ska-web-apps-documentation Documentation

Snehal Valame

Jan 17, 2023

USER GUIDE

The Sensitivity Calculator is an application for calculating the sensitivity of the SKA Mid-Frequency Aperture Array.

This Project was generated with Angular CLI version 13.

Development server

Run *npm start* for a dev server. Navigate to *http://localhost:4200/sensitivity-calculator*. The app will automatically reload if you change any of the source files.

The necessary steps are:

1. Install Git if you don't already have it.

To find if Git is installed on your computer, type in a terminal: git –version. The output will either say which version of Git is installed, or that git is an unknown command.

If Git is not there, point your browser to https://git-scm.com/book/en/v2/Getting-Started-Installing-Git and follow the instructions for installation.

2. Clone the sensitivity calculator from the SKA Git repository by moving to the destination directory on your machine, and typing: git clone https://gitlab.com/ska-telescope/ost/ska-ost-senscalc-ui.git.
3. Run *npm start* for a dev server. Navigate to *http://localhost:4200/sensitivity-calculator*.

Enter the code directory with: cd ska-ost-senscalc-ui ## Code scaffolding

Run *ng generate component component-name* to generate a new component. You can also use *ng generate directive|pipe|service|class|guard|interface|enum|module*.

Build

Run *ng build* to build the project. The build artifacts will be stored in the *dist/* directory.

Running unit tests

Run *make test* to execute the unit tests.

Running static code analysis

Run *make lint* to lint the code.

Installing project dependencies

Run *make install_dependencies* to install latest project dependencies from package.json and package-lock.json

Deploying to Kubernetes

Makefiles with common targets are available in the submodule. To build the Docker image run the following

` make oci-build-all `

The umbrella Helm chart can then be deployed with

` make k8s-install-chart `

and uninstalled with ` make k8s-uninstall-chart `

Once installed, the UI should then be available externally at http://<HOST_IP>/<KUBE_NAMESPACE>/sensitivity-calculator/. If using minikube, run *minikube ip* to find the host IP. *KUBE_NAMESPACE* is set to *ska-ost-senscalc-ui* by default. The backend component will also be deployed to a separate pod, which the web application will make requests to.

Running integration (Cypress) tests

Run *npm run cypress:run* to execute integration tests locally. This requires the application to be running. Note that this also requires the SC back-end to be running and available at the location defined in the [runtime-configuration.json file](./src/ska_ost_senscalc_ui/assets/runtime-configuration.json).

To run integration tests in Kubernetes:

1. Deploy the application to k8s (see above)
2. Set the *baseUrl* in *cypress.json* to point to *http://<HOST_IP>/<KUBE_NAMESPACE>/sensitivity-calculator/*
3. Run tests with *npm run cypress:run*

Deploying to Review Environment

The review environment is specific for each MR and is hosted on STFC and is live for two hours after it's deployed.

There are two review jobs:

- *deploy_review_ui*: This only deploys the frontend and *API_BACKEND_URL* must be set correctly when triggering the job otherwise the frontend won't be able to talk to the backend. The backend is deployed similarly from its own repository.
- *deploy_review_umbrella*: This job deploys both the frontend and backend to the same namespace. The backend used is defined in the chart and a published image in contrast to the above so it's easier to review just the frontend.

The review jobs are automatically run for every push for a branch and can be accessed as *https://k8s.stfc.skao.int/<BRANCH_NAME>*. They can be stopped by triggering the *stop_review* job manually from the pipeline.

Deploying to Staging Environment

The staging environment is a long living deployment in STFC and can be controlled from the manual jobs from the master branch.

There are two environments similar to review environments for only frontend and both frontend and backend. Only difference is with the umbrella chart, *API_BACKEND_URL* is set automatically to the staging backend service.

Deploying to Production environment

Master branch chart can be deployed to production at any time by using the pipeline jobs in deploy stage by triggering a deploy job. Similarly, the existing deployment can be stopped using the stop jobs or GitLab's environment page by following here: https://gitlab.com/ska-telescope/ost/ska-ost-senscalc/-/environments/11337767

The deployment is done against a AWS cluster where the access is restricted.

Restrictions

The production environment is called *production-ska-ost-senscalc* in GitLab's UI and can be seen from here: https://gitlab.com/ska-telescope/ost/ska-ost-senscalc/-/environments/11337767 Only the maintainers can make a deployment and all deployments should be approved from the above interface when a new job is triggered for deployment. These settings can be changed by the project owner or the team-system if needed.

