Parametric and Optimization Workflows using LUME-ACE3P

ACE3P Webinar

David Bizzozero

November 19, 2025

SLAC NATIONAL ACCELERATOR LABORATORY

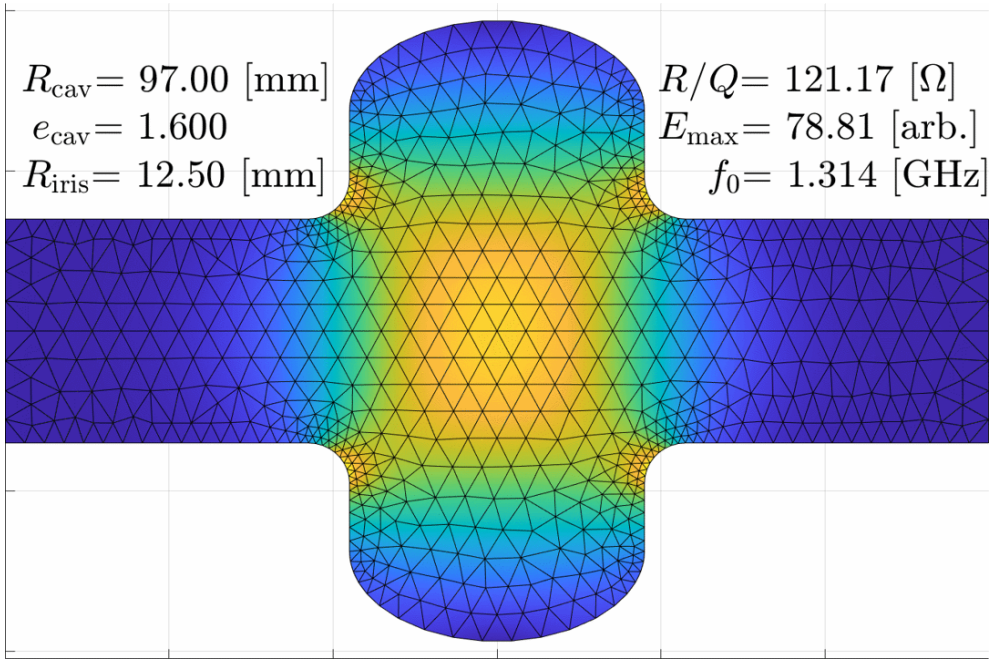
Stanford University

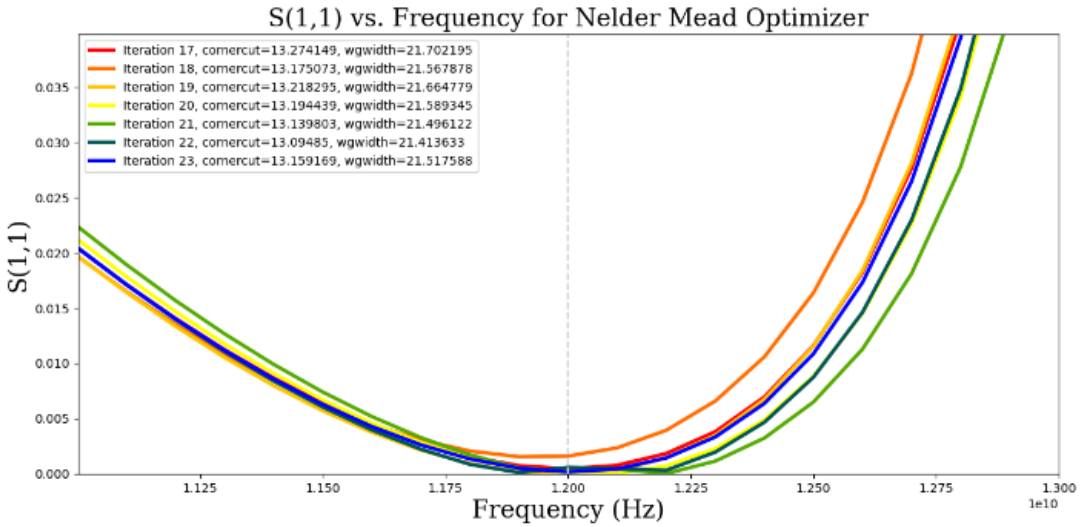
U.S. DEPARTMENT of ENERGY

Introduction to LUME-ACE3P

LUME-ACE3P – set of Python code interfaces and workflows involving Cubit, ACE3P, and postprocessing tasks for parametric sweeping or optimization (built upon LUME† framework)

