events that are missed (green: missed *on* event, black: missed *off* event) when thresholding the binned data with threshold = 3.

The lower panel showing the raw data, photon durations in the order recorded (blue) with the real identification (green). Clearly this trajectory is smoother than the binned data. Although in this case also the binned data is not “too” noisy we show here cases where filtering the raw data is simple but the binned data is too noisy: these are labeled with # and @ in both panels.

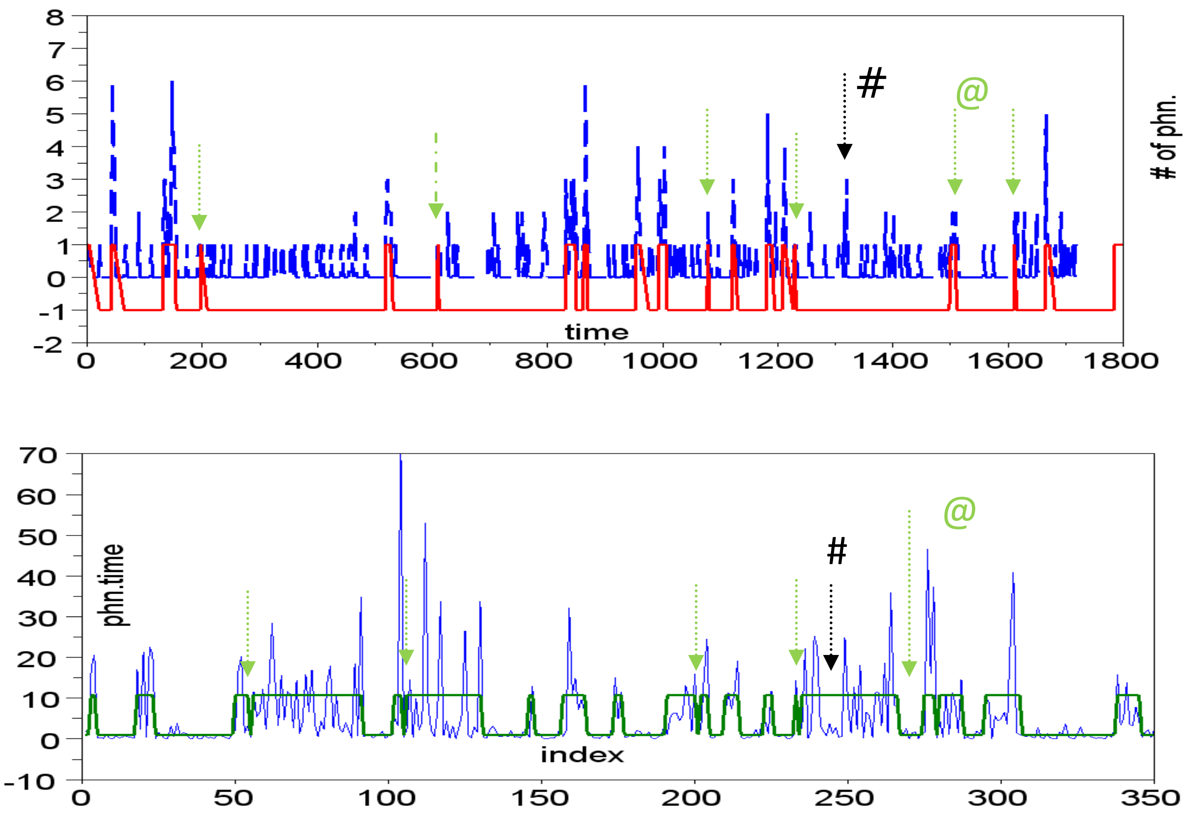

Figure 1 supplementary information.- The binned data (upper panel), and the raw data (lower panel), the photon durations. In both panel, we show also the clean data. Description is presented in the text

A more complicated case.

The data was generated from KS 2A, yet the rates are: λ_{on}=1/10,γ_{on}=1, λ_{off}=1/49, γ_{off}=1/10.
I have changed the rate controlling the *off* durations: here this rate is just 1/49 and in the previous example λ_{off} is 1/99. This resulting in many *off* events with few photons. This increases the noise: there are not so few *off* events that do not have the required amount of photons needed in order seeing a very slow one. In general, we see here that the binned data is rather noisy, clearly relative with the raw data representation of photon durations in the order recorded. I solved such data with my methods (see table 1). In the figure, we show with arrows all the cases where the binned data is *on* but smaller than the threshold: threshold =3. We show these cases also in the raw data representation. At least case 2 &3 (chronological order) are solved with the filter presented in the main text yet are missed in the binned data with any filtering technique. This is at least 25% better.