Documentation

Documentation and a User Guide can be found in the `docs` folder

Further help

[![Documentation Status](https://readthedocs.org/projects/ska-telescope-ska-ost-senscalc/badge/?version=latest){]}(https://developer.skao.int/projects/ska-ost-senscalc/en/latest/?badge=latest)

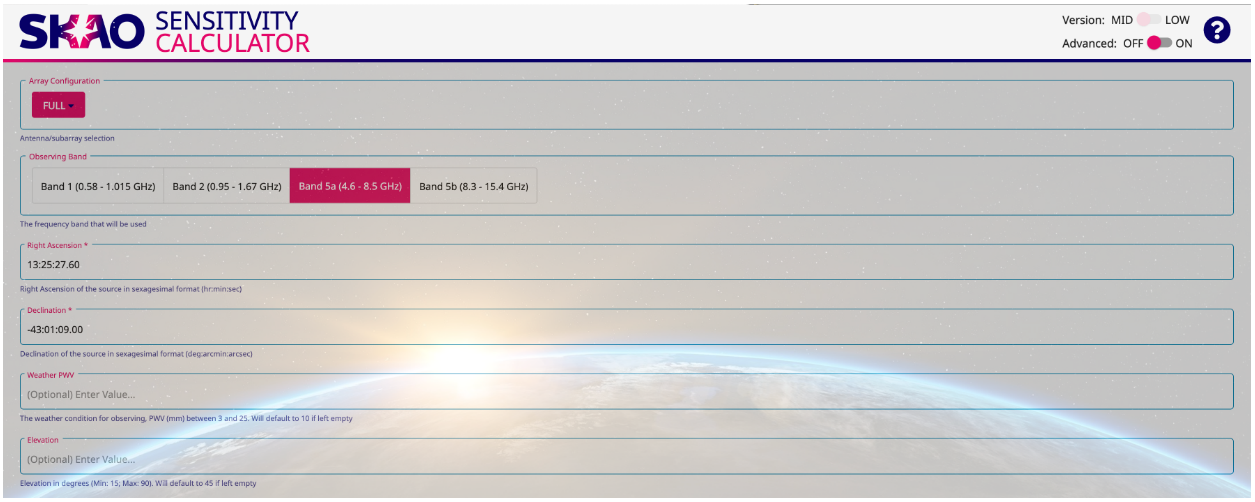
To get more help on the Angular CLI use *ng help* or go check out the [Angular CLI Overview and Command Reference](https://angular.io/cli) page.

CHAPTER ONE

USER GUIDE

Users control the Calculator via a web page interface, published at URL `http://example.com`.

The observing configuration is described by setting values in the web interface. ‘Universal’ inputs required for all observing modes, such as the target position, are displayed in the upper part of the web page as shown in Fig. 1.



Array Configuration
FULL
Antenna/subarray selection

Observing Band
Band 1 (0.58 - 1.015 GHz) | Band 2 (0.95 - 1.67 GHz) | Band 5a (4.6 - 8.5 GHz) | Band 5b (8.3 - 15.4 GHz)
The frequency band that will be used

Right Ascension *
13:25:27.60
Right Ascension of the source in sexagesimal format (hr:min:sec)