Parameter sweeps with Omega3P*





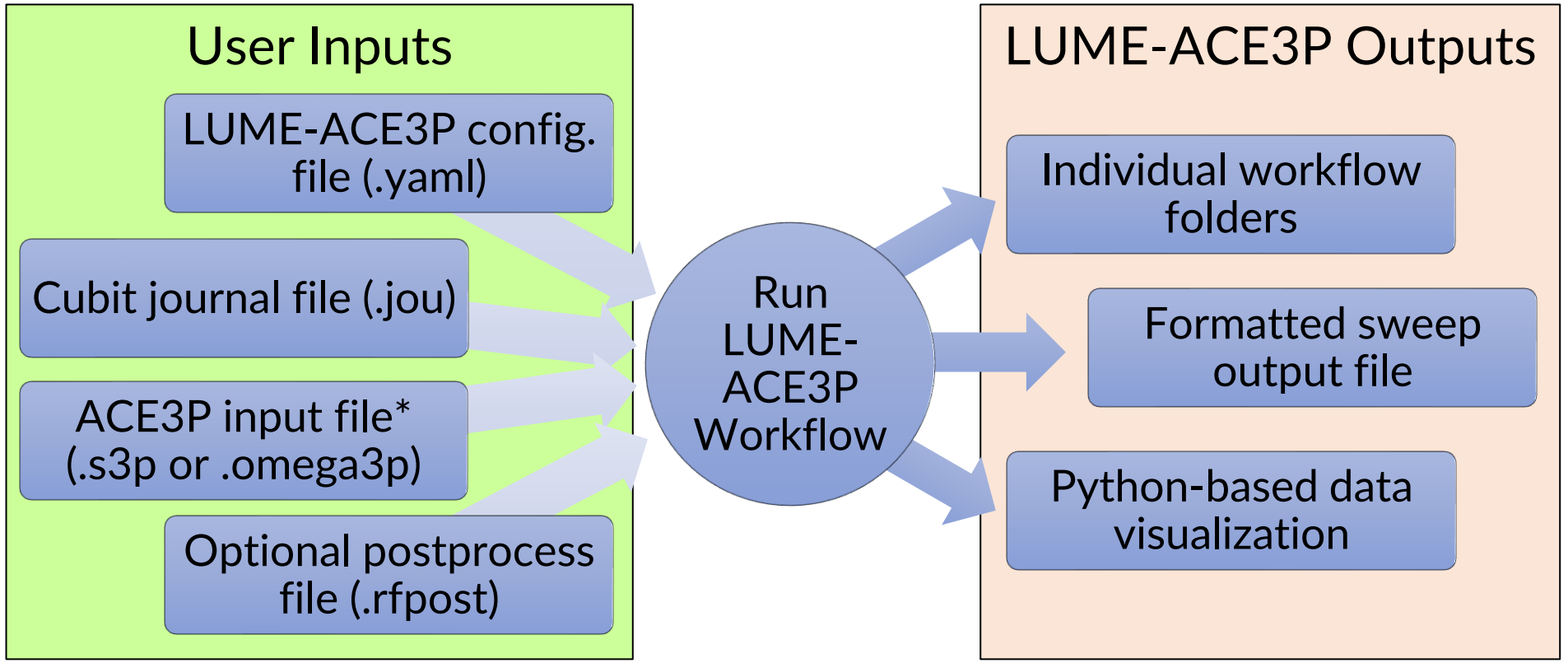
Reflection minimization with S3P

†: Lightsource Unified Modeling Environment: https://github.com/slaclab/lume
*: Postprocess plot animation code not included

Webpage: https://github.com/slaclab/lume-ace3p

Overview of LUME-ACE3P Files

LUME-ACE3P requires a configuration file and input files for Cubit and ACE3P*



*: ACE3P input file parameters can be included directly into the LUME-ACE3P configuration file instead with YAML formatting

LUME-ACE3P Setup on Perlmutter

To set up LUME-ACE3P on Perlmutter:

1. Source ACE3P libraries with setup script (can be placed in .bashrc file)
   - `source $CFS/ace3p/perlmutter/CPU/perlmutter-ace3p-spack.sh`
2. Load a Conda manager on login node such as the default one (also can be in .bashrc)
   - `module load conda`
3. Load the LUME-ACE3P Conda environment (also can be in .bashrc or batch script)
   - `conda activate $CFS/ace3p/software/lume-ace3p`
4. Copy examples folder to scratch (recommended), home, or another directory
   - `cp -r $CFS/ace3p/lume-ace3p/examples .`

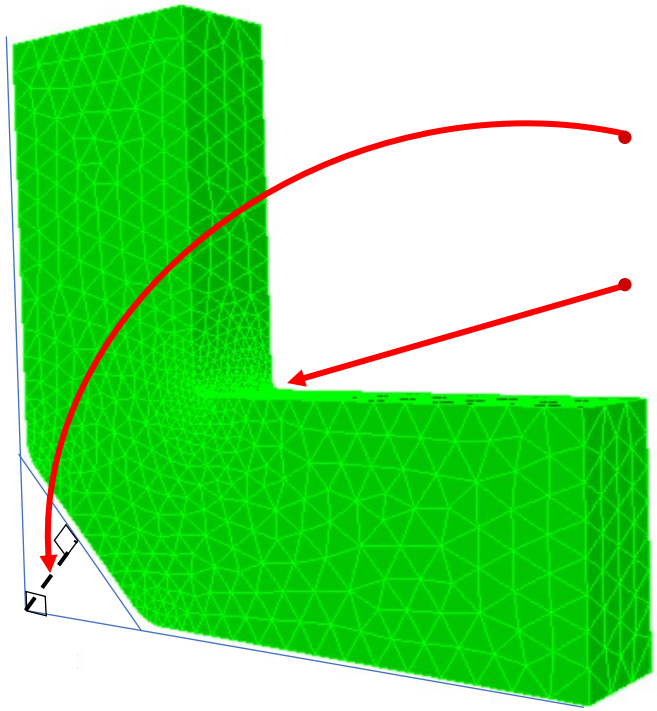
1

S3P Parameter Sweep Example

Example 1: S3P Parameter Sweep

S-parameter scan for 90° bend with 2 shape parameters

- Two-port network setup (2x2 S-matrix)
- Frequency scan range 9.424~12.424 GHz



Parameter 1: corner chamfer length

- 'cornercut' = 12~16 mm

Parameter 2: inner corner radius

- 'rcorner2' = 4~16 mm

See S3P tutorial for additional information at
https://confluence.slac.stanford.edu/display/AdvComp/Materials+for+CW23

Example 1: S3P Parameter Sweep: Cubit File Setup

Cubit journal file contents

bend-90degree.jou

```
reset

#{wgwidth = 22.86}
#{wgheight = 10.16}
#{wglength = 60.0}
#{cornercut = 16}
#{rcorner1 = 1}
#{rcorner2 = 5}

…

Sideset 7 surface 1
Sideset 8 surface 3
Sideset 6 surface all except 1 3

block 1 volume all
block 1 element type tetra10
volume all scale 0.001
export Genesis "bend-90degree.gen" block all overwrite

quit
```

Parameters to sweep (numbers after "=" sign will be overwritten)

Geometry and design instructions go here (not edited by LUME-ACE3P)

Sideset numbers must *exactly* match with ACE3P input file references

Export filename must *exactly* match import filename in ACE3P (.gen format automatically converted to .ncdf)

Example 1: S3P Parameter Sweep: S3P File Setup

S3P input file contents

bend-90degree.s3p

```
ModelInfo: {
  File: ./bend-90degree.ncdf

  BoundaryCondition: {
    Exterior: 6
    Waveguide: 7, 8
  }
}

…

FrequencyScan: {
  Start: 9.424e+9
  End:   12.424e+9
  Interval: 0.25e+9
}

Port: {
  ReferenceNumber: 7
  NumberOfModes: 1
}
…
```

Filename must *exactly* match export from Cubit with .ncdf suffix (automatic conversion)

Sideset numbers must *exactly* match with Cubit journal file references

Other S3P input file containers

Frequency scan range is set here in ACE3P input file *not* in the configuration YAML file!