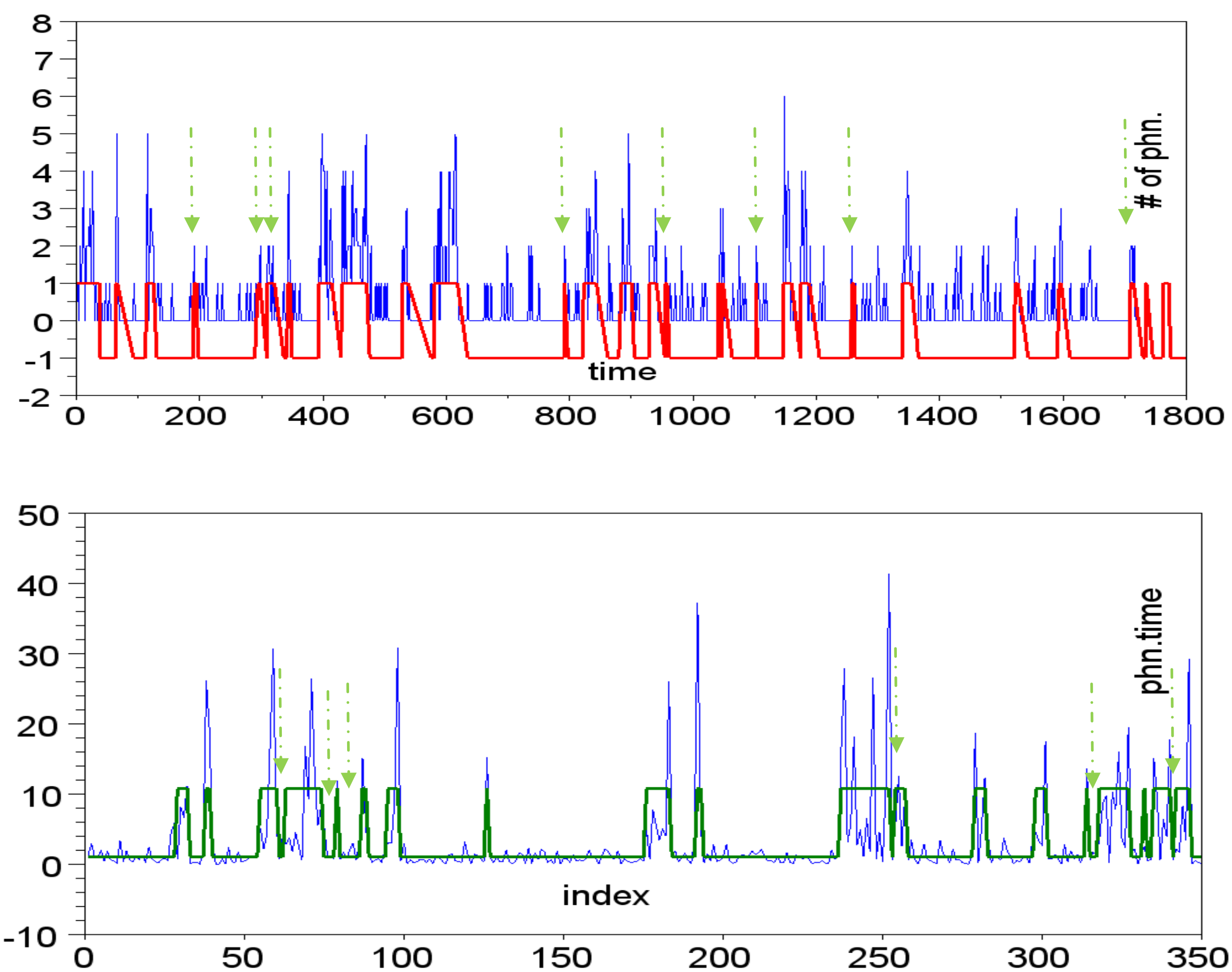

Figure 2 supplementary information.- The description is presented in the text

The complicated case.

The data was generated from KS 2A, yet the rates are: λ_{on}=1/10,γ_{on}=1, λ_{off}=1/33, γ_{off}=1/10. I have changed the rate controlling the *off* durations: here this rate is just 1/33 and in the previous example λ_{off} is 1/49.

