NASA OpenSSP Web Interface

1   Introduction

Following the successful Tropical Rainfall Measuring Mission (TRMM), NASA and JAXA have collaborated again on the Global Precipitation Mission (GPM) to extend satellite precipitation remote sensing to higher latitude zones beyond the tropics. The probability of observing snowfall and thus irregular solid hydrometeors has greatly increased. To improve the physical precipitation retrieval involving these complicated particles, Kuo et al (2016) have created an extensive database, named NASA OpenSSP database (OpenSSP in short), of ~10,000 realistic synthetic (rimeless) solid hydrometeors and their orientation-averaged non-polarimetric single scattering properties (SSPs) at 15 microwave frequencies.

The complexity and volume of the OpenSSP database are both considerable, yet users are often interested in only a subset of the data, e.g. a subset of frequencies and/or hydrometeor types etc. It has become evident that it is wasteful and counterproductive to require every user to download the entire database and to understand the semantics of its netCDF packaging before making use of the data. **NASA OpenSSP Web Interface** is therefore created to make it easier for users to access and obtain targeted subsets of data from the database.

The queries submitted through the OpenSSP Web Interface are basically composed of two elements:

1) Particle selection - subsetting the synthetic hydrometeors contained in the OpenSSP database and
2) Frequency selection - subsetting the 15 microwave frequencies (or bands), of which single-scattering properties are wanted.

The OpenSSP Web Interface supports a few popular ways that particle selection is done. These ways are therefore implemented in different sub-interfaces of the browser interface (see paragraph below).

OpenSSP Web Interface is actually comprised of two interrelated interfaces, i.e.

1) a browser interface named "OpenSSP Query Builder" and
2) a web service application programming interface (API).

The browser interface is designed and constructed to facilitate intuitive *interactive* interrogation, extraction, and access of the data contained in the OpenSSP database, whereas the web service API affords better automation to accomplish the same. Since there are a few distinctive ways the particle subsetting can be done, several browser (sub-)interfaces are implemented on different tabs to accommodate them.   Interactive requests constructed through the browser interface,

however, are translated in real time into web service calls and displayed near the bottom of the browser interface. As such, the browser interface serves as a training tool for building web API queries; it is thus also called a *query builder* as described in Section 2. The web service API is described in Section 3.

What this is really saying is that the interactive query interface allows the user to construct a query that could be submitted through an API client, to get the same results. This needs to be tested.

2 The Browser Interface - *OpenSSP Query Builder*

This section describes the browser interface, also known as OpenSSP Query Builder. **Subsection 2.1** describes how to gain access to the interface. The rest of the subsections correspond to the six tabs of the interface, i.e. **Pristine**, **Aggregate**, **Modeled PSD**, **m-D & PSD**, **Scattering Data Package**, and **Particle Structure**.

2.1 Gaining Access

A registered email address is required for accessing the **OpenSSP Query Builder** browser interface at

https://storm.pps.eosdis.nasa.gov/storm/OpenSSP.jsp

The "Register here" link (in the red oval in **Fig. XXX**) to the right of the email address input field on the web page is for such a purpose. The registration process is simple and lightweight, after which just the registered email address is needed to gain access to the interface, i.e. no password is required.

The interface is named **OpenSSP Query Builder** because it not only constructs the queries based on user interactions with the browser interface, but also assists the user in learning and building the queries that can be used as web service calls (i.e API Queries) in scripts to automate OpenSSP data access.

Successful login starts the present session. A web page like **Fig. XXX** becomes visible. The "Active Selection Area" takes up most of the interface space. This is the primary area where user interactions with the browser interface take place.

The system-generated Session ID for the present session, which is customizable, is displayed above the Active Selection Area. By default, the Session ID is a text string containing a timestamp with millisecond resolution; when used together with the registered email address it forms a system-wide unique ID of the session. Users can modify the Session ID by changing the text string to suit their own preference, e.g. organizing queries for the same study/purpose. However, when it is customized, its system-wide uniqueness is no longer guaranteed.

Below the Active Selection Area shows the actual API Query Call that will be issued to the system for execution when the user hits the "Submit API Query" or "Email Data" button. The API

Query Call is revised dynamically according to user interactions with the browser interface. (The registered email in the API Query Call is blotted out in the figure.)

There are presently six (6) active tabs in the Active Selection Area corresponding to six (6) sub-interfaces, **Pristine**, **Aggregate**, **Modeled PSD**, **m-D & PSD**, **Scattering Data Package**, and **Particle Structure**. Each tab implements a specific (sub-)interface supporting a distinct manner of access independent of the others, as detailed in sections below.
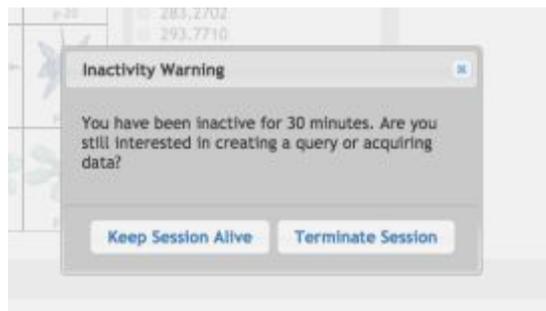
After 25 minutes of inactivity, a Inactivity Warning text box (see **Fig. XXX**) will pop up to solicit a response. If no response is given in additional 5 minutes, the current session terminates.

2.2 Pristine

The **Pristine** interface (see **Fig. XXX**) allows the subsetting of individual pristine hydrometeors by Type in OpenSSP. It features four columns with the following headings: Type, Size (in μm), Selected, and Frequency (in GHz). Trailing the text of each column heading is a question-mark icon (🔵), clicking which pops up a text box with information regarding the content or meaning of the column. This is called a "tool tip" or "help tip", which is implemented liberally throughout the browser interface to offer real-time assistance and enhanced usability. The subsections below describe the purposes and functions of these columns.

2.2.1 Type[1]

The pristine hydrometeors in OpenSSP are generated using an improved and parallelized version of the cellular automaton model (nicknamed "*Snowfake*") of Gravener and Griffeath (2009, GG09 hereafter) that simulates the vapor deposition growth of atmospheric ice crystals. During the growth simulation, a snapshot of the particle structure is written out to a file every 500 iterations, containing a byte array in binary form. (These particle structure data can be obtained from the "**Particle Structure**" tab.) Thus, out of the growth history of one pristine type, these snapshots provide a range of varying particle sizes. Thumbnail images in the Type column are extracted from the figures of GG09 paper, each representing a collection of synthetic hydrometeors of one pristine type featured in OpenSSP. Clicking on one of these thumbnail images highlights the image in yellow and selects the collection. The Size column is populated accordingly with size values accompanied by checkboxes (see next subsection).

---

[1] "Type" is used instead of "habit" because the geometry characteristic that "habit" refers to may change during crystal growth. For example, a pristine crystal may start out as a plate (a habit) but grow into a dendrite (another habit).

More than one thumbnail image (i.e. pristine type collection) can be selected per query. Clicking the thumbnail image toggles the selection. That is, clicking on a deselected thumbnail image selects it and clicking on an already selected (highlighted) image deselects it.

2.2.2 Size

Once a thumbnail image is selected in the Type column, the Size column is populated accordingly with the sizes of particles belonging to the selected type, expressed in *liquid equivalent volume diameter*[2] (aka *mass equivalent diameter*, in μm, i.e. micron), of the particles belonging to the selected type in the database. The sizes in the Size column update automatically when more types are selected (or deselected). Clicking the checkbox beside a size selects it and adds it to the Selected column. Clicking the checkbox of a selected size deselects it. No more than 20 sizes can be selected individually in this manner[3]. A "Select All" checkbox is available near the top of the column, checking which selects all the sizes of all selected types.

Once Type and Size selections are made, the relevant part of the corresponding API Query Call is updated, in the form of "size=p-??/*.*um", where "p-??" ("p-" followed by two decimal digits) is the identifier of the pristine type and "*.*um" is the radius (in μm) of the *ice sphere equivalent volume*[4] (the r_eq_vol column of the query results; see subsection **2.2.5**). If the "Select All" checkbox is checked, the size portion of the API Query Call condenses to "size=p-??/all". (The registered email address used to get the figure has been blotted out.)

When the OpenSSP database was created, the *ice sphere equivalent volume radius* was used and remained as the primary index for particle sizes, which is thus used by the API Query Call to search the database directly. The community, however, is more accustomed to using liquid equivalent volume diameter (or, mass equivalent diameter) as particle size index. Therefore, the OpenSSP browser interface opts for using the latter and requires a conversion to construct the API Query Call:

$$D_{mass} = \sqrt[3]{\frac{6\rho_{ice}V_{ice}}{\pi}} = \sqrt[3]{8\rho_{ice}r^3} = 2r\sqrt[3]{\rho_{ice}}$$

---

[2] Liquid equivalent volume diameter: The diameter of the spherical drop when the ice is melted into water.
[3] The limit of 20 sizes is set to avoid the API Query call becoming too long and unwieldy.
[4] Ice sphere equivalent volume radius: The radius of the ice sphere with the same volume as the irregular ice hydrometeor.

where D_{mass} is the liquid equivalent volume diameter, r the ice sphere equivalent volume radius, and $\rho_{ice} = 0.916\,\text{g cm}^{-3}$ the density of ice.

2.2.3 Selected

The Selected column lists all of the items selected in both the Size and Frequency columns in numerical order. This column is implemented to minimize the need to scroll through the Size column to deselect selected sizes. A selection can be removed from the column by clicking the underlined "X" next to it. The corresponding checkboxes in their respective columns will be cleared accordingly. The API Query Call below will also reflect the change(s). If the "Select All" button is checked in the Size column, the word "All" will be placed in the Selected column, similarly with an underlined "X" next to it.

2.2.4 Frequency

The Frequency column lists all 15 microwave frequencies for which scattering properties have been calculated for every hydrometeor. These are chosen to be close to frequencies used by microwave imager/sounder instruments on various satellite platforms. Individual frequencies can be selected (deselected) by clicking on the checkboxes next to them. Selected frequencies are also added to the API Query Call.

Similarly, checking (unchecking) the "Select All" checkbox will select (deselect) all the frequencies. The relevant part of the API Query Call reflects the selection(s) and change(s) accordingly.

2.2.5 Submit Query

Once there is at least one selection in each of the Shape, Size, and Frequency categories, the "Submit API Query" and "Email Data" buttons become functional, which, when pressed, execute the API query.