Port mode S-parameters will be calculated and stored in sweep output file

Example 1: S3P Parameter Sweep: Config. File Setup

LUME-ACE3P configuration file contents

demo_s3p_sweep.yaml

```
workflow_parameters :
  'mode' : 'parameter_sweep'
  'module' : 's3p'
  'cubit_input' : 'bend-90degree.jou'
  'ace3p_input' : 'bend-90degree.s3p'
  'ace3p_tasks' : 32
  'ace3p_cores' : 4
  'ace3p_opts' : '--cpu-bind=cores'
  'workdir' : 'lume-ace3p_s3p_workdir'
  'workdir_mode' : 'auto'
  'sweep_output' : True
  'sweep_output_file' : 's3p_sweep_output.txt'

input_parameters :
  'cornercut' :
    'min' : 12.0
    'max' : 16.0
    'num' : 5
  'rcorner2' :
    'min' : 4.0
    'max' : 16.0
    'num' : 3
```

Parameter sweep mode for s3p run

Input filenames for cubit and ace3p

HPC options for ACE3P part of workflow

Working directory settings (auto-append parameter values to workdir)

Sweep output toggle and filename

Input names must *exactly* match those in Cubit journal

Input values to sweep from ‘min’ to ‘max’ in ‘num’ steps (in this example: total of 5x3 workflow evaluations)

Example 1: S3P Parameter Sweep: Batch Script Setup

S3P sweep batch script contents

run_lume-ace3p_s3p_sweep_perlmutter.batch

```
#!/bin/bash

#SBATCH --constraint=cpu
#SBATCH --account=m349
#SBATCH --qos=debug
#SBATCH --job-name=lume-ace3p_psweep_demo
#SBATCH --output=output-%j.txt
#SBATCH --error=error-%j.txt
#SBATCH --nodes=1
#SBATCH --time=0-00:30:00

export PYTHONPATH='/global/cfs/cdirs/ace3p/lume-ace3p'

python /global/cfs/cdirs/ace3p/lume-ace3p/lume_ace3p/run_lume_ace3p.py demo_s3p_sweep.yaml
```

Typical HPC settings

Python path set to LUME-ACE3P location

Python code to initiate and run LUME-ACE3P

User-defined configuration script

Example 1: S3P Parameter Sweep: Outputs

Workflow folders will automatically be created

- Naming structure given in configuration file
  - Parameter values appended automatically to name (with underscores)
- All possible combinations of inputs will be evaluated
  - Note: consider allocating enough time and resources to run all workflows!

Sweep output text file contains *all* S-parameter data for all swept parameters

- Columns: parameters, frequency, S-matrix entries
  - Each “Reflection.out” data file (from S3P) combined into 1 large file
- Rows: one parameter combination and one frequency
- Tab delimited format (useful to import into spreadsheet)
- Can be parsed by LUME-ACE3P python plotting tool “s3p_sweep_plot.py”

Example 1: S3P Parameter Sweep: Output File

S-parameter outputs parsed from “Reflection.out” files

s3p_sweep_output.txt

cornercut	rcorner2	Frequency	S(0,0)	S(0,1)	...
12.0	4.0	9.4240E09	0.157987	0.000184	
12.0	4.0	9.6740E09	0.147880	0.000223	
12.0	4.0	9.9240E09	0.140172	0.000275	
...	...	...	...	...	
12.0	4.0	1.2174E10	0.381495	0.017433	
12.0	4.0	1.2424E10	0.639386	0.047314	
12.2	4.0	9.4240E09	0.149214	0.000187	
12.2	4.0	9.6740E09	0.138512	0.000225	
...	...	...	...	...	
16.0	16.0	1.2174E10	0.099065	0.005209	
16.0	16.0	1.2424E10	0.105860	0.008727	

Reflection.out file contents for one set of input parameters

(original file in “_12.0_4.0” suffixed folder)

All combinations of input parameters and frequencies

Input parameters

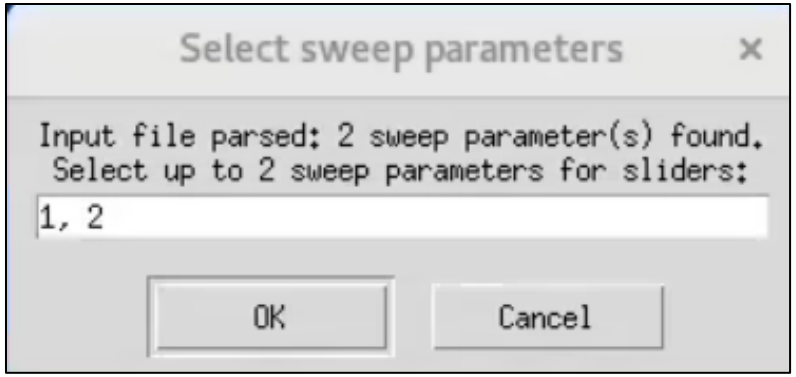
S-parameter outputs

Number of rows =
(# values of parameter combinations)
x (# of frequencies)