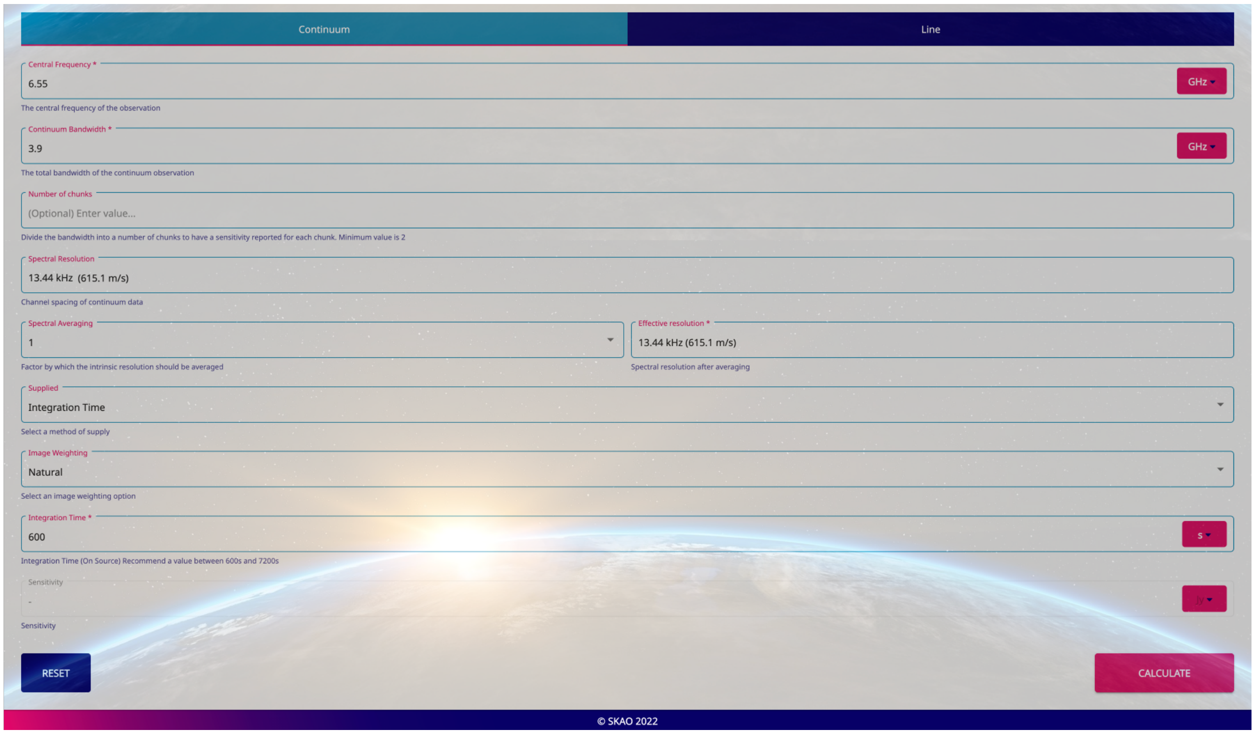
Declination *
-43:01:09.00
Declination of the source in sexagesimal format (deg:arcmin:arcsec)

Weather PWV
(Optional) Enter Value...
The weather condition for observing, PWV (mm) between 3 and 25. Will default to 10 if left empty

Elevation
(Optional) Enter Value...
Elevation in degrees (Min: 15; Max: 90). Will default to 45 if left empty

Figure 1 . Screenshot showing the ‘universal’ part of the Calculator page.

Configuration details that depend on observing mode are separated from the universal inputs by tabs. The User can switch between modes by selecting either the `Continuum` or `Line` tab, as shown in Fig. 2.



Central Frequency *
6.55 GHz
The central frequency of the observation

Continuum Bandwidth *
3.9 GHz
The total bandwidth of the continuum observation

Number of chunks
(Optional) Enter value...
Divide the bandwidth into a number of chunks to have a sensitivity reported for each chunk. Minimum value is 2