- If "Submit API Query" is pressed, the results of the query are displayed in another browser page (or tab) as "tab separated values" (tsv), with "Download As Text File" buttons available at the top and the bottom of the page, which, when pressed, downloads a text file with ".tsv" extension. **Fig. XXX** shows an example. There are two groups of lines starting with a "#" character at the top and bottom, containing descriptive information regarding the data section sandwiched in between them. The group at the top, referred to as "header" henceforth, records the version of the OpenSSP API, the registered email address used to submit the query, the Session ID, and the "request string"[5] of the API Query Call, whereas the group at the bottom, referred to as "footer" henceforth, explains the column headings of the data section.
- If "Email Data" button is pressed, the results of the query are emailed, from the "Help Desk" email account of PPS, to the registered email address of the session as a "tab

---

[5] One needs to concatenate the "domain", i.e. "https://storm.pps.eosdis.nasa.gov/storm/", with the "request string" to form a valid web service call.

separated values (tsv)" file in a **gzip**'ed attachment with the suffix ".tsv.gz". Although the filename is meaningless to the user (automatically generated to ensure uniqueness), the subject of the email message contains the Session ID.

In both cases, the ".tsv" file can be readily imported into, say, Microsoft Excel or other spreadsheet software. The "**Email Data**" approach is preferred when the data volume of the request is expected to be large, e.g. many hydrometeors, sizes, or frequencies. In addition, in either case, users can start building the next query using while waiting for the query results.

2.2.6 Email Session Queries
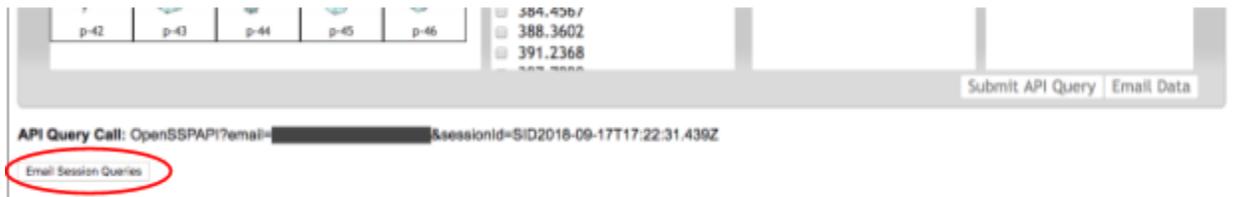
After a (at least one) **API Query Call** is submitted, the "Email Session Queries" button appears below the **API Query Call** line under the Active Selection Area (**Fig. XXX**). Pressing the "Email Session Queries" button will cause

1. An email message containing the Session ID and a list of all submitted queries of the session to be sent to the registered email used for the session (see **Fig. XXX**), and
2. The conclusion of the present session and the start of a new session with a new **Session ID**.

Once the email message is sent, the "Email Session Queries" disappears until a new API Query is submitted.

2.2.7 Pristine Interface Tutorial

Please be reminded that a registered email address is needed to access the **OpenSSP Query Builder** browser interface. See Section 2.1 for the registration process. Once a registered email address is entered, a (default but customizable) unique **Session ID** is assigned and the interface opens with the **Pristine** tab being active.

To construct a **Pristine** query, both particle and frequence selections must be made. Clicking on a thumbnail image (or more) in the Type column selects a pristine type. When a type is selected, its sizes are populated in the Size column. Up to 20 sizes can be selected by clicking, hence checking, the checkboxes next to the size values. Selected sizes can be deselected by clicking, hence unchecking, the filled checkboxes or by clicking the "X" next to the values in the Selected column.

Frequencies can be selected at any time by clicking, hence checking, the checkbox next to the desired frequency value. A selected frequency can be deselected by clicking, hence unchecking, the filled checkbox or the "X" next to the value in the Selected column. Any number of frequencies can be requested. As the selections are made or revised, the "API Query Call" field below the Active Selection Area will adjust accordingly so that the call can be copy-pasted if necessary.

Once all required elements are specified, clicking the "Submit API Query" button pops up a new browser window (or tab) displaying the query results. If the query is complex, e.g. involving many particles or frequencies, it may take a minute or two for the results to show up. It is also possible to request the query results be emailed to the registered email address by clicking the "Email Data" button.

After the submission of the first query of the present session, the "Email Session Queries" button will appear below the API Query Call field on the browser interface. Pressing the button ends the present session, starts a new session with a different Session ID, and causes an email message to be sent to the registered email address. The message will contain the just ended Session ID and a list of all submitted queries during that session.

2.3 Aggregate

Most of the naturally occurring snowflakes are in the form of aggregates, i.e. collections of aggregated pristine crystals. Thus, the particles in OpenSSP has an aggregate-to-pristine ratio of

approximately 7:2, i.e. 7039 aggregates and 2014 pristine crystals. The **Aggregate** interface is almost identical to the **Pristine** interface except for a couple of distinctions:

1. It has, however, fewer options (thumbnail images) in the Family column, because only nine (9) pristine types were used to generate nine (9) corresponding families of aggregates. The component particles of every aggregate in the same family are all from the same pristine type, which is often referred to as "habit". Thus, these aggregates are *mono-habit aggregates*.
2. There is an extra column, titled "Aggregate Examples", with Help information and sample images of aggregates generated by aggregating pristine crystals corresponding to the selected thumbnail image(s).

Otherwise, the selection process is identical to that of Pristine.

Additionally, in a query for aggregates, when the "Select All" checkbox is checked, there will be multiple aggregate collection identifiers (in the form of a-*dddd*, where each *d* is a decimal number between 0 and 9) following the "size=" argument in the API Query Call. This is because there are multiple aggregation experiments for a given pristine type. This also means that there are many more sizes in an **Aggregate** "Select All" query than in a similar **Pristine** query. As such, returning these results may take significantly longer.

> Help Text (Different from 1.2)
>
> Family: These shapes originate from a paper by Janko Gravner and David Griffeath, the developers of the "snowfake" algorithm that is used to generate synthetic snowflakes in OpenSSP. Aggregate families refer to the pristine synthetic snowflakes that are impacted together to generate crystals.
>
> Aggregates: These images represent examples of the crystals aggregated from the selected type.

2.3.1 Aggregate Interface Tutorial

Please be reminded that a registered email address is needed to access the **OpenSSP Query Builder** browser interface. See **Section 2.1** for the registration process. Once a registered email address is entered, a (default but customizable) unique Session ID is assigned and the interface opens with the **Pristine** tab being active.
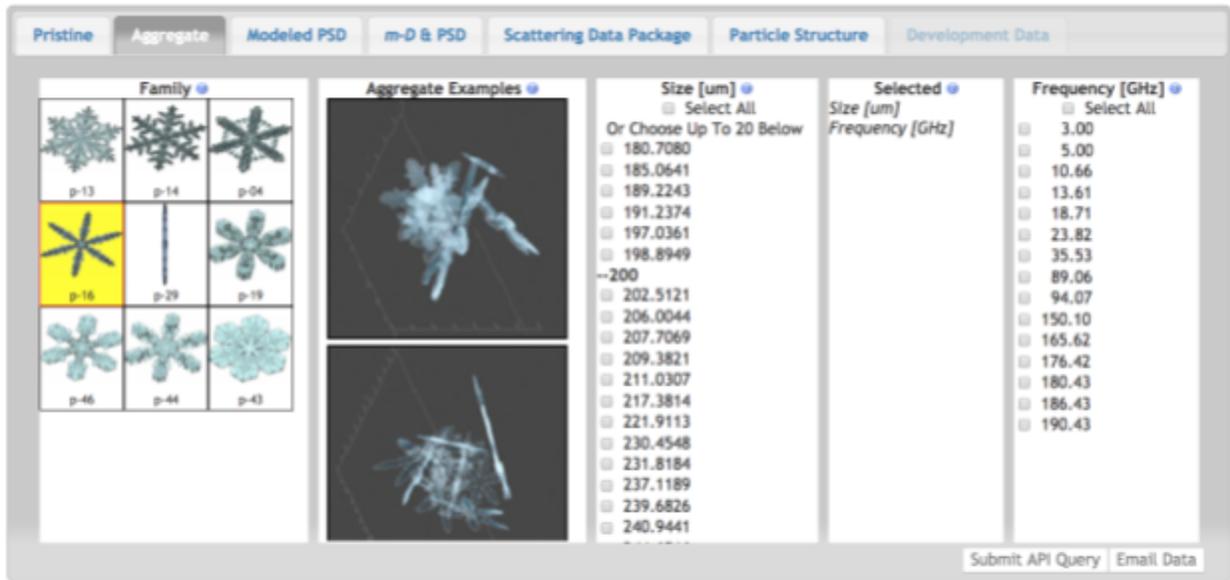
Once the interface appears, click on the **Aggregate** tab to select it. Particle selection is made by clicking on a thumbnail image (or more) in the Family column. The aggregate families are ordered in the same way as they appear in Kuo et al. (2016). Upon selecting an aggregate family, its corresponding set of example images appear in the "Aggregate Examples" column.