Example 1: S3P Parameter Sweep: Sweep Plotting Tool

To run s3p_sweep_plot.py on Perlmutter:

1. Load/refresh the LUME-ACE3P Conda environment
   - `conda activate $CFS/ace3p/software/lume-ace3p`
2. Copy plotting folder to scratch, home, or desired directory
   - `cp -r $CFS/ace3p/lume-ace3p/plotting .`
3. Run plotting script via command line with input file argument
   - `python s3p_sweep_plot.py s3p_sweep_demo_output.txt`
   - If no input file provided, a pop-up will prompt for one
4. Select up to 2 parameter indices for plot sliders
   - These are the column indices from the sweep output file
   - Frequency is *not* a parameter (it is used as the x-axis)

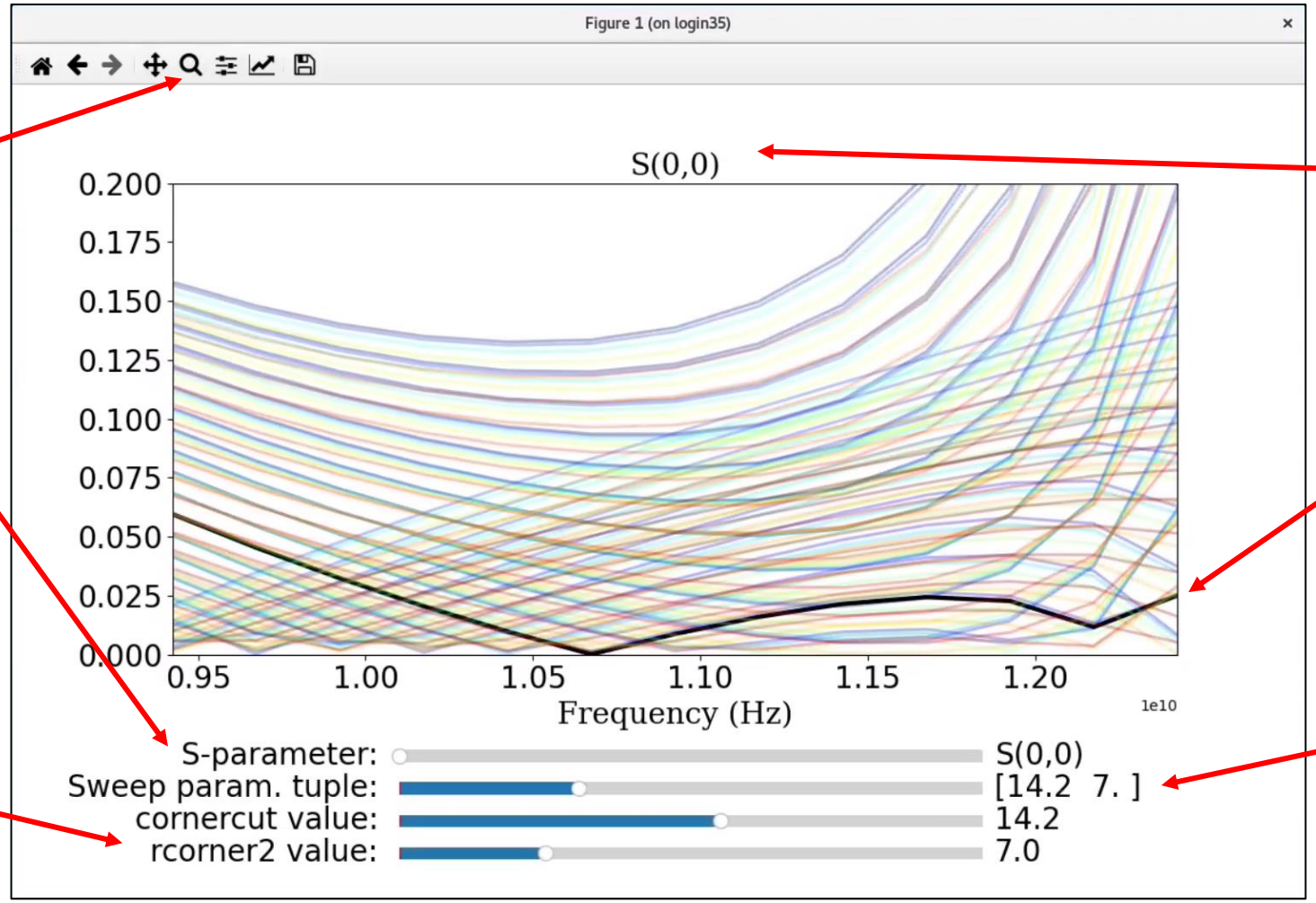