Spectral Resolution
13.44 kHz (615.1 m/s)
Channel spacing of continuum data

Spectral Averaging
1
Factor by which the intrinsic resolution should be averaged

Effective resolution *
13.44 kHz (615.1 m/s)
Spectral resolution after averaging

Supplied
Integration Time
Select a method of supply

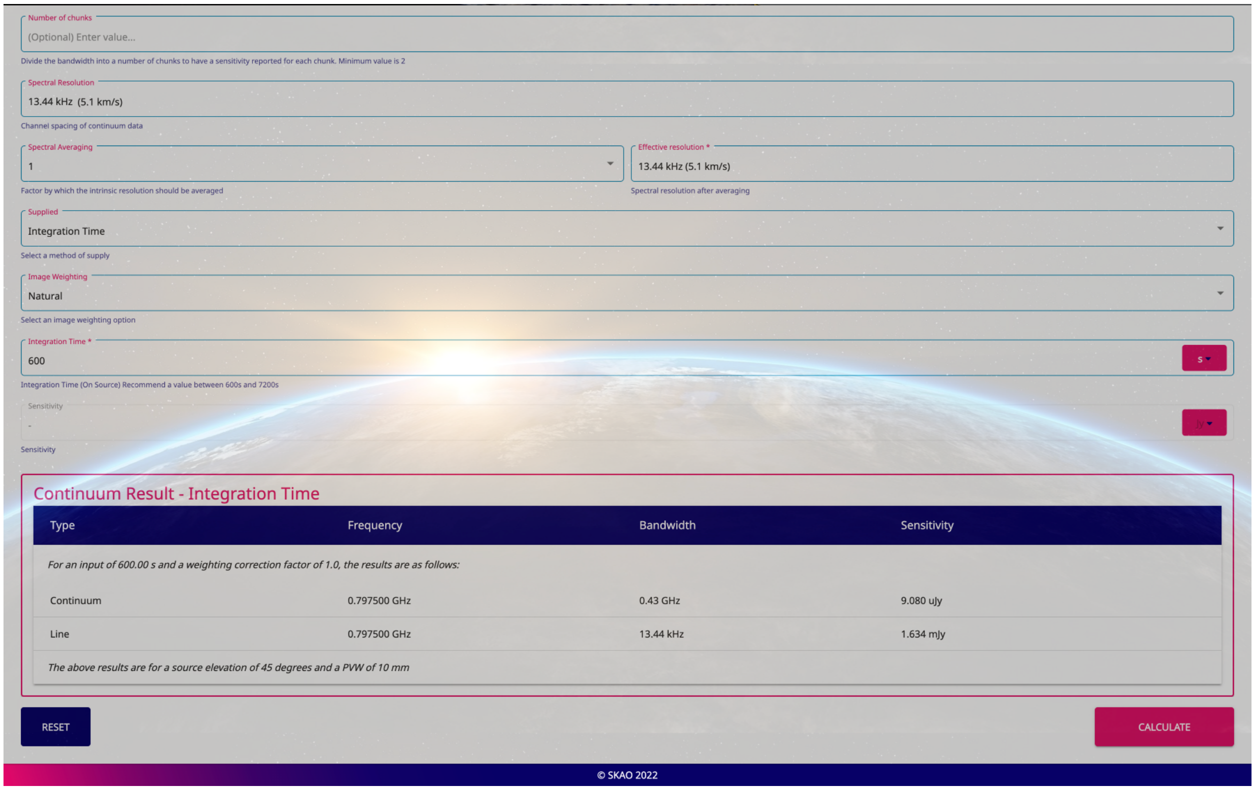
Image Weighting
Natural
Select an image weighting option

Integration Time *
600 s
Integration Time (On Source) Recommend a value between 600s and 7200s

Sensitivity
-
Sensitivity

Figure 2 . Screenshot showing the tab for `Continuum` mode.

Once configured the Calculator can be used to either calculate the sensitivity for a given on-source integration time, by selecting 'Integration Time' under the `Supplied` drop down, entering the time and clicking `calculate`, or calculate the integration time required to reach a given sensitivity, by selecting 'Sensitivity' under the `Supplied` drop down, entering the sensitivity and clicking `calculate`. Fig. 3 shows an example report provided to the user when this is done.



Number of chunks
(Optional) Enter value...
Divide the bandwidth into a number of chunks to have a sensitivity reported for each chunk. Minimum value is 2

Spectral Resolution
13.44 kHz (5.1 km/s)
Channel spacing of continuum data

Spectral Averaging
1
Factor by which the intrinsic resolution should be averaged

Effective resolution *
13.44 kHz (5.1 km/s)
Spectral resolution after averaging

Supplied
Integration Time
Select a method of supply

Image Weighting
Natural
Select an image weighting option

Integration Time *
600 s
Integration Time (On Source) Recommend a value between 600s and 7200s

Sensitivity
- Jy
Sensitivity

Continuum Result - Integration Time

Type	Frequency	Bandwidth	Sensitivity
For an input of 600.00 s and a weighting correction factor of 1.0, the results are as follows:			
Continuum	0.797500 GHz	0.43 GHz	9.080 uJy
Line	0.797500 GHz	13.44 kHz	1.634 mJy
The above results are for a source elevation of 45 degrees and a PVW of 10 mm			

Figure 3 : Screenshot showing the report for the total continuum noise for a hypothetical observation. No PWV (Precipitable Water Vapour) or elevation angle were specified, and so the calculation is performed for default values.

1.1 Inputs

The calculator inputs can be categorised by the observing mode they fall under. **Universal** inputs are those that apply regardless of the selected observing mode.

