Custom Layers Example for Supported DSPC layer.

Example of using alternative languages to make a Custom layers to model a DSPC supportd bilayer.

First, we'll run the project using a conventional Matlab custom model. Then, we will do the same calculation using the same model, but written in Python or C++.

1. Normal Matlab Custom Model.

Start by making the class and setting it to a custom layers type:

```
problem = projectClass('Orso lipid example - custom layers');
problem.setModelType(modelTypes.CustomLayers.value);
problem.setGeometry(geometryOptions.SubstrateLiquid.value);
```

First we need to set up a parameters group. We will be using a pre-prepared custom model file, (at the end of the worksheet). Use this to set up the parameters block...

We need to add the relevant parameters we are going to need to define the model (note that Substrate Roughness' always exists as parameter 1..

```
Parameters = {
    %       Name                min         val         max     fit?
        {'Oxide thick',         5,          20,         60,     true   };
        {'Oxide Hydration'      0,          0.2,        0.5,    true   };
        {'Lipid APM'            45          55          65      true   };
        {'Head Hydration'       0           0.2         0.5     true   };
        {'Bilayer Hydration'    0           0.1         0.2     true   };
        {'Bilayer Roughness'    2           4           8       true   };
        {'Water Thickness'      0           2           10      true   };
        };

 problem.addParameterGroup(Parameters);
 problem.setParameter(1,'min',1,'max',10); % Change the substrate roughness
limits
```

Need to add the relevant Bulk SLD's. Change the bulk in from air to silicon, and add two additional water contrasts:

```
% Change bulk in from air to silicon....
problem.setBulkIn(1,'name','Silicon','min',2.07e-6,'value',2.073e-6,'max',2.0
8e-6,'fit',false);

% Add two more values for bulk out....
problem.addBulkOut('SLD SMW',1e-6,2.073e-6,3e-6,true);
```

```
problem.addBulkOut('SLD H2O',-0.6e-6,-0.56e-6,-0.3e-6,true);

problem.setBulkOut(1,'fit',true,'min',5e-6);
```

Now add the datafiles. We have three datasets we need to consider - the bilayer against D2O, Silicon Matched water and H2O. Load these datafiles in and put them in the data block....

```
% Read in the datafiles
root = getappdata(0, 'root');
D2O_data = dlmread(fullfile(root, '/examples/normalReflectivity/customLayers/
c_PLP0016596.dat'));
SMW_data = dlmread(fullfile(root, '/examples/normalReflectivity/customLayers/
c_PLP0016601.dat'));
H2O_data = dlmread(fullfile(root, '/examples/normalReflectivity/customLayers/
c_PLP0016607.dat'));

% Add the data to the project
problem.addData('Bilayer / D2O', D2O_data(:,1:3));
problem.addData('Bilayer / SMW', SMW_data(:,1:3));
problem.addData('Bilayer / H2O', H2O_data(:,1:3));

problem.setData(2,'dataRange',[0.013 0.37]);
problem.setData(3,'dataRange',[0.013 0.37]);
```

```
Warning: dataRange(2) can't be more than max data value - resetting to 3.299600e-01
```

```
problem.setData(4,'dataRange',[0.013 0.37]);
```

```
Warning: dataRange(2) can't be more than max data value - resetting to 3.304800e-01
```

Add the custom file to the project....

```
%                          name          filename        language    path
problem.addCustomFile('DSPC Model', 'customBilayer.m',
supportedLanguages.Matlab, pwd);
```

Also, add the relevant background parameters - one each for each contrast:

```
% Change the name of the existing parameters to refer to D2O
problem.setBackgroundParam(1,'name','Backs par
D2O','fit',true,'min',1e-10,'max',1e-5,'val',1e-6);

% Add two new backs parameters for the other two..
problem.addBackgroundParam('Backs par SMW',1e-10,1e-6,1e-5,true);
problem.addBackgroundParam('Backs par H2O',1e-10,1e-6,1e-5,true);

% And add the two new constant backgrounds..
```

```
problem.addBackground('Background SMW','constant','Backs par SMW');
problem.addBackground('Background H2O','constant','Backs par H2O');


% And edit the other one....
problem.setBackgroundName(1,'Background D2O');
problem.setBackground(1,'source','Backs par D2O');

% Finally modify some of the other parameters to be more suitable values
% for a solid / liquid experiment.

% Set the scalefactor...
problem.setScalefactor(1,'Value',1,'min',0.5,'max',2,'fit',true);


% Finally modify some of the other parameters to be more suitable values
% for a solid / liquid experiment.

% Set the scalefactor...
problem.setScalefactor(1,'Value',1,'min',0.5,'max',2,'fit',true);
```