Example 1: S3P Parameter Sweep: Sweep Plotting Tool

Plotting window options and zoom settings

S-parameter slider (numbered by ports and modes in sweep output file)

Individual parameter sliders can be adjusted



Currently selected S-parameter (column name in sweep output file)

Currently selected data curve highlighted in black

S-parameter tuple lists the parameter value (folder name) combination

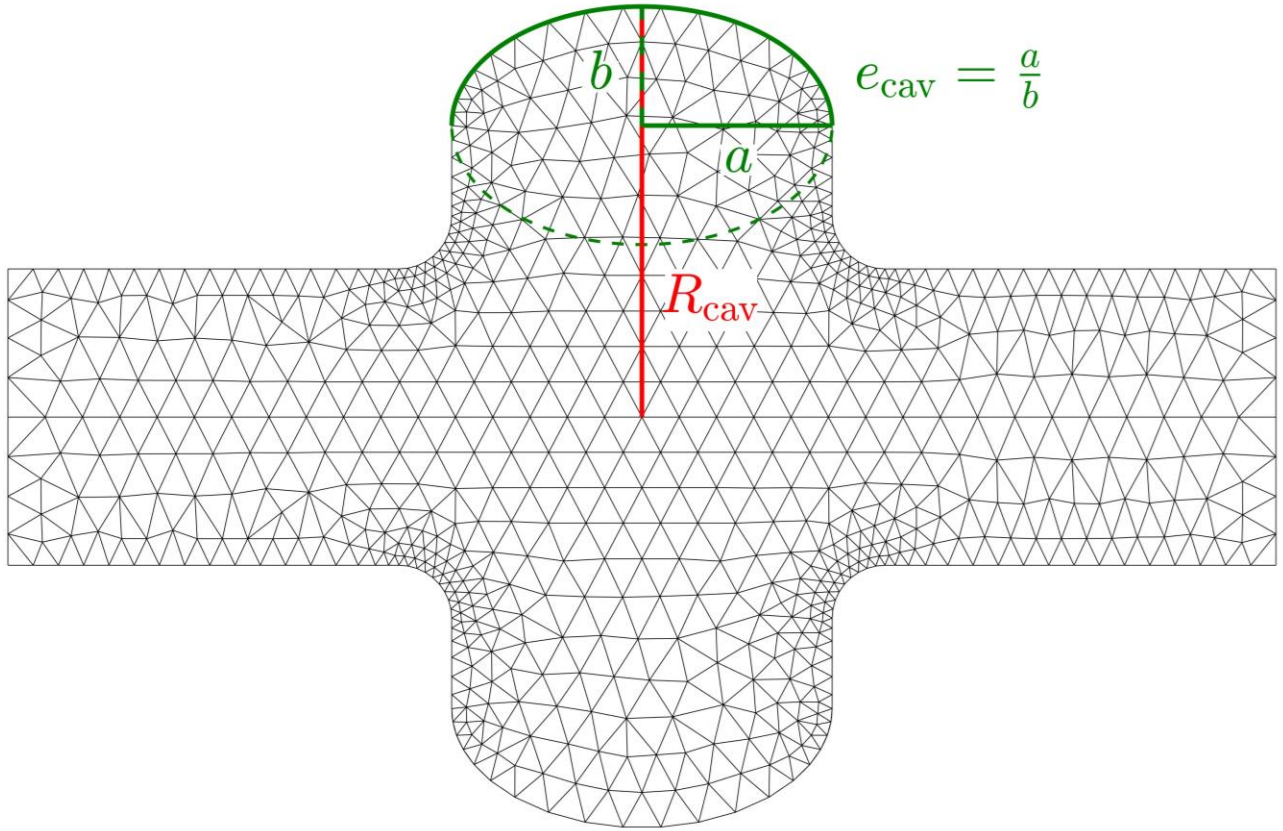
2

Omega3P Parameter Sweep Example

Example 2: Omega3P Parameter Sweep

Rounded pillbox eigenmode and R/Q calculation with 2 shape parameters

- Parameter 1: cavity outer wall radius
  - ‘cavity_radius’ = 90~120 mm
- Parameter 2: cavity ellipticity ratio
  - ‘ellipticity’ = 0.5~1.25