Corresponding sizes appear in Size column upon the selection of an aggregate family. Up to 20 sizes can be selected individually by checking the checkboxes next to the size values. Selected

sizes can be deselected by unchecking the filled checkboxes or by clicking the "X" next to the values in the Selected column.
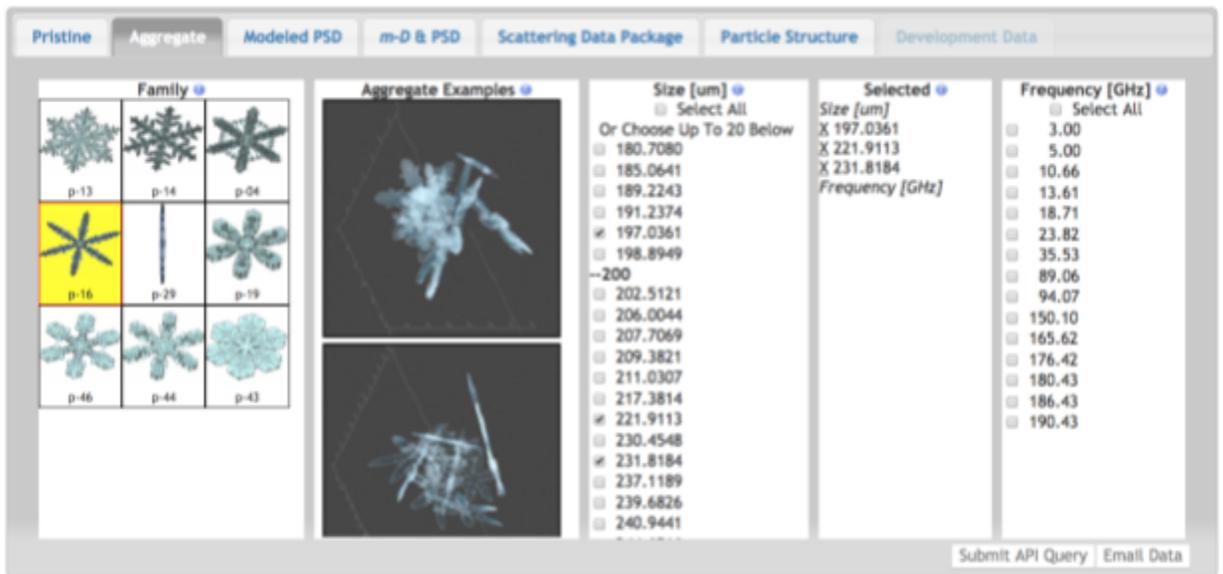
API Query Call: OpenSSPAPI?email=▬▬▬▬▬&frequency=035.525GHz,094.065GHz,165.615GHz&size=a-0081/000101.460403um,a-0086/000114.269501um,a-0081/000119.371002um&sessionId=SID2018-09-17T17:22:31.439Z

Frequencies can be selected at any time by checking the checkboxes next to the desired frequency values. A selected frequency can be deselected by unchecking the filled checkbox or the "X" next to the value in the Selected column. Any number of frequencies may be requested.

As the selections are made or revised, the "API Query Call" field below the Active Selection Area will adjust accordingly so that the call can be copy-pasted if necessary. Once all required elements are specified, clicking the "Submit API Query" button pops up a new browser window (or tab) displaying the query results. If the query is complex, e.g. involving many particles or

frequencies, it may take a minute or two for the results to show up. It is also possible to request the query results be emailed to the registered email address by clicking the "Email Data" button.

Similarly, if this is the first query of the present session, after its submission the "Email Session Queries" button will appear below the API Query Call field on the browser interface (see **Fig. XXX**). Pressing the button ends the present session, starts a new session with a different Session ID, and causes an email message to be sent to the registered email address. The message will contain the just ended Session ID and a list of all submitted queries during that session.

2.4 Modeled PSD

Most of the radiative transfer simulations require SSPs of an ensemble of hydrometeors, rather than those of individual hydrometers. Since the ensemble are usually constrained by particle size distributions (PSDs), we implemented the "**Modeled PSD**" interface to make obtaining the SSPs of an ensemble convenient.

API Query Call: OpenSSPAPI?&sessionId=SID2018-09-17T17:22:31.439Z
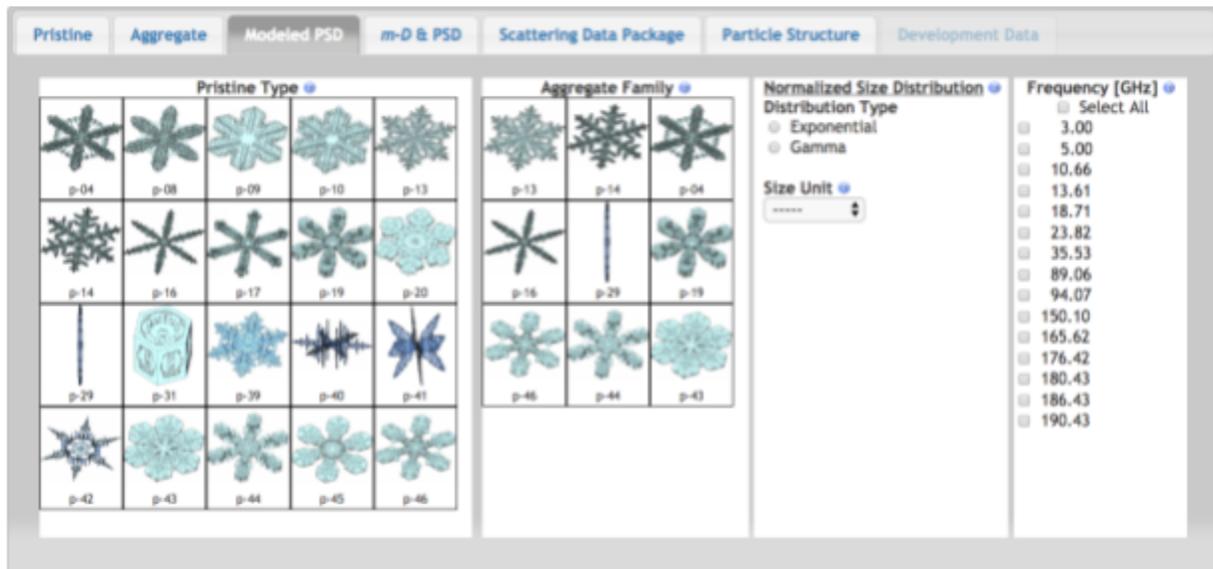
The "**Modeled PSD**" interface is organized into four (4) columns with the first two, i.e. "Pristine Type" and "Aggregate Family", for particle selection, next for "Normalized Size Distribution" specification, and the last for Frequency selection.

2.4.1 "Pristine Type" and "Aggregate Family"

Multiple particle thumbnail images can be selected in "Pristine Type" and/or "Aggregate Family". That is, a particle selection can have a mixture of pristine types and aggregate families. When a thumbnail image is selected and a unit has been specified using the "Size Unit" dropdown menu, the "Minimum Size", "Maximum Size", and "Number of Sizes" fields in the

"Normalized Size Distribution" column are filled. They are updated as more thumbnail images are chosen to reflect the minimum and maximum sizes of the entire selection.

2.4.2 Normalized Size Distribution
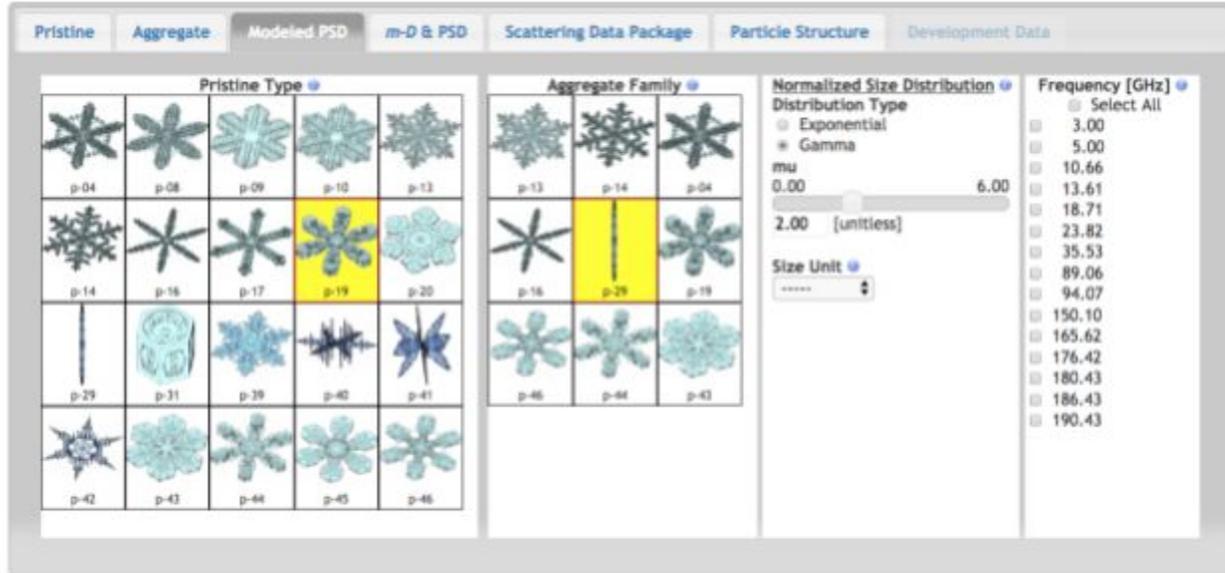
API Query Call: OpenSSPAPI?&sessionId=SID2018-10-03T16:49:40.066Z

A PSD is specified according to the normalized size distribution of Testud et al (2001)[6] as a function of *liquid equivalent volume diameter*, to be consistent with the size in Pristine and Aggregate interfaces. One out of two distribution types, i.e. normalized Exponential and Gamma distributions, can be selected. If normalized Gamma distribution is chosen, a slide bar will appear with a reasonable range for the shape parameter, μ (a dimensionless number), and one must also specify a μ value for the distribution (see **Fig. XXX**).

There is no need to specify N_0^* of the normalized distribution of Testud et al (2001). It is automatically set to normalize the water content of the PSD to 1 g m^{-3} so users can scale the resultant PSD to desirable water content easily.

Three units, i.e. Micron, Millimeter, and Meter, are available from the "Size Unit" dropdown menu to specify the unit for mean (liquid) volume diameter D_m, "Bin Size", "Minimum Size", and "Maximum Size". (The actual sizes of the particles are not affected.) The maximum possible D_m for the chosen particle selection is reflected in the "Maximum D_m" field. If a value greater than the maximum is specified, an error message in red: "D_m value is too large for selected particles" will be displayed beneath the field. "Bin Size" is defaulted to 0.1 mm (equivalently, 100 micron or 1.00e-4 meter) but customizable. It is not recommended, however, to use a bin size smaller

---

[6] The tool tip (accessible by clicking icon next to the column heading) contains a link to the reference.

than 0.1 mm, because it may result in some empty bins, i.e. no particle in the database with size in the range.

The geometric properties and SSPs of the particles in a size bin are arithmetically averaged to represent those for the bin, e.g. the backscatter efficiency of a bin. Integral quantities over the PSD, e.g. water content, is performed using the.trapezoidal rule.

The function of "Submit Inputs" button is described in a later subsection, i.e. **2.4.4**.

2.4.3 Frequency

Frequency selection works exactly the same as that on the **Pristine** or **Aggregate** interface.