Now add the three contrasts as before:

```
% D2O contrast..
problem.addContrast('name',         'Bilayer / D2O',...
                    'background',   'Background D2O',...
                    'resolution',   'Resolution 1',...
                    'scalefactor',  'Scalefactor 1',...
                    'BulkOut',      'SLD D2O',...
                    'BulkIn',       'Silicon',...
                    'data',         'Bilayer / D2O');

% SMW contrast..
problem.addContrast('name',         'Bilayer / SMW',...
                    'background',   'Background SMW',...
                    'resolution',   'Resolution 1',...
                    'scalefactor',  'Scalefactor 1',...
                    'BulkOut',      'SLD SMW',...
                    'BulkIn',       'Silicon',...
                    'data',         'Bilayer / SMW');

% SMW contrast..
problem.addContrast('name',         'Bilayer / H2O',...
                    'background',   'Background H2O',...
                    'resolution',   'Resolution 1',...
                    'scalefactor',  'Scalefactor 1',...
                    'BulkOut',      'SLD H2O',...
                    'BulkIn',       'Silicon',...
                    'data',         'Bilayer / H2O');
```

And set the model for each..

```
problem.setContrastModel(1,'DSPC Model');
problem.setContrastModel(2,'DSPC Model');
problem.setContrastModel(3,'DSPC Model');
```

Look at the complete model definition before sending it to RAT;

```
disp(problem)
```

```
     modelType: 'custom layers'
experimentName: 'Orso lipid example - custom layers'
      geometry: 'substrate/liquid'


Parameters: ------------------------------------------------------------------------------------------

p            Name             Min    Value    Max    Fit?
_    _____________________    ___    _____    ___    _____

1    "Substrate Roughness"     1        3      10    true
2    "Oxide thick"             5       20      60    true
3    "Oxide Hydration"         0      0.2     0.5    true
4    "Lipid APM"              45       55      65    true
5    "Head Hydration"          0      0.2     0.5    true
6    "Bilayer Hydration"       0      0.1     0.2    true
7    "Bilayer Roughness"       2        4       8    true
8    "Water Thickness"         0        2      10    true


Bulk In: ---------------------------------------------------------------------------------------------

p      Name          Min          Value         Max        Fit?
_    _________    _________    _________    _________    _____

1    "Silicon"    2.07e-06    2.073e-06    2.08e-06    false


Bulk Out: --------------------------------------------------------------------------------------------

p      Name         Min         Value         Max        Fit?
_    _________    ______    _________    _________    _____

1    "SLD D2O"     5e-06     6.35e-06    6.35e-06    true
2    "SLD SMW"     1e-06    2.073e-06       3e-06    true
3    "SLD H2O"    -6e-07     -5.6e-07      -3e-07    true


Scalefactors: ----------------------------------------------------------------------------------------

p         Name           Min    Value    Max    Fit?
_    _______________    ___    _____    ___    _____

1    "Scalefactor 1"    0.5      1       2     true


Backgrounds: -----------------------------------------------------------------------------------------

(a) Background Parameters:
```

p	Name	Min	Value	Max	Fit?
1	"Backs par D2O"	1e-10	1e-06	1e-05	true
2	"Backs par SMW"	1e-10	1e-06	1e-05	true
3	"Backs par H2O"	1e-10	1e-06	1e-05	true

(b) Backgrounds:

p	Name	Type	Source	Value 1	Value 2	Value 3	Value 4
1	"Background D2O"	"constant"	"Backs par D2O"	""	""	""	""
2	"Background SMW"	"constant"	"Backs par SMW"	""	""	""	""
3	"Background H2O"	"constant"	"Backs par H2O"	""	""	""	""