This set up resulting in many *off* events with few photons: shorter events than in the previous example. This increases the noise: there are not so few *off* events that do not have the required amount of photons needed in order seeing a very slow one. Again: in general, we see here that the binned is rather noisy, clearly relative with the raw data representation of photon durations in the order recorded. I solved also such data with my methods (see table 1). In the figure, we show with arrows all the cases where the binned data is *on* but smaller than the threshold: threshold =3. These are shown in green. We show with a black arrow when having an *off* event with many photons in a bin: larger than 3. We show all these cases also in the raw data representation. At least case 1 & 2 & 9 (chronological) are solved with the filter presented in the main text yet are missed in the binned data with any filtering technique. This is at least 33% better. We show in panel 3 that also increasing the bin size does not improve the situation: one additional *on* event is identified but one *off* event has many photons in a bin.

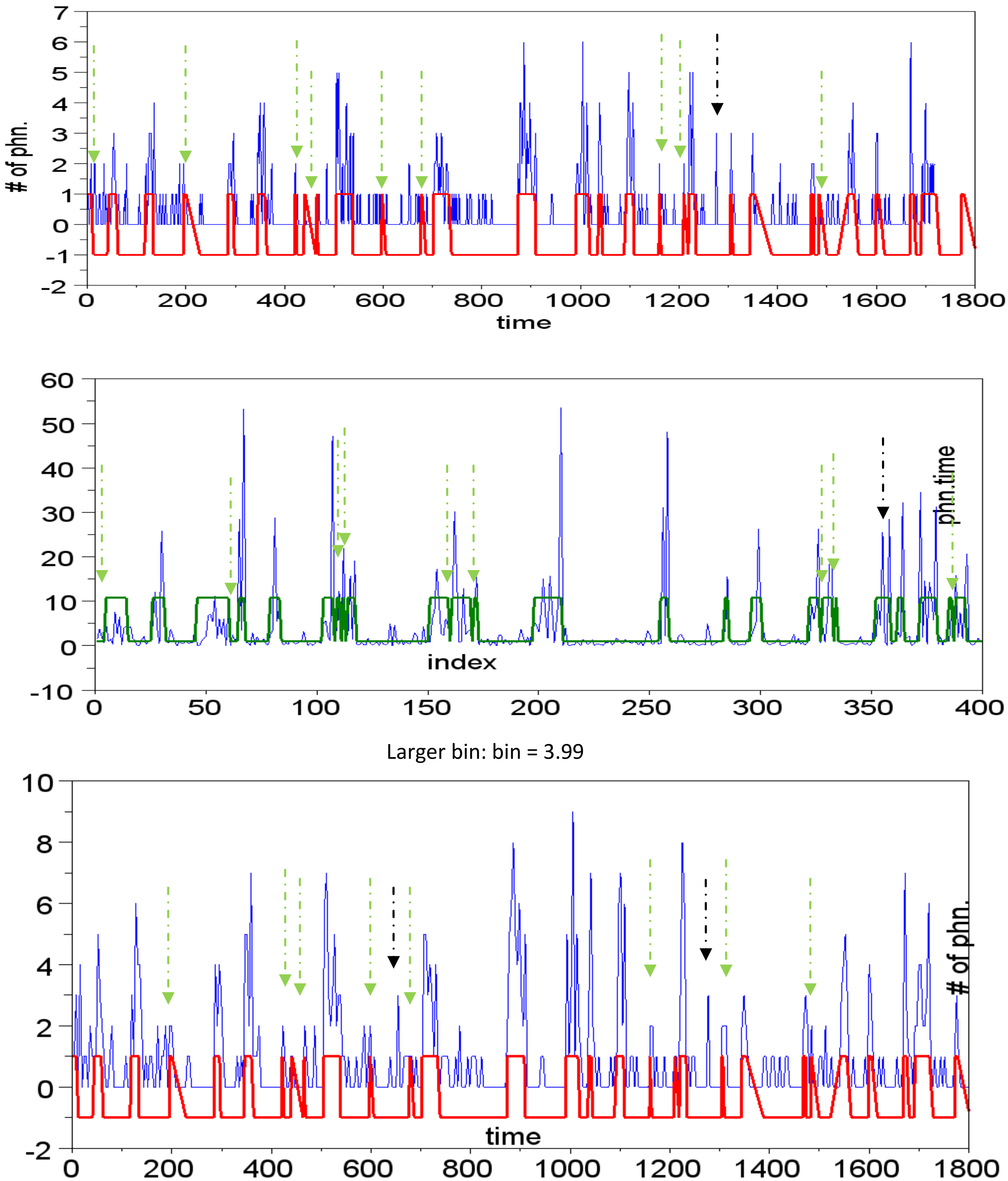

Figure 3 supplementary information.- The description is presented in the text