- Outputs from acdtool postprocess:
  - R/Q, Emax, Emax location (x,y,z)

See Omega3P tutorial for additional information at
https://confluence.slac.stanford.edu/display/AdvComp/Materials+for+CW23

Example 2: Omega3P Parameter Sweep: Input File Setup

Cubit + ACE3P input files set up similarly as in S3P example:

- Make sure variable names match exactly
- Make sure export mesh file names are consistent
- Make sure sideset numbers are consistent

Acdtool postprocess input file contents

pillbox-rtop.rfpost

```
RFField
{
   …
}
RoverQ
{
   ionoff       =       1
   modeID1      =      -1
   modeID2      =      -1
   x1           =       0.00000
   x2           =       0.00000
   y1           =       0.00100
   y2           =       0.00100
   z1           = 100000000.00000
   z2           = 100000000.00000
}
maxFieldsOnSurface
{
   ionoff       =       1
   surfaceID    =       6
}
```

Standard RFField container inputs

RoverQ section must be toggled on to use output parsing!

maxFieldsOnSurface section must be toggled on to use output parsing!

surfaceID must be set and match sideset number for output parsing

Example 2: Omega3P Parameter Sweep: Config. File Setup

Omega3P sweep config. file contents

demo_omega3p_sweep.yaml

```
workflow_parameters :
  'mode' : 'parameter_sweep'
  'module' : 'omega3p'
  'cubit_input' : 'pillbox-rtop.jou'
  'ace3p_input' : 'pillbox-rtop.omega3p'
  'rfpost_input' : 'pillbox-rtop.rfpost'
  'ace3p_tasks' : 16
  'ace3p_cores' : 16
  'ace3p_opts' : '--cpu-bind=cores'
  'workdir' : 'lume-ace3p_omega3p_workdir'
  'workdir_mode' : 'auto'
  'sweep_output' : True
  'sweep_output_file' : 'omega3p_sweep_output.txt'

input_parameters :
  'cav_radius' :
    'min' : 90.0
    'max' : 120.0
    'num' : 4
  'ellipticity' :
    'min' : 0.5
    'max' : 1.25
    'num' : 4
```

Parameter sweep mode for omega3p

Input filenames

HPC options for ACE3P part of workflow

Working directory settings

Sweep output toggle and filename

Input names must *exactly* match those in Cubit journal

Input values to sweep from ‘min’ to ‘max’ in ‘num’ steps (in this case total of 4x4 workflow evaluations)

Example 2: Omega3P Parameter Sweep: Config. File Setup

Omega3P sweep config. file contents cont.

demo_omega3p_sweep.yaml

```
workflow_parameters : …

input_parameters : …

output_parameters :
  'R/Q' : ['RoverQ', '0', 'RoQ']
  'Mode_freq' : ['RoverQ', '0', 'Frequency']
  'E_max' : ['maxFieldsOnSurface', '6', 'Emax']
  'loc_x' : ['maxFieldsOnSurface', '6', 'Emax_location', 'x']
  'loc_y' : ['maxFieldsOnSurface', '6', 'Emax_location', 'y']
  'loc_z' : ['maxFieldsOnSurface', '6', 'Emax_location', 'z']
```

Output parameters formatted to read specific section blocks in “rfpost.out” file (see LUME-ACE3P guide for details)

RoverQ block format:
- Mode ID: “0”
- Column name: “Frequency”

Parameter keys only used for column names in sweep output file

Section blocks formatted as string lists

maxFieldsOnSurface block format:
- Surface ID: “6”
- Data name 1: “Emax_location”
- Data name 2: “x”

Example 2: Omega3P Parameter Sweep: Outputs

Workflow folders will automatically be created

- Naming structure given in configuration file
  - Parameter values appended automatically to name
- All possible combinations of inputs will be evaluated
  - Note: consider allocating enough time and resources to run all workflows!