Resolutions: ----------

(a) Resolutions Parameters:

p	Name	Min	Value	Max	Fit?
1	"Resolution par 1"	0.01	0.03	0.05	false

(b) Resolutions:

p	Name	Type	Source	Value 1	Value 2	Value 3	Value 4
1	"Resolution 1"	"constant"	"Resolution par 1"	""	""	""	""

Custom Files: ----------

Name	Filename	Function Name	Language	Path
"DSPC Model"	"customBilayer.m"	"-"	"matlab"	"...ocs/API/examples/miscellaneous/a

Data: ----------

Name	Data	Data Range	Simulation Range
"Simulation"	"No Data"	"-"	"[0.0050 , 0.7000]"
"Bilayer / D2O"	"Data array: [146 x 3]"	"[0.0130 , 0.3700]"	"[0.0057 , 0.3961]"
"Bilayer / SMW"	"Data array: [97 x 3]"	"[0.0130 , 0.3300]"	"[0.0076 , 0.3300]"
"Bilayer / H2O"	"Data array: [104 x 3]"	"[0.0130 , 0.3305]"	"[0.0063 , 0.3305]"

Contrasts: ----------

p	1	2	3
"Name"	"Bilayer / D2O"	"Bilayer / SMW"	"Bilayer / H2O"
"Data"	"Bilayer / D2O"	"Bilayer / SMW"	"Bilayer / H2O"
"Background"	"Background D2O"	"Background SMW"	"Background H2O"
"Background Action"	"add"	"add"	"add"
"Bulk in"	"Silicon"	"Silicon"	"Silicon"
"Bulk out"	"SLD D2O"	"SLD SMW"	"SLD H2O"

```
"Scalefactor"          "Scalefactor 1"     "Scalefactor 1"     "Scalefactor 1"
"Resolution"           "Resolution 1"      "Resolution 1"      "Resolution 1"
"Resample"             "false"             "false"             "false"
"Model"                "DSPC Model"        "DSPC Model"        "DSPC Model"
```

Make a controls block....

```
controls = controlsClass();
controls.procedure = 'DE';
controls.display = 'final';
controls.parallel = 'contrasts';
```

And send this to RAT...

```
[problem,results] = RAT(problem,controls);
```

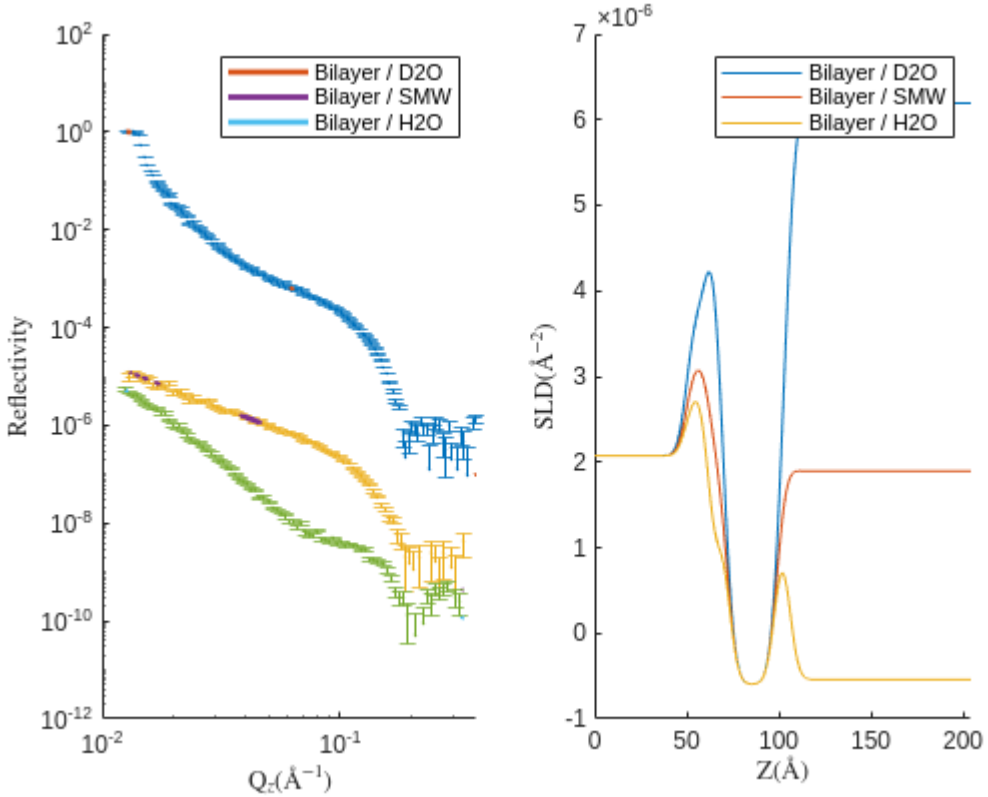
```
Starting RAT ___________________________________________________________________________________________


Running Differential Evolution

Final chi squared is 5.99325
Elapsed time is 1.944532 seconds.

Finished RAT ___________________________________________________________________________________________
```

```
plotRefSLD(problem,results);
```



