Supporting Information

A simple and quantitative method to evaluate ice recrystallization kinetics using the circle Hough transform algorithm

Luuk L.C. Olijve,[1,2] Anneloes S. Oude Vrielink,[1,2] Ilja K. Voets[1,2,3*]

[1]Institute for Complex Molecular Systems, Eindhoven University of Technology, Post Office Box 513, 5600 MD Eindhoven, Netherlands. [2]Laboratory of Macromolecular and Organic Chemistry, Department of Chemical Engineering and Chemistry, Eindhoven University of Technology, Post Office Box 513, 5600 MD Eindhoven, Netherlands. [3]Laboratory of Physical Chemistry, Department of Chemical Engineering and Chemistry, Eindhoven University of Technology, Post Office Box 513, 5600 MD Eindhoven, Netherlands.

Contents

Time evolution of mean ice crystal size in the presence of $AFGP_{1-5}$, QAE WT and QAE T18N

Polarized optical microscopy experiments were performed to monitor the time evolution of the mean ice crystal volume ($\langle R_n \rangle^3$) in the presence of the ocean pout type III AFP (Figure 1, 0.02 μM < c < 25 μM), the T18N mutant (Figure 2, 0.02 μM < c < 25 μM), and AFGP1-5 (Figure 3, $6.7 \cdot 10^{-5}$ < c < 6.7 μM). Images were collected for 90 min starting from the moment when the sample had reached a temperature of -7 °C, which is taken as t = 0 minutes. As expected, the rate at which $\langle R_n \rangle^3$ increases (i.e., k_d, which corresponds to the slope of the curves in Figure 1-3), increases as the AFP concentration decreases. Above a threshold concentration, the mean ice crystal size remains virtually constant during the experiment, which means that ice recrystallization is effectively inhibited at and above that high AFP concentration. For $AFGP_{1-5}$, k_d reaches $k_d \sim 0$ μm^3/min at a much lower AFP concentration than for wild-type QAE and the T18N mutant, which indicates that $AFGP_{1-5}$ has a higher IRI activity.

We note that k_d is not entirely time-independent during all measurements. In some experiments, k_d decreases somewhat during the assay, which may be due to a transition from a 'liquid phase diffusion' rate limiting process to a 'liquid-to-ice phase transfer' rate limiting process; the latter caused by the adsorption of AFPs on the ice surface.[1] This underlines the importance of prolonged, quantitative image analysis during a substantial time window, rather than inspection of a limited number of frames at a particular point in time (such as e.g. at t = 90 min only). Herein, the concentration-dependent k_d of QAE WT, QAE T18N, and $AFGP_{1-5}$ were determined as $k_d = \Delta\langle R_n \rangle^3 / \Delta t$ with $\Delta t = t_{90} - t_{30}$ = 60 minutes for QAE WT and QAE T18N, and $\Delta t = t_{90} - t_0$ = 90 minutes for $AFGP_{1-5}$.