Sweep output file contains *only* selected output parameter data

- All “RFpost.out” data parsed and combined into 1 file
- Detailed data can be viewed in corresponding folder
- No plotting tool available yet

Example 2: Omega3p Parameter Sweep: Output File

Omega3P outputs parsed from “rfpost.out” files

omega3p_sweep_output.txt

cav_radius	ellipticity	R/Q	Mode_freq	E_max	loc_x	loc_y	loc_z
90.0	0.5	109.911	1408893500.0	39396300.0	0.042877	0.005648	0.045602
100.0	0.5	129.086	1253557500.0	39584400.0	0.031376	0.029998	-0.04555
110.0	0.5	141.315	1128487300.0	39521500.0	0.041845	0.011548	0.04555
120.0	0.5	148.111	1025946500.0	39509900.0	0.0017699	0.043794	-0.045426
90.0	0.75	104.071	1446025900.0	38818600.0	0.026719	0.032004	0.046238
...							
110.0	1.25	130.122	1202641400.0	38694400.0	0.018834	0.03911	-0.04555
120.0	1.25	140.528	1089757900.0	39148600.0	0.031376	0.029998	-0.04555

Input parameters

Output parameters

- All input parameter combinations swept
- Column header names from config. file
- Tab delimited (useful to import into spreadsheet)

3

Optimization with LUME-ACE3P

Optimization with LUME-ACE3P

LUME-ACE3P integration with Xopt (work in progress)

- Limited to scalar optimization (multi-objective work in progress)
- Limited to S3P S-parameter optimization (Omega3P work in progress)
- Visit LUME-ACE3P website: https://github.com/slaclab/lume-ace3p for latest updates

Optimization algorithms available:

- Nelder-Mead simplex optimization
- Expected-improvement Bayesian optimization
- More algorithms to be added (multi-fidelity coming soon)

To use Xopt in LUME-ACE3P:

- Same input file setup to parameter sweeping except for configuration file

S3P Optimization: Config. File Setup

S3P optimization config. file example contents

demo_omega3p_sweep.yaml

```
workflow_parameters :
   'mode' : 'scalar_optimize'
   'module' : 's3p'
   'cubit_input': 'bend-90degree.jou'
   'ace3p_input': 'bend-90degree.s3p'
   'ace3p_tasks': 16
   'ace3p_cores': 16
   'ace3p_opts' : '--cpu-bind=cores'
   'workdir': 'lume-ace3p_xopt_workdir'

vocs_parameters :
    'variables' :
        'cornercut': [14,17]
        'rcorner1': [0.5,2.5]
    'objectives' :
        's_parameter' : 'S(0,0)'
        'frequency' : 12.0e+09
        'optimization' : 'MINIMIZE'

xopt_parameters :
    'generator' : 'NelderMeadGenerator'
    'num_random' : 0
    'num_step' : 25
```

Optimization mode for s3p

Input filenames

HPC options for ACE3P part of workflow

Working directory settings

Variable names and ranges

S-parameter name, frequency, and goal type (min or max)

Optimization algorithm, initial random evaluations, iteration max number of steps

S3P Optimization: Output Files

Output: “sim_output.txt” – (default name)

- Formatted from Xopt data structure (see https://xopt.xopt.org for more details)
  - Columns are iteration, input parameters, objective value, runtime duration
  - Only optimization objective recorded (i.e. single S-matrix entry at one frequency)
  - Useful for checking optimization progress

Output: “sim_output_all_values.txt” – (default name)

- Formatted similarly to S3P sweep output
  - Columns are iteration #, input parameters, frequency, S-matrix entries
  - Entire “Reflection.out” file saved for each parameter selection (iteration)
  - Useful for plotting/analyzing optimization behavior for S-matrix at many frequencies

4

Additional Comments

Troubleshooting LUME-ACE3P

Troubleshooting checklist:

- Check if *perlmutter-ace3p-spack.sh* is sourced
- Check if LUME-ACE3P Conda environment is loaded
  - Command line should show (lume-ace3p) before username
  - Due to python library conflicts, you may need to load the LUME-ACE3P environment *after* sourcing the setup script (check the $PYTHONPATH variable)
- Check if Cubit journal can handle full range of parameters given
  - Be mindful of sideset ID changes for certain parameter combinations
- Check if batch script settings are sufficient for entire sweep
  - Because the sweeps are done in serial (parallel evaluation not available yet), set enough time for all evaluations in batch script!
- Check https://github.com/slaclab/lume-ace3p website for additional tips and updates
- Visit *#ace3p-user* Slack channel for additional support