2.4.4 Submit Inputs
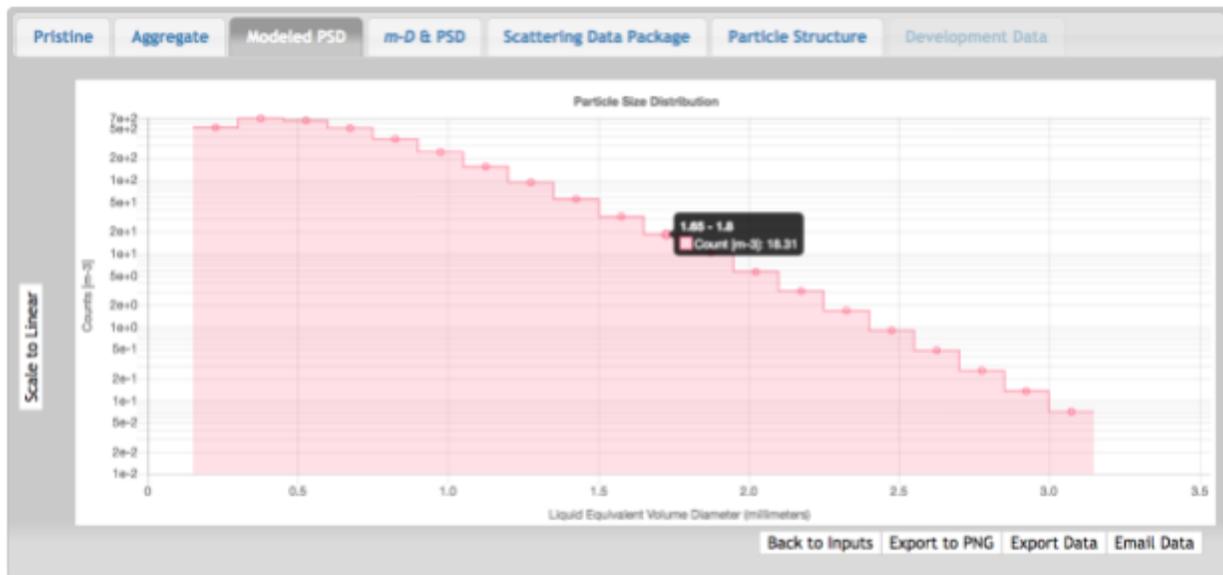
API Query Call: OpenSSPAPI?email=▮▮▮▮▮▮▮▮&size=p-19/all,a-0020/all,a-0021/all,a-0022/all,a-0023/all,a-0024/all&dsubm=1.2&bs=0.15&bunit=millimeters&mu=2&frequency=018.713GHz

Once all of the required inputs have been specified, the "Submit Inputs" button becomes functional, otherwise it remains inactive. Clicking the "Submit Inputs" button triggers the construction of a graph in the Active Selection Area. The graph is in the form of a histogram plotting the particle number count for each size bin, according to the distribution type (i.e. exponential or gamma) chosen (see **Fig XXX**). (To convert the per-size-bin count to concentration, one needs to divide the count by the bin size.) The particle count for each bin can be read off by hovering the cursor over the circle of the corresponding bin in the graph (aka *mousing over*).

The y-axis scale for the graph can be switched from logarithmic (the default) to linear using the button labeled with "Scale to Linear" to the left, or switched back from linear to logarithmic using the same button but labeled with "Scale to Logarithmic".

There are four (4) buttons below the graph: "Back to Inputs", "Export to PNG", "Export Data", and "Email Data", whose functions are self-explanatory:

- Back to Inputs - allows use to go back and adjust the inputs.
- Export to PNG - exports the graph as a PNG file for download.
- Export Data - exports the query results, including SSPs and other geometric properties.
- Email Data - email the query results to the registered email address used for the session.

Similar to the **Pristine** and **Aggregate** interfaces, query results (i.e. Data) are expressed as tab separated values (tsv), which can be easily imported into spreadsheet applications. Each row of the data section corresponds to a size bin. The columns generally contain simple arithmetic means of the property derived from the particles in the corresponding size bins. For example, the backscatter efficiency for the size bin 0.5-0.6 mm is the arithmetic average of backscatter efficiencies of the selected particles with liquid equivalent volume diameters between 0.5 mm and 0.6 mm.

Since this type of queries involves generally more particles and more complicated processing, it could take a few minutes to complete. An alert message is displayed to remind users of this fact, when "Export Data" or "Email Data" is requested.

Help Text

Pristine Type: These shape types originate from a paper by Janko Gravner and David Griffeath, the developers of the "snowfake" algorithm that is used to generate synthetic snowflakes in OpenSSP. As opposed to the aggregates, which are generated from impacting multiple pristine synthetic snowflakes together, pristines are generated simply from water vapor preferentially depositing onto the surface of the snowflake. For the chart, the sizes of pristine crystals generated as well as properties of those crystals are used to generate the bins and are aggregated for the output text.

Aggregate Family: These shapes originate from a paper by Janko Gravner and David Griffeath, the developers of the "snowfake" algorithm that is used to generate synthetic snowflakes in OpenSSP. Aggregate families refer to the pristine synthetic snowflakes that are impacted together to generate crystals. For the chart, the sizes of aggregate crystals generated from these families as well as properties of those crystals are used to generate the bins and are aggregated for the output text.

Normalized Size Distribution: For more information, see Testud, et al. 2001

Size Unit: This field represents the units for the bin size and D_m inputs. It also sets the units for the bins on the graph.

Number Concentration Unit: This field represents the units for the $N_0{}^*$ input.

2.4.5 Modeled PSD Interface Tutorial

Please be reminded that a registered email address is needed to access the **OpenSSP Query Builder** browser interface. See **Section 2.1** for the registration process. Once a registered email address is entered, a (default but customizable) unique Session ID is assigned and the interface opens with the **Pristine** tab being active.
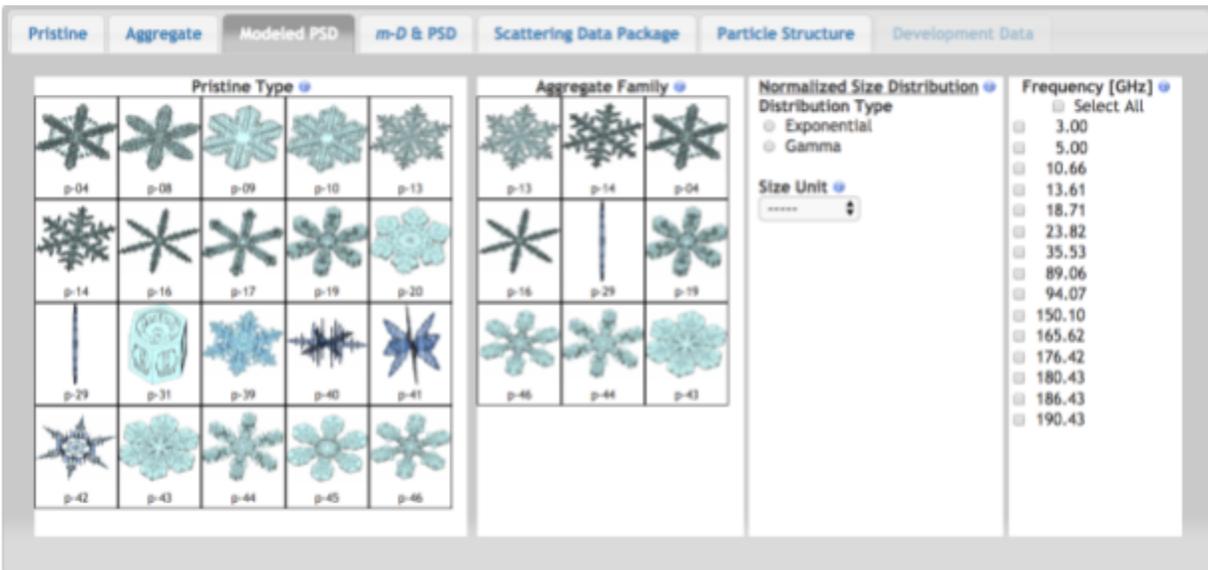
API Query Call: OpenSSPAPI?&sessionId=SID2018-09-17T17:22:31.439Z

Once the interface appears, click on the **Modeled PSD** to select it. There are four (4) columns on the **Modeled PSD** interface, with "Pristine Type" and "Aggregate Family" as the first two for particle selection, "Normalized Size Distribution" next for PSD specification, and Frequency the last for frequency selection (see **Fig. XXX**).
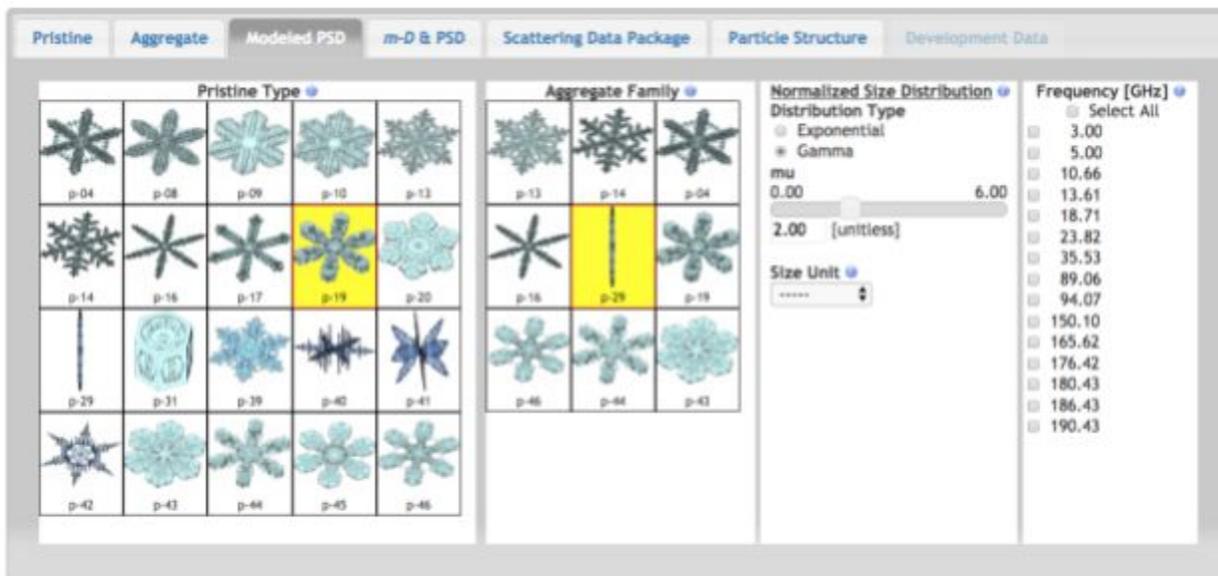
Particle selection is made by clicking on the thumbnail image(s) in Pristine Type and/or Aggregate Family, similar to particle selection in the **Pristine** and **Aggregate** interfaces, but here a selection can contain both Pristine Type and Aggregate Family (see **Fig. XXX**).
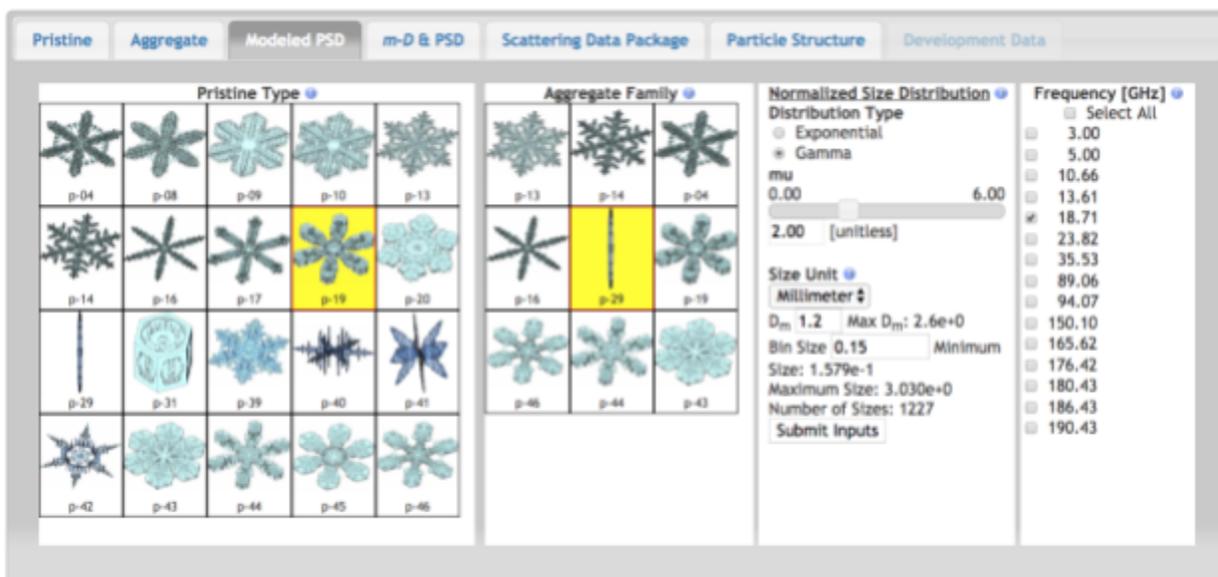
One of two normalized size distributions can be selected, either "Exponential" or "Gamma," using the radial buttons next to them. For this tutorial, Gamma distribution is selected (see **Fig. XXX**). Upon selecting Gamma, a new input field appears for the size distribution shape