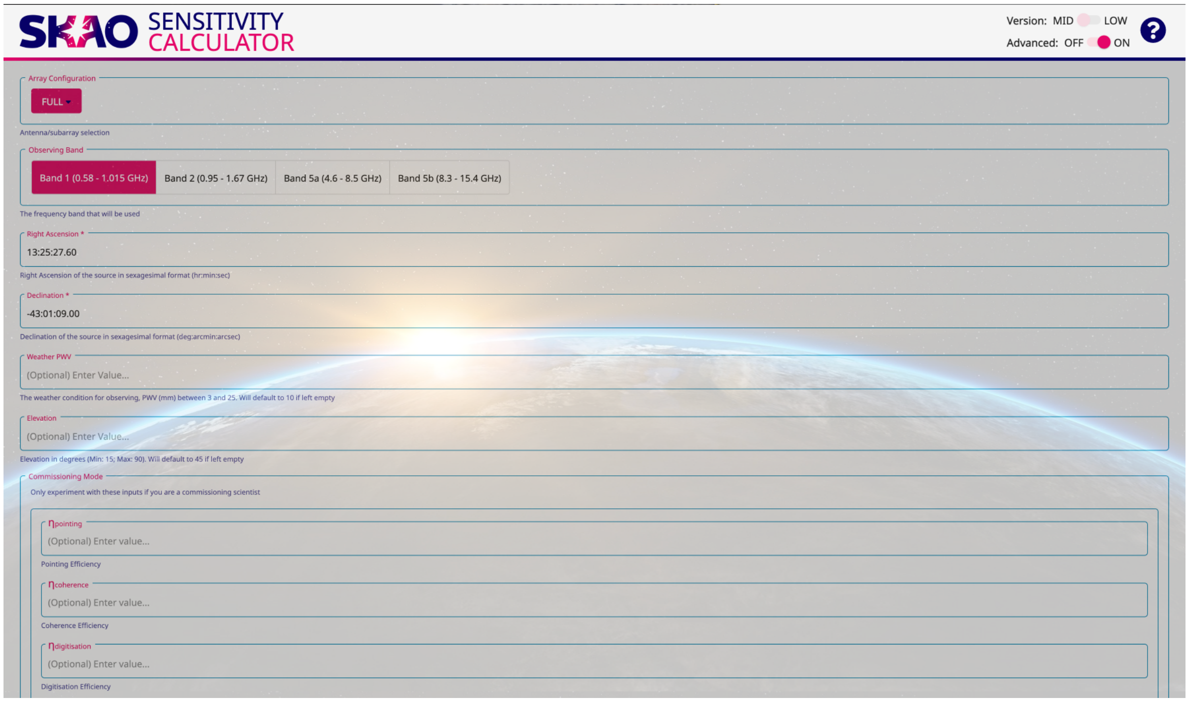
1.1.1 Universal

- **Observing Band** The selected band to use for the observation. For the "full" subarray the frequency ranges for the bands are:
  - Band 1: 0.58GHz - 1.015GHz
  - Band 2: 0.95GHz - 1.67GHz
  - Band 5a: 4.6GHz - 8.5GHz
  - Band 5b: 8.3GHz - 15.4GHz
- **Right Ascension** and **Declination** The equatorial coordinates of the observed source. The sensitivity is calculated for the time at which the target reaches its maximum elevation, crossing the meridian.
- **Array Configuration** Preset list of array configurations. Click on the tab to choose from:
  - SKA1 (133 x 15m): just the SKA1 antennas
  - full: all SKA1 and MeerKAT antennas
  - meerKAT: just the MeerKAT antennas

Not currently available:

- custom: activates the nSKA and nMeer fields where the user can enter the number of SKA and MeerKAT antennas directly.
- core
- extended

- **Weather PWV** If no value is set in the weather PWV (Precipitable Water Vapour) field then results will be given for the default value of 10mm, corresponding to "average" conditions. The PWV is used in the calculation of the atmospheric brightness temperature, T_{atm}. Since T_{sys} is dependent on T_{sky} and therefore T_{atm} and the weather conditions, if the user decides to manually edit T_{sys} in 'Advanced mode', or any of the variables it depends on, the option to set the PWV will be removed.
- **Elevation** The user can use this field to specify the elevation at which the target will be observed. If no value is set then a default of 45 degrees is assumed. If the given elevation is never reached by the target, then the target's zenith elevation will be used. The actual elevation assumed for the sensitivity calculation is reported in the result table.
- **'Advanced Mode' Inputs** By activating the 'Advanced' switch in the top right hand corner of the calculator, the user is given direct access to some of the parameters used in the sensitivity calculation, as shown in Fig. 4. The calculator front-end automatically enables/disables inputs to avoid conflicts as the user selects which one they want to edit. These are passed to the calculator back-end as hard-coded values which will override the calculated defaults.