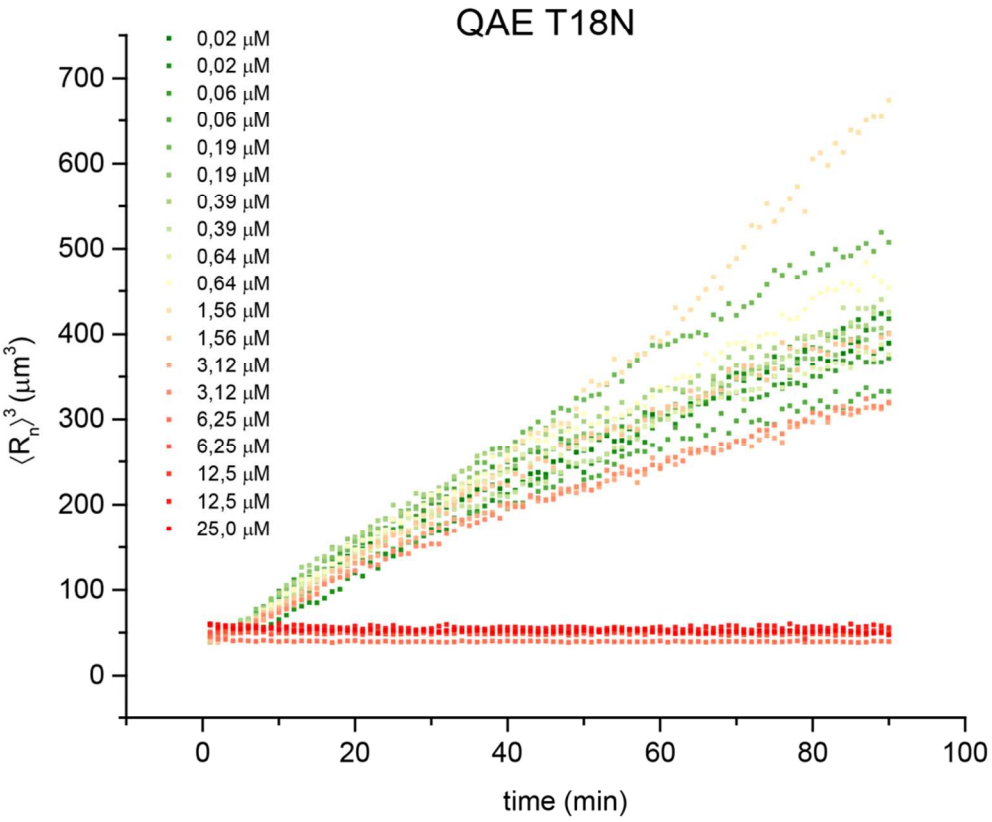

Figure 1: *$\langle R_n \rangle^3$ versus time for different concentrations (0.02 µM < c < 25 µM) of QAE T18N. IRI measurements at concentrations up to c=12,5 µM were performed in duplicate. Images were recorded for 90 minutes with a 1 minute time interval.*

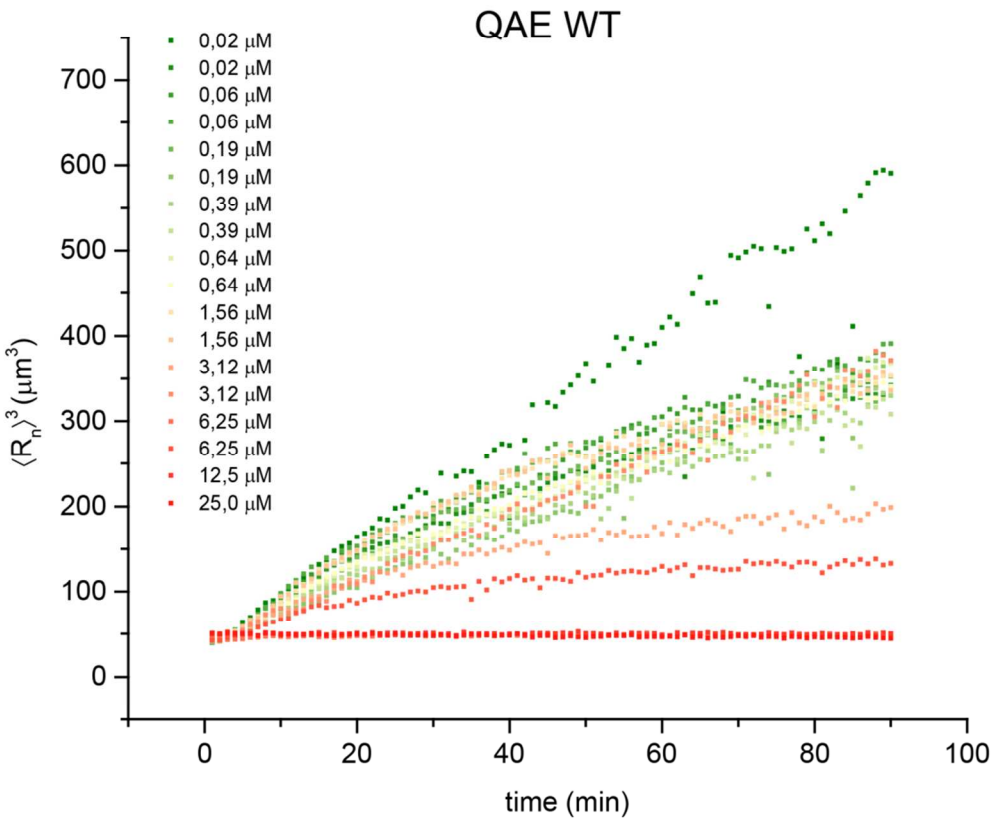

Figure 2: $\langle R_n \rangle^3$ *versus time for different concentrations (0.02 µM < c < 25 µM) of QAE WT. IRI measurements at concentrations up to c=6,25 µM were performed in duplicate. Images were recorded for 90 minutes with a 1 minute time interval.*