parameter, μ, a dimensionless parameter correlated with the shape of the gamma distribution. (No shape parameter is needed for Exponential distribution.)

The desired Size Unit that a user wishes to work with can be specified using the dropdown menu containing the following options: Micron, Millimeter, or Meter. (The actual size of the particles is not affected.) This determines the unit used for the input of D_m and Bin Size (the latter is defaulted to 100 micron = 0.1 mm = 10^{-4} m).

Maximum D_m of the selected particle collection(s) is reported below the D_m input field. Specifying a D_m greater than Maximum D_m results in an error. In fact, it is advised against selecting a D_m too close to Maximum D_m, even if it is smaller than Maximum D_m. Once the Size Unit has been specified, maximum and minimum size values, as well as the number of sizes, of the selected collection(s) will also be reported. D_m = 1.2 mm and Bin Size of 0.15 mm are specified for this tutorial.

One of more frequencies should be selected next by clicking the checkbox(es) next to it/them. Note that the API Query Call remains constant during the interactions above. This is different from the dynamic update in the previous two interfaces.
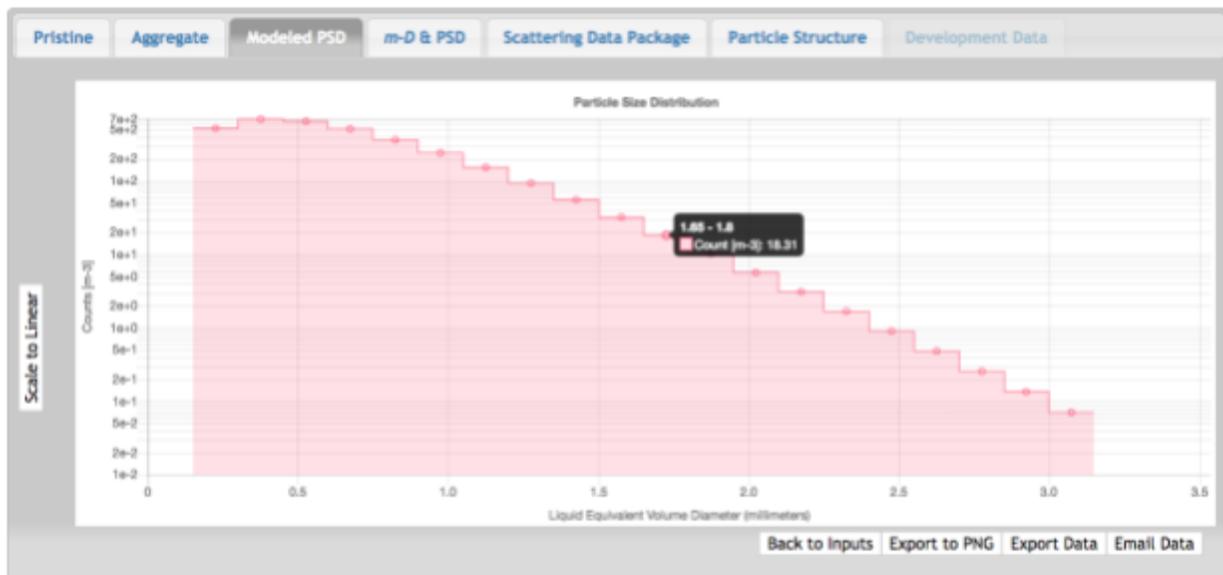
API Query Call: OpenSSPAPI?email=▮▮▮▮▮▮▮▮▮▮▮&size=p-19/all,a-0020/all,a-0021/all,a-0022/all,a-0023/all,a-0024/all&dsubm=1.2&bs=0.15&bunit=millimeters&mu=2&frequency=018.713GHz

Once particle selection, PSD specification, and frequency selection have been made, one can click the "Submit Inputs" button in the Normalized Size Distribution column, to display the graph of the PSD applied to the selected particle collection(s) as a histogram(see **Fig. XXX**). The horizontal bars (line segments) of the histogram mark the counts of particles in the corresponding bins. The number of bins is determined by the Bin Size specified and the difference between maximum and minimum sizes of the selected particle collection(s). The graph supports a couple

interactive features. The button to the left of the y-axis toggles the y-axis scale between linear and logarithmic. Mousing over the circle at the bar atop each bin causes the exact count value to be displayed (which must be divided by the bin size to obtain concentration).

Note that the `API Query Call` below the graph has now been updated according to the choices made to generate the graph. Four buttons, i.e. "Back to Inputs", "Export to PNG", "Export Data", and "Email Data", are available below the graph, whose functions are described in subsection **2.4.4**. The graph can be exported as a PNG image by clicking the "Export to PNG" button. Pressing the "Export Data" button causes query results to be displayed as tab separated values (tsv) in a separate browser page or tab (see **Fig. XXX**).

2.5 m-D & PSD

A snowing volume inevitably contains a mixture of hydrometeors with diverse shapes and sizes. Since it is impossible to distinguish the shape and size compositions in such a volume with radar or radiometer signal, we usually assume the hydrometeors therein follow certain *m-D* relations that are statistically similar to those observed by in situ particle probes. Thus, in retrieval applications, SSPs are usually derived from PSDs subjected to the constraint of mass-dimension (*m-D*) relation(s). At this (sub-)interface, users can specify an *m-D* relation to further constrain a PSD that they also specify.

This interface also sports four (4) columns, but there are no thumbnail images for particle selection. Instead, the first column contains a brief introduction of interface (see **Fig. XXX**). The entire particle database is searched for those particles, pristine or aggregate, that satisfy the *m-D* relation constraint. The specification of PSD in the "Normalized Size Distribution" column is identical to that of the "**Modeled PSD**" interface.

2.5.1 *m-D* Relation

The mass-dimension relation is expressed in the following form

$$m = aD^b .$$

Since this is an empirical relation relating the value of particle maximum dimension D in centimeter (cm) to the value of particle mass m in gram (g), both coefficients a and b are unitless. The initial values for a and b are 0.0070 and 2.20 respectively but can be adjusted using the slide bars in increments of 10^{-5} (between 4.00×10^{-3} and 9.00×10^{-3}) for a and 10^{-2} (between 1.80 and 2.40) for b. These ranges of values are restricted to bounds suggested by observations, but provide sufficient flexibility for exploration. Thus, although a and b are also customizable by directly entering values into the corresponding text fields below, it is not recommended to use values outside the recommended range.

The probability of a particle matching exactly the *m-D* relation as specified is extremely small. Thus, a small deviation from the exact relation must be allowed in practice. This deviation is expressed in percentage of the mass. Naturally, a balance needs to be achieved when specifying the deviation because:

- The smaller the deviation the fewer hydrometeors in the database would likely qualify; the risk of finding no qualifying hydrometeor in some PSD size bins increases.
- The larger the deviation the greater is the risk of departing from the specified *m-D* relation.

Help Text (Different from Modeled PSD)

Mass/Dimension Relationship: There are three variables to define that comprise the mass-dimension relationship in the form $m=aD^b$. The first two have default values that come from observed values discussed in Heymsfield et al. (2010), but can be adjusted depending on observed/theoretical distributions. The third value is the "Mass Percentage Deviation," which is the tolerance for how off the idealized relationship the mass of a particle can be for it to be included in the dataset. Any particle would be included within the range $m-m*(d/100) \leq m < m+m*(d/100)$, where d is this "Mass Percentage Deviation."

Mass Deviation: Because it is statistically unlikely that a particle would lie exactly on the mass-dimension relationship line selected by the user, it is essential that the user define a tolerance interval to include relevant particles. This value represents the percentage tolerance above and below the mass value in the mass-dimension relationship for which particles will be included in the size distribution and statistical output.

2.5.2 Normalized Size Distribution

Please see **subsection 2.4.2** in "**Modeled PSD**" section above.

2.5.3 Frequency

Please see previous Frequency subsections.