SKAO SENSITIVITY CALCULATOR

Version: MID LOW

Advanced: OFF ON

Array Configuration

FULL

Antenna/subarray selection

Observing Band

Band 1 (0.58 - 1.015 GHz) | Band 2 (0.95 - 1.67 GHz) | Band 5a (4.6 - 8.5 GHz) | Band 5b (8.3 - 15.4 GHz)

The frequency band that will be used

Right Ascension *

13:25:27.60

Right Ascension of the source in sexagesimal format (hr:min:sec)

Declination *

-43:01:09.00

Declination of the source in sexagesimal format (deg:arcmin:arcsec)

Weather PWV

(Optional) Enter Value...

The weather condition for observing, PWV (mm) between 3 and 25. Will default to 10 if left empty

Elevation

(Optional) Enter Value...

Elevation in degrees (Min: 15; Max: 90). Will default to 45 if left empty

Commissioning Mode

Only experiment with these inputs if you are a commissioning scientist

ηpointing

(Optional) Enter value...

Pointing Efficiency

ηcoherence

(Optional) Enter value...

Coherence Efficiency

ηdigitisation

(Optional) Enter value...

Digitisation Efficiency

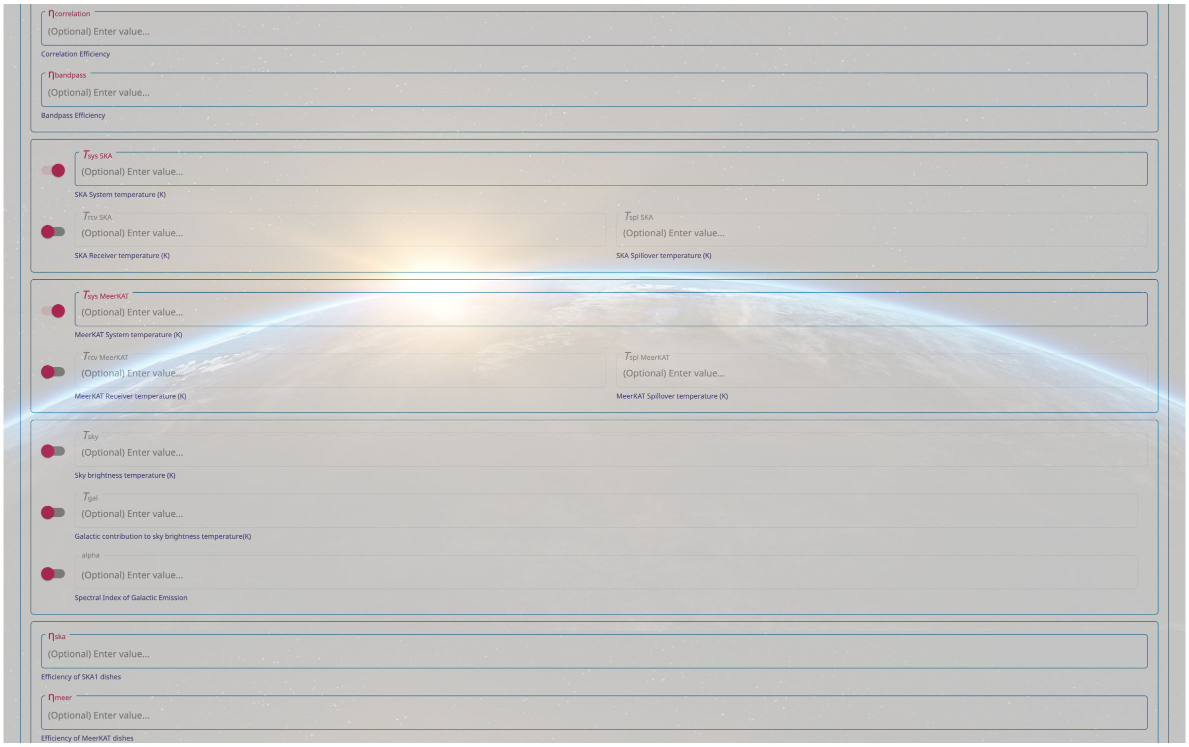

$\eta_{correlation}$
(Optional) Enter value...
Correlation Efficiency
$\eta_{bandpass}$
(Optional) Enter value...
Bandpass Efficiency
$T_{sys\ SKA}$
(Optional) Enter value...
SKA System temperature (K)
$T_{rcv\ SKA}$
(Optional) Enter value...
SKA Receiver temperature (K)
$T_{spl\ SKA}$
(Optional) Enter value...
SKA Spillover temperature (K)
$T_{sys\ MeerKAT}$
(Optional) Enter value...
MeerKAT System temperature (K)
$T_{rcv\ MeerKAT}$
(Optional) Enter value...
MeerKAT Receiver temperature (K)
$T_{spl\ MeerKAT}$
(Optional) Enter value...
MeerKAT Spillover temperature (K)
T_{sky}
(Optional) Enter value...
Sky brightness temperature (K)
T_{gal}
(Optional) Enter value...
Galactic contribution to sky brightness temperature(K)
alpha
(Optional) Enter value...
Spectral Index of Galactic Emission
η_{ska}
(Optional) Enter value...
Efficiency of SKA1 dishes
η_{meer}
(Optional) Enter value...
Efficiency of MeerKAT dishes

Figure 4 . Expanded view of an example use-case for the additional inputs on the 'Advanced mode' version of the calculator.

1.1.2 Continuum

- **Central Frequency** The central frequency for the observation. Must be within the selected band.
- **Bandwidth** The bandwidth for the observation. Must be fully contained within the selected band.