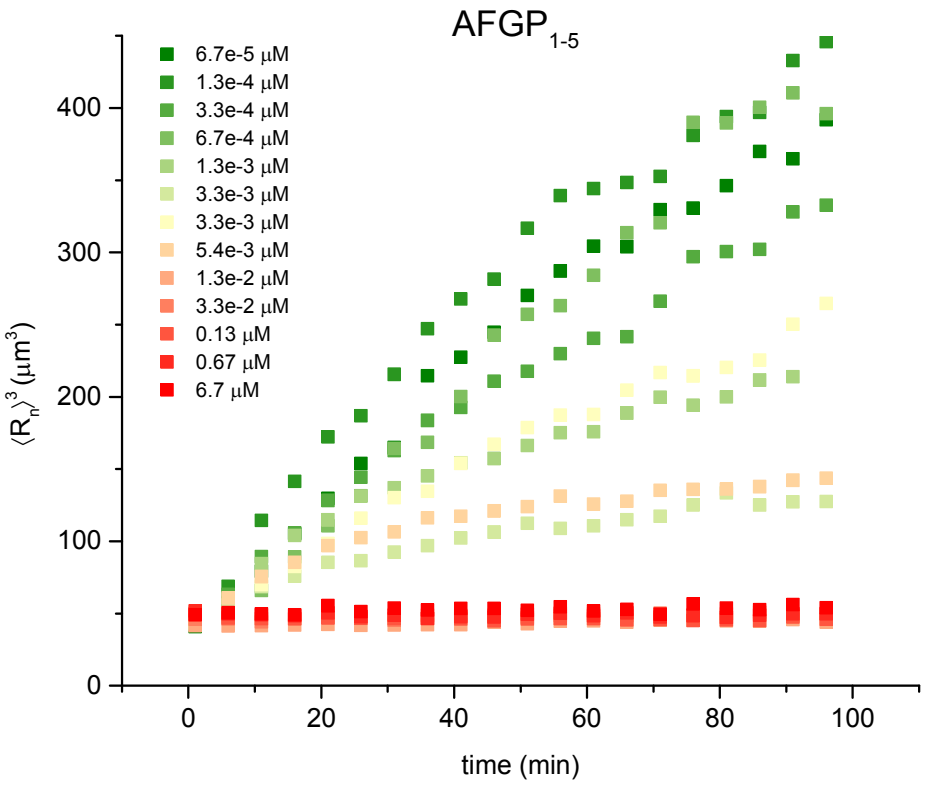

Figure 3: *$\langle R_n \rangle^3$ versus time for different concentrations ($6.7 \cdot 10^{-5} < c < 6.7$ μM) of $AFGP_{1\text{-}5}$. Images were recorded for 90 minutes with a 5 minute time interval.*

Sonocrystallization measurements on QAE WT and QAE T18N

In contrast to the IRI measurements, a large difference in activity between QAE WT and QAE T18N is observed when analyzed by sonocrystallization.[2, 3] To corroborate these results and circumvent batch-to-batch differences, we performed sonocrystallization measurements on the same batch of QAE T18N as the IRI experiments described in the main text. During the sonocrystallization assay of thermal hysteresis activity, samples are cooled to -6 °C before application of a sonopulse at t ~1390 seconds to induce freezing (Figure 4). Three measurements were performed on a 4.6 mg/ml solution of QAE T18N (0.57 mM), which exhibits a freezing plateau at T_f = -0.14 °C. We determine thermal hysteresis (TH) as the difference between the freezing point of the sample and the melting point of the buffer, hence TH = T_f - T_m. Using the freezing plateau of the buffer (T_m = -0.074 °C), we obtain for the thermal hysteresis of the 4.6 mg/ml solution of QAE T18N a value of TH = 0.07 °C. Similarly, for a 4.2 mg/ml solution of QAE WT (0.52 mM), we have T_f = -0,71 °C, which gives TH = 0.64 °C. These measurements indicate that the T18N mutation has a large effect on the thermal hysteresis activity of QAE measured by sonocrystallization. Moreover, we find a longer freezing plateau for QAE WT than for QAE T18N.