2.5.4 Submit Inputs

Similar to the "**Modeled PSD**" interface, once all of the required inputs have been specified, the "Submit Inputs" button becomes functional, otherwise it remains inactive. Clicking the "Submit Inputs" button triggers the construction of two graphs with each on a tab, "Particle Size Distribution" and "Mass-Dimension Relation", in the Active Selection Area. The "Particle Size Distribution" graph is similar to that in the "**Modeled PSD**" interface with the same controls.

The "Mass-Dimension Relation" graph plots mean mass (y-axis) against the mean maximum dimension (x-axis) for each bin using hollow-diamond symbols (see **Fig. XXX**). The corresponding mean mass and mean maximum dimension values will be shown, when the diamond symbol is moused over. Clicking on a diamond displays an inset graph plotting individual mass and maximum dimension values in the corresponding bin.

The four (4) buttons below the graph, i.e. "Back to Inputs", "Export to PNG", "Export Data", and "Email Data", work the same way as in the "**Modeled PSD**" interface (see **subsection 2.4.4** above).

2.5.5 *m-D* & PSD Interface Tutorial

Please be reminded that a registered email address is needed to access the **OpenSSP Query Builder** browser interface. See **Section 2.1** for the registration process. Once a registered email address is entered, a (default but customizable) unique Session ID is assigned and the interface opens with the **Pristine** tab being active.
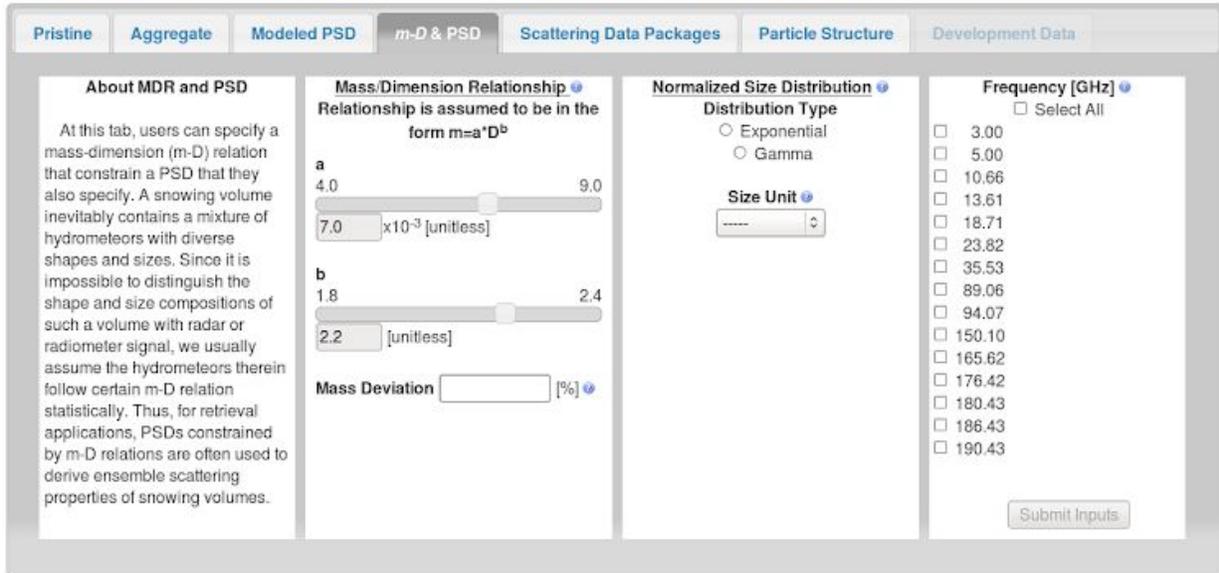
Once the interface appears, click the **m-D & PSD** to select it. There are four (4) columns in the Active Selection Area (see **Fig. XXX**) with the first (leftmost) The desired mass-dimension relation can be specified in the second (2nd) column and the PSD in the third (3rd). Similar to previous interfaces, frequency selection is made in the last (rightmost) column.
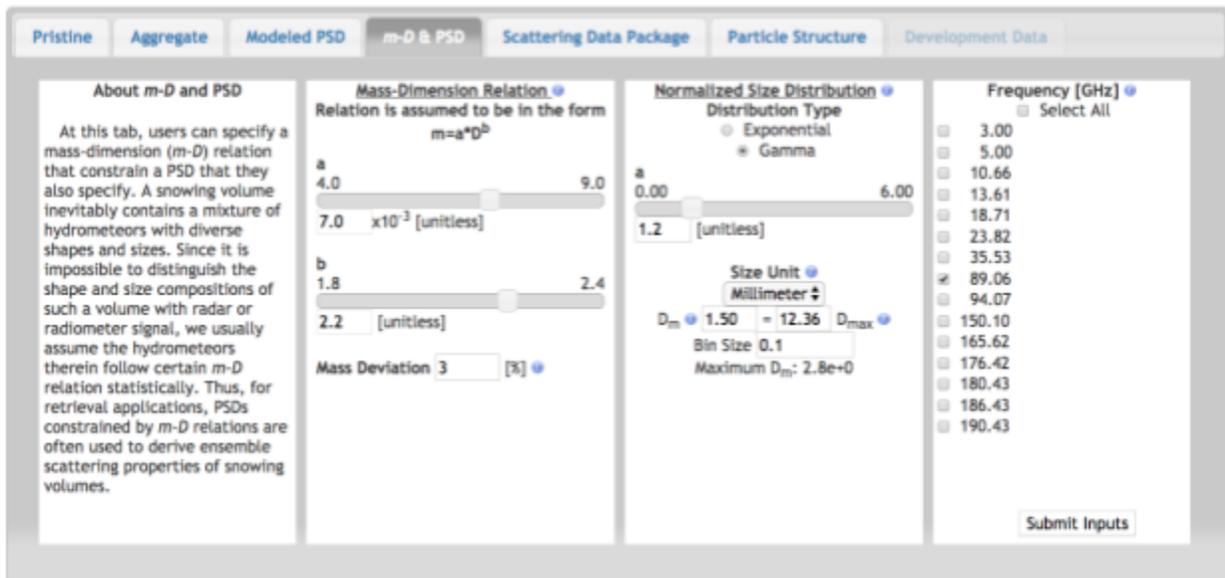
The first thing to do in this tab is to define your mass/dimension relationship using either the sliders or the input fields for variables *a* and *b*. Note that the slider and input field for *a* are scaled by 10^{-3}. In this tutorial, *a* and *b* are left unchanged at their defaults. The *m-D* relation tolerance is specified as 3% of the corresponding mass. The remaining inputs are identical to those found in the **Modeled PSD** interface. A fully filled-out interface looks like **Fig. XXX**..
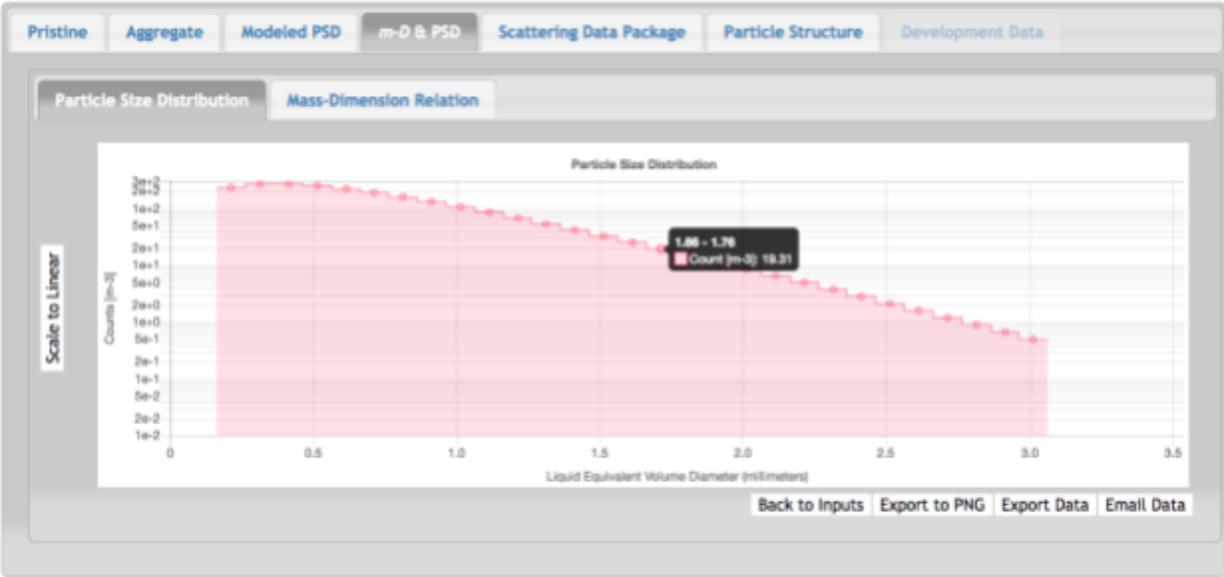
API Query Call: OpenSSPAPI?email=███████████████&size=all/all&dsubm=1.5&bs=0.1&bunit=millimeters&mu=1.2&frequency=089.062GHz&aval=0.007&bval=2.2&diff=3

