Yersinia enterocolitica Type III secretion injectisomes form regularly spaced clusters which incorporate new machines upon activation

Mikhail Kudryashev, Andreas Diepold, Marlise Amstutz, Judith P. Armitage, Henning Stahlberg and Guy R. Cornelis

Supporting Information

Table S1

List of strains and plasmids used in this study

Y. enterocolitica strains used in this study. Besides the wild type strain E40, all strains are based on the multi-effector knock-out auxotrophic strain IML421*asd*, which is based on E40 itself.

Strain	Relevant characteristics	References
WT	E40 (pYVe40)	(Sory *et al*, 1995)
IML421*asd*	E40 (pYVe40 $yopO_{\Delta 2\text{-}427}$ $yopE_{21}$ $yopH_{\Delta 1\text{-}352}$ $yopM_{23}$ $yopP_{23}$ $yopT_{135}$) $\Delta asd_{\Delta 292\text{-}610}$	(Kudryashev *et al*, 2013)
AD4085	IML421*asd* *(egfp-yscQ)*	(Kudryashev *et al*, 2013)
AD4306	IML421*asd* *(egfp-yscD) (mutated with pAD306)*	(Diepold *et al*., submitted)
AD4334	IML421*asd* (*yscV-mCherry) (mutated with pAD334)*	(Diepold *et al*., submitted)
MA4039	IML421*asd* *(egfp-yscQ)* $\Delta minD$	(Kudryashev *et al*, 2013)

Expression and suicide vectors

Plasmid	Relevant characteristics	References
pAD306	pKNG101 *egfp-yscD*$^{+}$ (*egfp* and flexible linker cloned in-frame at the N-terminus of *yscD*)	(Diepold *et al*., submitted)
pAD334	pKNG101 *yscV-mCherry*$^{+}$ (*mcherry* and flexible linker cloned in-frame at the C-terminus of *yscV*)	(Diepold *et al*., submitted)

Text S1. Matlab code for automated spot counting

```
% Dynamo is available at www.dynamo-em.org
% read more in  Castaño-Díez, D, et al, JSB, 2012
% save the original TIFF stacks and AVI, then

for moviename = 1:14;
   clear vol;
   for slice = 1:24 %24 is the size of the stack

      sl = aviread(['stacks/MAAD4006_Ox_' num2strtotal(moviename,2) '_R3D_D3D.avi'],slice);
      % adding normalized slices to the volume
      vol(:,:,slice) = (double(sl.cdata))-mean(mean(double(sl.cdata)));

   end
   % writing out the volume for further procesing
   vol = dynamo_normalize(double(vol));
   dwrite(vol, ['movie_' num2str(moviename) '.em']);

end

%% Pick two tips of each cell, I used tom_particles
cnt = 0; % total cell count

for movienum = 1:14;
   % reading in volumes and picked cell's coordinates
vol = dread(['movie_' num2str(movienum) '.em']);
motl = dread(['motl_movie' num2str(movienum) '.em']);

% normalization and addition of small random values
vol = dynamo_normalize(vol(:,:,4:22));vol = vol + rand(size(vol))* 0.0001;

% loop over bacteria in each stack
for i = 1:size(motl,2)/2
   %calculating centers and lengths of every bacteria in pixels
   centers(i,1:3) = (motl(6:8,i*2) + motl(6:8,i*2-1))/2;% + [-7 -7 0]';
   len(i) = sqrt(sum(abs(motl(6:8,i*2) - motl(6:8,i*2-1))).^2);

   cntpt(i) = 0;
   % edges of the boxes for sub-cropping
   c1 = floor([centers(i,1:2)-(len(i)/2 +10) 4]);
   c2 = floor([centers(i,1:2)+(len(i)/2 +10) 22]);
   try
      svol = vol(c1(1):c2(1), c1(2):c2(2), :);
   catch
      break;
   end
   %generation of a mask for the cell by just low pass filter, optional
   svol1 = svol;
      maxcc = 100;
   % loop: finding the highest value and erasing it.
   while maxcc>5
      [a maxcc] = peak_det_2(svol.*mask);
      svol = svol.*(1-dynamo_sphere(3,size(svol), a, 3));
      svol1 = svol1 + dynamo_sphere(1,size(svol), a, 0)*50;
      cntpt(i) = cntpt(i) + 1;
   end
   len_all(cnt+1) = len(i);
   dots_all(cnt+1) = cntpt(i);
   cnt = cnt+1;
end %end of the bacteria loop
end % end of the movie loop
```