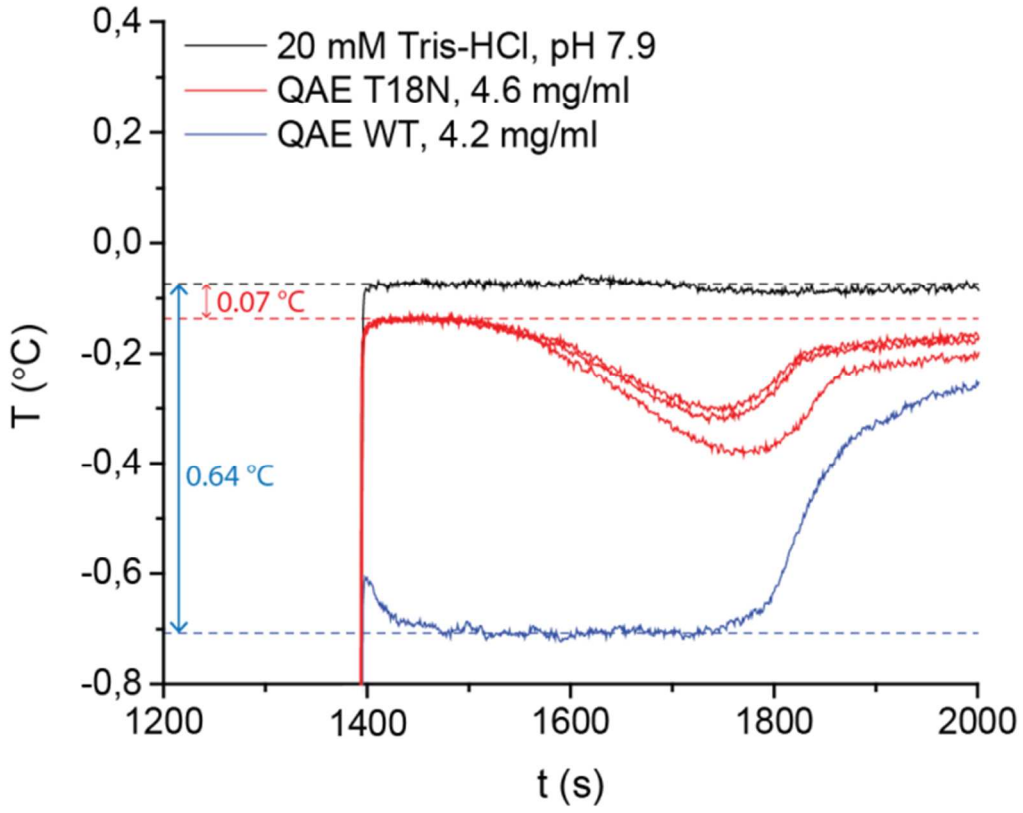

Figure 4: *Sonocrystallization measurements on QAE T18N and QAE WT. A thermal hysteresis TH = 0.07 °C was obtained for QAE T18N at 4.6 mg/ml (0.57 mM), which is significantly smaller than TH = 0.64 °C for QAE WT at 4.2 mg/ml (0.52 mM).*