Upon clicking on Submit Inputs, the site switches to showing a particle size distribution graph in a tab with the same name (see **Fig. XXX**). Similar to the PSD plot in the **Modeled PSD** interface, the y-axis scale can be toggled between logarithmic and linear using the button to the left and mousing over the circle atop the histogram bar displays the exact count values. The plot can also be exported as a PNG image.
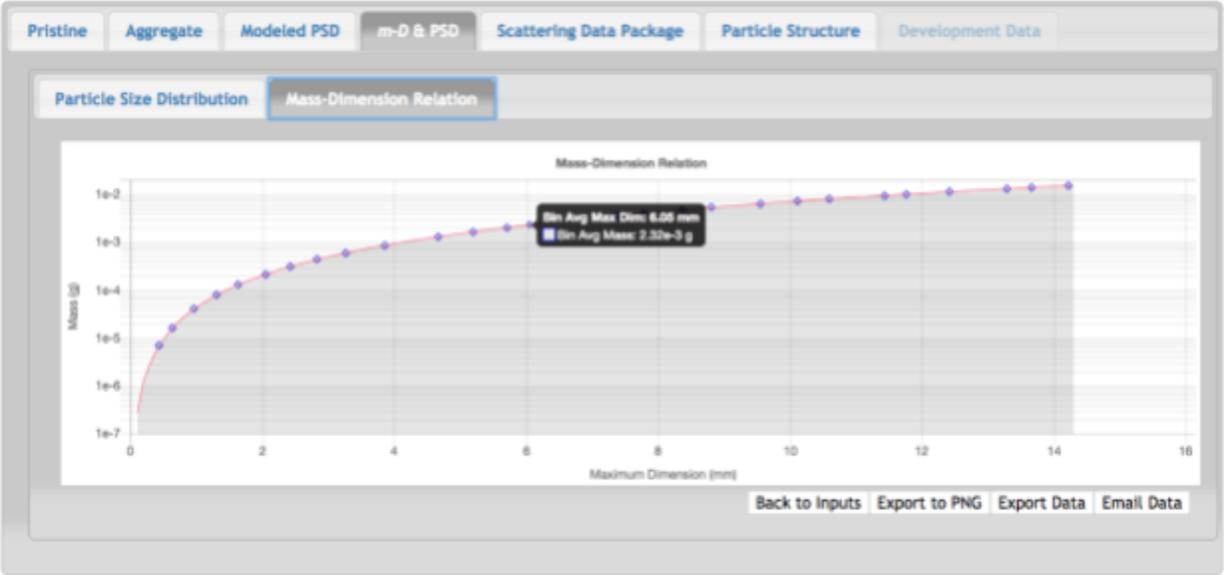
API Query Call: OpenSSPAPI?email=▉▉▉▉▉▉▉▉▉▉&size=all/all&dsubm=1.5&bs=0.1&bunit=millimeters&mu=1.2&frequency=089.062GHz&aval=0.007&bval=2.2&diff=3

In addition to the PSD histogram, the Mass-Dimension Relation tab displays the mass-dimension relation, which plots the *m-D* relation based on the *a* and *b* inputs with the bin-averages from the qualifying particles overlaid as hollow diamonds, supporting the mouse-over feature (see **Fig. XXX**). Clicking on a diamond symbol brings up an inset on the right side (see **Fig. XXX**), which shows individual particles in the corresponding bin in relation to the *m-D* relation. This chart can also be exported as PNG.

The same mouse-over feature also works in the inset (see **Fig. XXX**).

"**Email Data**" is especially recommended when using *m-D* **& PSD** interface to submit queries, as the need to search the entire OpenSSP synthetic hydrometer collection for those satisfying the *m-D* relation constraint inevitably leads to longer processing time. The query results are similarly organized as those of **Modeled PSD**, but each size bin contains only those particles within the tolerance of the *m-D* relation. **Fig. XXX** shows an example when the results have been ported into Microsoft Excel.

2.6 Scattering Data Packages

The **Scattering Data Package** interface is designed for users who wish to acquire the full OpenSSP database in HDF5 format, or its frequency specific subset(s). Desired frequency subset(s) are selected using the checkbox next to the frequency in GHz. Clicking the "Order Archives" button starts the download.

It is important to note that:

1. There is **no** corresponding web service support for this interface, so the API Query Call on this interface is irrelevant.
2. These archives do not contain particle structure data of the synthetic snowflakes, which can be obtained with the interface of the next tab described in the next section.

2.7 Particle Structure
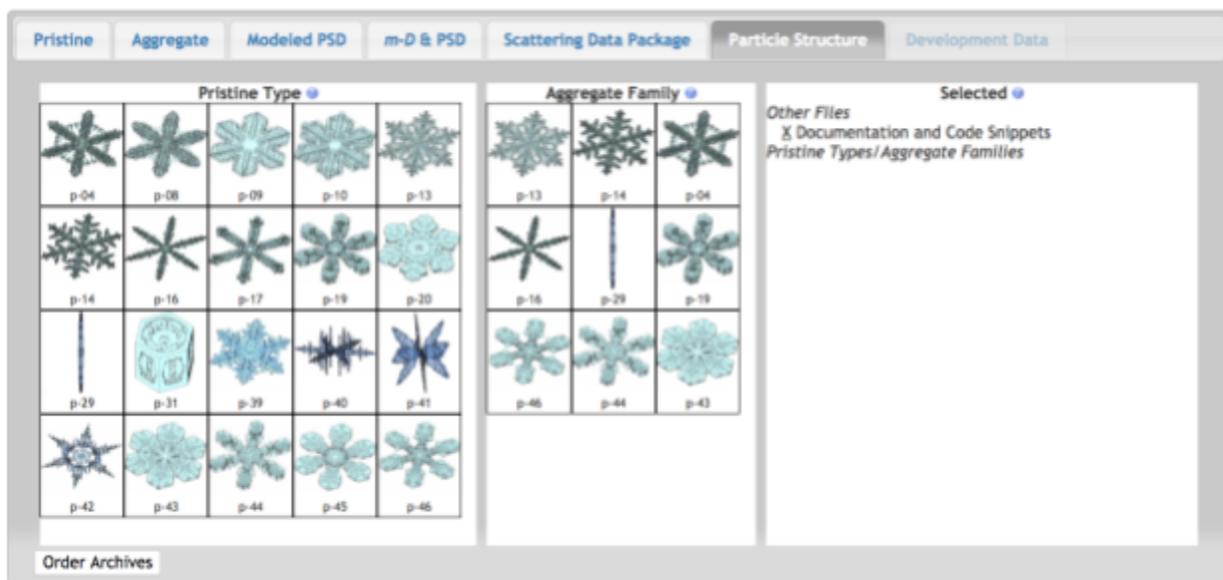
The **Particle Structure** interface provides the ability to download gzipped tar archives of particle structure files grouped by Pristine Type and Aggregate Family. Multiple thumbnail images may be selected for download at the same time. Clicking on the thumbnail images toggles the status between selected and deselected as before with the other interfaces. Selected collections can also be deselected by clicking the underlined "X" next to the collection name in the Selected column. These downloaded files contain the hydrometeor particle structures in its original *Snowfake* resolution, before they were subjected to the manipulation (described in sections 2.*b* and 3.*a* of Kuo et all 2016) to arrive at the inter-dipole distance of 50 μm.

By default, the download will include a text file (PartStructDoc.txt) that describes the format of the files contained in the archive, which can be deselected if not needed. In addition, it includes

code snippets in Python, IDL, and MatLab to demonstrate how to read the files in those languages.

Help Text

Selected Particles: This list comprises Pristine and Aggregate Particles currently selected to be included in the .tgz archives that will be downloaded. Clicking the "X" button to their left will remove them from the selection list.

3 The Web Service Application Programming Interface

3.1 DESCRIPTION

The OpenSSP web service application programming interface (henceforth OpenSSP API) call expects a series of keyword arguments and values to query the OpenSSP database and returns query results in either tab or comma-separated format. It requires an email registration as described in section **2.1**. It is designed to provide a means for more automated access to data within the OpenSSP HDF5 archive in a user-friendly and flexible manner. An example of using the API directly without the Query Builder interface is given in Example 5.

3.2 PARAMETERS AND VALUES

3.2.1 Email

The first required parameter is the registered email address of the user performing the query. The query will fail if the email is not registered.

3.2.2 Size

The size field contains combinations of the shape name and the volume equivalent radius in micrometers of the various stages of growth for a given experimental synthetic snowflake. These sizes are tied directly to the shape selected, but because some sizes overlap between multiple snowflakes, it is a requirement that both the shape and size be listed together in the following format: "shape/size". Shape names begin with a lowercase letter, "p" for Pristine, "a" for Aggregate. Sizes must have six digits before the decimal (including leading zeros), and six digits after the decimal, followed by "um". Multiple shape/size combinations can be included in a given query. A maximum of 20 discrete shape/size combinations are allowable in a given query.

A variant of the size parameter is if the user is interested in all of the sizes for a given shape. In that case, the format would be the shape name, the forward slash, and then the word "all". These "all" statements can be chained together to include multiple shapes and sizes.

Examples for "size" keyword:

- `size=p-09/000139.002106um,p-10/000184.949203um`
- `size=a-0094/000099.489769um`
- `size=p-17/all,p-14/al`

3.2.3 Frequency

The frequency field includes one or more frequencies of radiation used to determine the scattering properties of the synthetic snowflake of the shape/size selected. The frequency value

must have three digits before the decimal (including leading zeros), and three digits after, followed by GHz.

Like with the size parameter, there is a shorthand for including all frequencies. In this case, the user can simply place the word "all" as the frequency parameter.

Examples for "frequency" keyword:

- `frequency=023.816GHz`
- `frequency=013.609GHz,094.065GHz,035.525GHz`
- `frequency=all`

3.2.4 Delimiter

The delimiter can be "comma", in which case the response will have commas separating the fields in the response. Anything else (or no value) will return a response with tabs separating the fields.

3.2.5 Text

By default, the query response comes in html format and includes a button for allowing downloading the data in tsv or csv format (depending on the delimiter). Adding the parameter "`text=true`" will give the response in raw text form. This is ideal for automated requesting once you have already figured out the queries in which you are interested.

3.2.6 PSD Parameters

For specifying a PSD, one needs to provide the following parameters dsubm (i.e. D_m

There are seven parameters involved in generating a response based on a particle size distribution (as well as size). Six of these are numerical and are in units of meter: gamma (only required to define the shape of a gamma distribution, exclude for an exponential distribution), nnstar (the parameter for N_0^*), dsubm (the parameter for D_m), bs (the parameter for bin size, which is a constant value), min (the minimum diameter in the first bin), and max (the maximum diameter in the last bin). The final parameter is "bunit", which is one of three values: microns, millimeters, meters. These define the organization and presentation of the output file. Output values are arithmetically averaged within each bin. Empty bins contain "-9999" values.

Example for PSD related keywords:

- `dsubm=1.2&bs=0.15&bunit=millimeters&mu=2`

3.2.7 *m-D* Relation Parameters

There are three additional parameters required when generating a particle size distribution based on a mass-dimension relationship. All three are numerical and unitless: a (coefficient for

mass-dimension relationship), b (exponent for mass-dimension relationship), and diff (percent difference for particles in particle size distribution based on mass-dimension relationship). Output values are arithmetically averaged within each bin based on particle size distribution. Empty bins contain "-9999" values.

Example for *m-D* relation kewords:
```
"a=0.007&b=2.2&diff=10"
```

3.3 RESPONSE

The API response, as stated previously, is a delimiter (tab by default) separated text file ideal for import into spreadsheet software or to be read programmatically. It always starts with a header line with the list of variable names described in Appendix A. Each subsequent line represents a combination of the shapes, sizes, and frequencies selected by the user. The file ends with a series of commented lines featuring descriptions of the variable names. API responses have the same set of variables except for one: bin count. This variable only appears in Particle Size Distribution API requests and gives the number of particles in each bin based on the distribution parameters defined.