Using a Python Custom Model.

RAT also allows the use of Python custom models (instead of Matlab). The purpose of this is because it is more logical to work with Python custom functions when working with the Python API for RAT, but also since there may be existing libraries in Python which the user might want to use in a custom model.

The format of the custom model is very similar to the Matlab version. In this example, our model is called customBilayer.py..

```
type customBilayer.py
```

```
Warning: Negative data ignored
# customBilayer.py
import numpy as np

def customBilayer(params, bulk_in, bulk_out, contrast):
    params = params
    bulk_in = bulk_in
    bulk_out = bulk_out

    sub_rough = params[0]
    oxide_thick = params[1]
    oxide_hydration = params[2]
    lipidAPM = params[3]
    headHydration = params[4]
    bilayerHydration = params[5]
    bilayerRough = params[6]
    waterThick = params[7]

    # We have a constant SLD for the bilayer
    oxide_SLD = 3.41e-6

    # Now make the lipid layers..
    # Use known lipid volume and compositions
    # to make the layers

    # define all the neutron b's.
    bc = 0.6646e-4     # Carbon
    bo = 0.5843e-4     # Oxygen
    bh = -0.3739e-4    # Hydrogen
    bp = 0.513e-4      # Phosphorus
    bn = 0.936e-4      # Nitrogen
    bd = 0.6671e-4     # Deuterium

    # Now make the lipid groups..
    COO = (4*bo) + (2*bc)
    GLYC = (3*bc) + (5*bh)
    CH3 = (2*bc) + (6*bh)
    PO4 = (1*bp) + (4*bo)
    CH2 = (1*bc) + (2*bh)
    CHOL = (5*bc) + (12*bh) + (1*bn)

    # Group these into heads and tails:
    Head = CHOL + PO4 + GLYC + COO
    Tails = (34*CH2) + (2*CH3)

    # We need volumes for each.
    # Use literature values:
    vHead = 319
    vTail = 782
```

```
    # we use the volumes to calculate the SLD's
    SLDhead = Head / vHead
    SLDtail = Tails / vTail

    # We calculate the layer thickness' from
    # the volumes and the APM...
    headThick = vHead / lipidAPM
    tailThick = vTail / lipidAPM

    # Manually deal with hydration for layers in
    # this example.
    oxSLD = (oxide_hydration * bulk_out[contrast]) + ((1 - oxide_hydration) * oxide_SLD)
    headSLD = (headHydration * bulk_out[contrast]) + ((1 - headHydration) * SLDhead)
    tailSLD = (bilayerHydration * bulk_out[contrast]) + ((1 - bilayerHydration) * SLDtail)

    # Make the layers
    oxide = [oxide_thick, oxSLD, sub_rough]
    water = [waterThick, bulk_out[contrast], bilayerRough]
    head = [headThick, headSLD, bilayerRough]
    tail = [tailThick, tailSLD, bilayerRough]

    output = np.array([oxide, water, head, tail, tail, head])

    return output, sub_rough
```

We add this to the project in exactly the same way as a Matlab cutom model....

```
problem.addCustomFile('pyDSPC','customBilayer.py','Python',pwd);
```

...then set the models of our contrasts accordingly....

```
for i = 1:3
    problem.setContrastModel(i,'pyDSPC');
end
```