Matlab scripts

Script 1 mainCHT

```
%% Ice recrystallization inhibition analysis using circe Hough transform
% Last update: May 5, 2016 (Anneloes Oude Vrielink)
% Eindhoven University of Technology
% Minimal requirements: MATLAB R2103a and Image Processing Toolbox
% reference: http://nl.mathworks.com/help/images/ref/imfindcircles.html

% This script runs the function_CHT for all (bmp) images in the current
% experiment / folder.

% Fill-in the number of images, time between subsequent images (min) and
% timepoint of the first image (min) before running the script.

% For each set of images an excel file is created that contains the
% increase in Rn^3 over time. All files that were generated by running this
% script and function_CHT are placed in a new folder.

clear,close all
clc

fname={}; d=dir('*.bmp'); fnames={d.name};

radius=[];data=[];

No_images = 90; % number of images
t_interval = 1; % time between subsequent images (minutes)
t_start = 1; % timepoint of first image (minutes)
time_array=t_start:t_interval:(t_start+(No_images-1)*t_interval);

% Read and analyze every images in folder.
for k=1:length(d)
    fname=d(k).name;
    img = imread(fname,'bmp');
    time = time_array(k);
    [radii, centers, radius] = functionCHT(time,img);
    results = [radii, centers];
    data = [data;radius];
    t = strtok(fname, '.');
    col_header = {'R (pixels)','center_x (pixels)','center_y (pixels)'};
    xlswrite(t, col_header, 'Sheet1', 'A1');
    xlswrite(t, results,'Sheet1','A2'); % For each image an excel file is
% generated that contains the radii (pixels) and center coordinates (pixels)
% of detected circles.
end

data=sortrows(data,1);
B = figure; t = data(:,1); y = data(:,3);
scatter(t,y); xlabel('Time (min)'); ylabel('Rn^3 (um^3)')
```

```
% Linear fit of time vs. radius^3
[p,bint] = polyfit(t,y,1);
p_err = sqrt(diag((bint.R)\inv(bint.R'))./bint.normr.^2./bint.df);
yfit = p(1).*t + p(2); hold on;
plot(t,yfit,'r-')
str = sprintf('r^3(t) = r0^3 + Kd*t, r0 = %5.3f +/- %5.3f, Kd = %5.3f +/-
%5.3f',p(2),p_err(2),p(1),p_err(1));
title(str); filename = sprintf('Kd.jpg', time); saveas(B,filename,'jpg')

% Write results to excel file.
results = [t, y, yfit];
currentDirectory = pwd;
[upperPath, deepestFolder, ~] = fileparts(currentDirectory);
col_header = {'t (min)','Rn^3 (um^3)','fit Rn^3 (um^3)'};
xlswrite(deepestFolder,col_header,'Sheet1','A1');
xlswrite(deepestFolder,results,'Sheet1','A2');

% Move files to new folder.
d = ['circle Hough analysis_',date];
mkdir(d);
movefile('*.xls',d);
movefile('*.jpg',d);
```

Script 2 function_CHT

```
% This script analyzes the ice grains in an image by performing the
% circular Hough transform.

% Fill-in the correction factor (um/pixel) to convert radii in pixels to
% radii in micrometers. Use 'imshow' and 'imdistline' to estimate the
% correction factor from an image of an calibration grid.

function [radii, centers, radius] = functionCHT(time,img)

cf = 0.2500; % correction factor (um/pixel)

% Find threshold and convert to binary image
level = graythresh(img); BW = im2bw(img,level);

BW2 = edge(BW,'canny',0.1);
figure,
imshow(imfuse(BW2,img,'falsecolor','Scaling','joint','ColorChannels','red-
cyan'))
title(time)

% It is possible that at the default sensitivity level all the circles are
% lower than the internal threshold, which is why no circles were detected.
% By default, 'Sensitivity', which is a number between 0 and 1, is set to
% 0.85. Increase 'Sensitivity' to 0.9.
% Default method is phase coding, alternative is twostage. In general, these
% two method are complementary in that have they have different strengths.
% Phase coding method is typically faster and slightly more robust to noise %
% than the two-stage method. But it may also need higher 'Sensitivity' levels
% to get the same number of detections as the two-stage method.
[centers1, radii1] = imfindcircles(BW2,[10
30],'Sensitivity',0.82,'Method','twostage');
[centers2, radii2] = imfindcircles(BW2,[31
50],'Sensitivity',0.84,'Method','twostage');
[centers3, radii3] = imfindcircles(BW2,[51
130],'Sensitivity',0.87,'Method','twostage');

centers = [centers1;centers2;centers3];
radii = [radii1;radii2;radii3];

figure
imshow(img)

% Save image with circles.
h = viscircles(centers,radii,'Linewidth',0.001);
filename = sprintf('Radii%d.jpg', time);
saveas(h,filename,'jpg')
close

% Plot histogram.
A = figure; xbins = 0:150; hist(radii,xbins);
filename = sprintf('Hist%d.jpg', time);
saveas(A,filename,'jpg')
```

```
close

% Determine radial number average, Rn.
c = hist(radii,xbins);
Rn = cf.*sum(xbins.*c) ./ sum(c); % Rn (um)
Rn3 = Rn^3; % Rn^3 (um^3)
radius = [time Rn Rn3];
```

Script 3 analyzeFolders

```
% This script can be used to perform the CHT analysis of the on all
% experiments in a specific folder. Each experiment has a separate folder
% containing the images of the experiment, and the scripts mainCHT and
% functionCHT.

% When, for some reason, the analysis of a certain experiment fails,
% the script continuous with the analysis of the next experiment.

% Fill-in the right working directory (pwd) before running the script.

folders = ls;
for a = 3:size(folders,1)
    folders = ls;
    cd([pwd,'\',folders(a,:)])
    try
        mainCHT_V4
        cd('D:\IRI measurements\experiment1’)
    catch
        warning('Problem in analysis of experiment.');
        cd('D:\IRI measurements\experiment1’)
    end
end
```