3.4 Examples: Using the API Call in Python

3.4.1 Example 1: Using numpy

While the Query Builder is designed for exploring OpenSSP data easily in different ways, the OpenSSP API Query can be called programmatically within your favorite language. One key thing to remember when calling the API from within your code is to set the parameter "text=true" to ensure that you are only receiving the raw csv/tsv file back and not the HTML wrapper that is used to display the response in a user's browser.

```
1    import urllib
2    import numpy as np
3    import matplotlib as m
4    m.use('Agg') #Because I am doing this on a server with no video drivers
5    import matplotlib.pyplot as plt
6
7    urlObject =
     urllib.urlopen('https://storm.pps.eosdis.nasa.gov/storm/OpenSSPAPI?email=ma
     tthew.r.lammers@nasa.gov&frequency=089.062GHz&size=p-40/all&text=true')
8    """Looking at the relationship between maximum dimension and
     scattering/absorption efficiency."""
9    """Looking at relationship of scattering and absorption efficiency to
     projected area"""
10   plt.plot(osspArray['Prj_Area[um^2]'],osspArray['Q_abs'],'ro',
     label='Absorption Efficiency')
```

```
11   osspArray = np.loadtxt(urlObject,comments='#',unpack=True,skiprows=2,
     usecols(4,8,9))
12   print osspArray
13   plt.plot(osspArray[0],osspArray[1],'ro',label='Scattering Efficiency')
14   plt.plot(osspArray[0],osspArray[2],'bo',label='Extinction Efficiency')
15   plt.legend()
16   plt.xlabel('Projected Area [um^2]')
17   plt.ylabel('Efficiency')
18   plt.savefig('AbsVsScat.png')
19   plt.close()
```

All the examples in this section (i.e. section 3.4) need to import the modules demonstrated in lines 1, 3, 4, 5. The code above

1. opens the output, i.e. results of the API request, as a file object (urlObject) (line 7),
2. loads the text of the file object into a numpy array (osspArray) containing the values from columns 4, 8, and 9, which are maximum dimension, scattering efficiency, and extinction efficiency respectively (line 11), and
3. plots the values on a basic matplotlib plot.

The plot uses maximum dimension for the x-axis and scattering/extinction efficiency for the y-axis. This is a fairly basic thing to do with the data, but it shows how just a few lines of code can generate a nice looking plot from the OpenSSP data. Below is the output plot from this code.
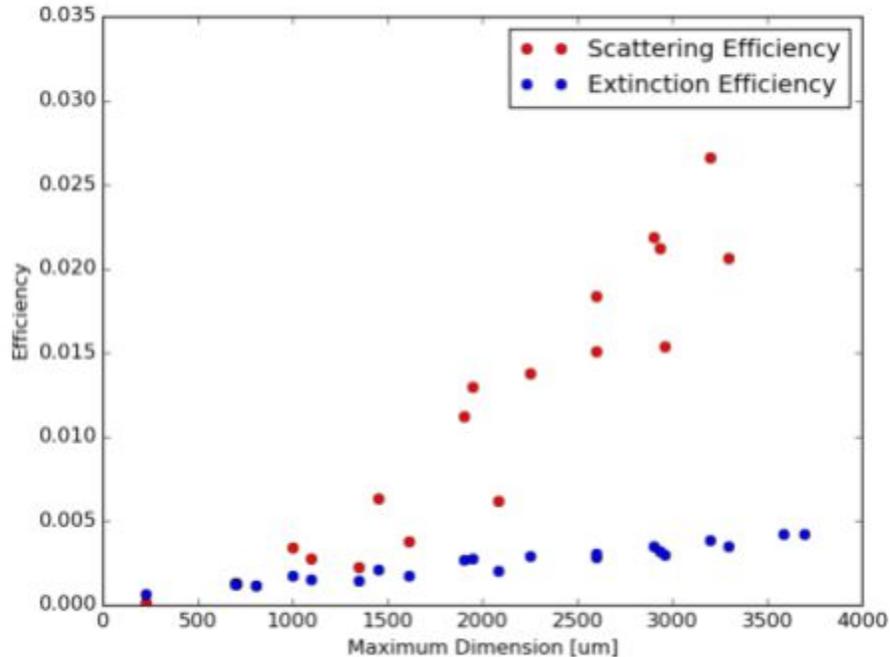
3.4.2 Example 2: Using pandas

While the first example above uses `numpy` to handle the tsv parsing, a more powerful package, called `pandas`, is available in Python for performing statistical analysis. It natively uses a storage construct called a "Data Frame," which enables indexing by string-based keys like the ones provided in the header line of the OpenSSP API return. Below is an example of how to use pandas to generate a similar plot using output from Tab 2, "Aggregate."

```
1    import urllib
2    import pandas
3    import matplotlib as m
4    m.use('Agg') #Because I am doing this on a server with no video drivers
5    import matplotlib.pyplot as plt
6
7    urlObject =
     urllib.urlopen('https://storm.pps.eosdis.nasa.gov/storm/OpenSSPAPI?email=ma
     tthew.r.lammers@nasa.gov&frequency=023.816GHz&size=a-0000/all,a-0001/all,a-
     0002/all,a-0003/all,a-0004/all&text=true')
8    #We read the tsv into a python pandas data frame
9    osspArray = pandas.read csv(urlObject,sep='\t',header=0,comment='#')
10   urlObject.close()
11   #Here we remove subheader rows that only contain frequency
12   osspArray = osspArray[~osspArray['Wvlngth[um]'].isna()]
```

```
13   print osspArray #Notice the difference between this output and numpy
14
15   """Looking at relationship of scattering and absorption efficiency to
     projected area"""
16   plt.plot(osspArray['r_eq_vol[um]'],osspArray['Q_ext'],'ro',
     label=Extinction Efficiency')
17   plt.plot(osspArray['r_eq_vol[um]'],osspArray['Q_sca'],'bo',
     label='Scattering Efficiency')
```

Here we notice a few differences between pandas and numpy.

1. Firstly, we see a dedicated "read_csv" function with a few different options on line 9.
2. Next, we have a straightforward way to remove the extra subheader lines that only contain frequencies because when those rows get pulled in, they have "NaN" values for every other column on line 12. We can easily reassign the data frame to exclude those rows with "NaN" values for frequency.
3. Finally, when we plot the variables, we can now select columns by those header names rather than by index values, making the code easier to read and understand (see lines 16-18).

3.4.3 Example 3: Binned PSD Data

Similar processing can be carried out with binned PSD data. In this example, we have chosen a gamma distribution with D_m = 1.8, bin size = 0.1, bin unit = millimeters, and μ = 2.5. Included are pristine particles from p-13 and p-39 and aggregate particles from p-19. We use pyplot from matplotlib as before to visualize the same variables from the API: extinction efficiency, backscatter efficiency, and scattering efficiency. Below is the code to handle the binned PSD output.

```
1    import urllib
2    import pandas
3    import matplotlib as m
4    m.use('Agg') #Because I am doing this on a server with no video drivers
5    import matplotlib.pyplot as plt
6
7    urlObject =
     urllib.urlopen('https://storm.pps.eosdis.nasa.gov/storm/OpenSSPAPI?email=ma
     tthew.r.lammers@nasa.gov&size=all/all&dsubm=2.3&bs=0.1&bunit=millimeters&av
     al=0.0057&bval=2&diff=20&frequency=035.525GHz&text=true')
8    #We read the tsv into a python pandas data frame
9    osspArray = pandas.read csv(urlObject,sep='\t',header=0,comment='#')
10   urlObject.close()
11   #No subheader rows currently in PSD returns, so no need to remove them
```

```
12    print urlObject
13
14    """Looking at relationship of scattering and absorption efficiency to
      projected area"""
15    plt.plot(osspArray['r_eq_vol[um]'],osspArray['Q_ext'],'ro',
      label=Extinction Efficiency')
16    plt.plot(osspArray['r_eq_vol[um]'],osspArray['Q_sca'],'bo',
      label='Scattering Efficiency')
```

3.4.4 Example 4: m-D & PSD

Our final example utilizes the "***m-D & PSD***" interface to validate an important aspect of the way particle size distributions are constructed in the OpenSSP API, i.e. the normalization to 1 g/m^3 Liquid Water Content. To do this, we first conduct a query with all of the relevant values (*a, b*, mass difference, bin size, and D$_m$ - line 7). We then compute the average mass for all of the particles within each bin using the average liquid equivalent volume radius (and assuming a density of 1 g/cm^3, or 1.0x10^6 g/m^3 - lines 14-16). Next, we multiply that by the bin count, and sum to get the total mass, which should be 1 gram (lines 17-20). Below is the code to do this and generate a cumulative distribution function based on the mass contribution from each bin.

```
1     import urllib
2     import pandas
3     import matplotlib as m
4     m.use('Agg') #Because I am doing this on a server with no video drivers
5     import matplotlib.pyplot as plt
6
7     urlObject =
      urllib.urlopen('https://storm.pps.eosdis.nasa.gov/storm/OpenSSPAPI?email=ma
      tthew.r.lammers@nasa.gov&size=all/all&dsubm=2.3&bs=0.1&bunit=millimeters&av
      al=0.0057&bval=2&diff=20&frequency=035.525GHz&text=true')
8     #We read the tsv into a python pandas data frame
9     osspArray = pandas.read_csv(urlObject,sep='\t',header=0,comment='#')
10    urlObject.close()
11    #No subheader rows currently in PSD returns, so no need to remove them
12    print urlObject
13    #Here we will use the data to show the bin counts are normalized
14    liquid_volumes = 4.0/3.0*np.pi*(osspArray['r_eq_vol[um]']/1.0e6)**3
15    #We have to divide by 1.0e6 to get the radius into meters
16    liquid_masses = liquid_volumes*1e6 #Multiply by 1e6 g/m^3 density
17    counted_mass = osspArray['bin count']*liquid_masses
18    print counted_mass
19    totmass = np.sum(counted_mass)
20    print totmass
```

```
21
22   #Calculate the cumulative sum
23   mass cdf = np.cumsum(counted mass)
24   plt.plot(osspArray['r eq vol[um]'],mass cdf,'r-')
25   plt.xlabel('Bin-Averaged Volume Equivalent Radius [um]')
26   plt.ylabel('Mass [g]')
27   plt.savefig('MassCDF.png')
28   plt.close()
```