Script 4 Calculate Kd

```
% This function derives the slope (Kd) from the Rn^3 versus time plots,
% of a set of experiments.

% Input excel file contains the Rn^3 verus time data for a set of
experiments.
% The output excel file contains the slopes and intercepts of the fitted
lines.

% Fill-in the desired time frames of the analysis before running the script.

function Calculate_Kd(filename)

[data,txt,raw] = xlsread(filename);

t= data(30:end-1,1); % Data of 30 min to 89 min will be analyzed.
results = [];

for i = 2 : size(data,2);
    y = data(30:end-1,i); % Data of 30 min to 89 min will be analyzed.
    [p,S]=polyfit(t, y, 1);
    slope = p(1);
    intercept = p(2);
    results = [results; slope,intercept];
end

% Write results to excel file.
col_header = {'slope','intercept'};
row_header = transpose(txt(2:end));
[pathstr,name,ext] = fileparts(filename);
name = [name,'_slopes.xls'];
xlswrite(name,col_header,'Sheet1','B1');
xlswrite(name,row_header,'Sheet1','A2');
xlswrite(name,results, 'Sheet1','B2');

end
```

Script 5 FitData

```
% This script fits the function fit_fun to the data of Kd versus c(uM).
% Input is excel file contains the Kd versus c (uM) data, the output excel
% file contains the fitted parameters.

function FitData(filename);

close all

data = xlsread(filename);

x_data = data(:,1); % First column contains the c (uM)
y_data = data(:,2); % Second column contains the slope/Kd (um3/min)

figure
semilogx(x_data,y_data,'o')
xlabel('c(uM)'), ylabel('Kd (um3/min)');

fit_fun=@(x,par)par(1)-par(1)./(1+exp((par(2)-x)./par(3)));

obj_fun=@(par)sum((y_data-fit_fun(x_data,par)).^2);

par_start=[2,4,0.1]; % Indicate starting parameters for Kd0, ci and s
% respectively.

par_fit=fminsearch(obj_fun,par_start);

Kd0 = par_fit(1)
ci = par_fit(2)
s = par_fit(3)

hold on

powers2 = -6:0.01:3;
x_data2 = 10.^powers2;

plot(x_data2,fit_fun(x_data2,par_fit))

y_fit = fit_fun(x_data2,par_fit);

% Find the inflection point in the fitted function (Ci), numerically.

y_diff = diff(y_fit);
y_diff = [0,y_diff];

[M,I] = min(y_diff);
inflection = [x_data2(I),y_fit(I)];

scatter(x_data2(I),y_fit(I),'*','red')

x = transpose(x_data2);
```

```
yfit = transpose(y_fit);
ydiff = transpose(y_diff);
results = [x,yfit,ydiff];

% Write results to excel file.
col_header = {'Kd0','ci','s','x_infl (uM)','y_infl
(um3/min)','x','yfit','ydiff'};
row_header = {'starting';'fitted'};
[pathstr,name,ext] = fileparts(filename);
name = [name,'_fit.xls'];
xlswrite(name,col_header,'Sheet1','B1');
xlswrite(name,row_header,'Sheet1','A2');
xlswrite(name,inflection, 'Sheet1','E2');
xlswrite(name,par_start,'Sheet1','B2');
xlswrite(name,par_fit,'Sheet1','B3');
xlswrite(name,results, 'Sheet1','G2');

end
```

[1] Budke, C.; Heggemann, C.; Koch, M.; Sewald, N.; Koop, T., Ice recrystallization kinetics in the presence of synthetic antifreeze glycoprotein analogues using the framework of LSW theory. *The Journal of Physical Chemistry B* **2009,** 113, (9), 2865-2873.

[2] Olijve, L. L.; Meister, K.; DeVries, A. L.; Duman, J. G.; Guo, S.; Bakker, H. J.; Voets, I. K., Blocking rapid ice crystal growth through nonbasal plane adsorption of antifreeze proteins. *Proceedings of the National Academy of Sciences* **2016**, 201524109.

[3] Meister, K.; Strazdaite, S.; DeVries, A. L.; Lotze, S.; Olijve, L. L.; Voets, I. K.; Bakker, H. J., Observation of ice-like water layers at an aqueous protein surface. *Proceedings of the National Academy of Sciences* **2014,** 111, (50), 17732-17736.