- **Number of chunks** The user can select an integer number of chunks to split the bandwidth up into. If they do, the output report will show the sensitivity (or integration time) for each chunk.
- **Spectral Averaging** The factor by which the intrinsic spectral resolution is averaged.
- **Supplied** Drop down allowing the user to swap between integration time and sensitivity as the input (giving the other as the output).
- **Image Weighting** The image weighting option to be used.
- **Integration Time** The integration time of the observation. Used when calculating the sensitivity that observing for this amount of time will achieve.
- **Sensitivity** The sensitivity for the observation. Used when calculating the integration time necessary to achieve this sensitivity.

1.1.3 Line

- **Zoom Frequency** For each zoom, the user can input a frequency for that zoom. When a value is entered, the next zoom becomes enabled, allowing a value to be entered. It can however be left blank, and the calculation will only be done for zooms which have a set frequency. This way, the user can select how many zooms they want (up to a maximum, currently 4).
- **Zoom Bandwidth and Resolution** For each zoom, the user can set a line resolution.
- **Supplied** Drop down allowing the user to swap between integration time and sensitivity as the input (giving the other as the output).
- **Image Weighting** The image weighting option to be used.
- **Integration Time** The integration time of the observation. Used when calculating the sensitivity that observing for this amount of time will achieve.
- **Sensitivity** The sensitivity for the observation. Used when calculating the integration time necessary to achieve this sensitivity.

CHAPTER

TWO

SENSITIVITY MODEL

The ‘system equivalent flux density’ (SEFD) for a single dish is given by:

$$SEFD_{dish} = \frac{2kT_{sys}}{\eta_A A}$$

where:

- k is the Boltzmann constant so that kT_{sys} measures the power received from background emission and all other sources of unwanted signal within the system, that is $T_{sys} = T_{spl} + T_{sky} + T_{rcv} + T_{cmb} + ...$
- η_A is the dish efficiency
- A is the geometric dish area.

The SEFD for an interferometer array made up of two types of dish is given by:

$$SEFD_{\mathrm{array}} = \frac{1}{\sqrt{\frac{n_{\mathrm{SKA}}(n_{\mathrm{SKA}}-1)}{SEFD_{\mathrm{SKA}}^2} + \frac{2n_{\mathrm{SKA}}n_{\mathrm{MeerKAT}}}{SEFD_{\mathrm{SKA}}SEFD_{\mathrm{MeerKAT}}} + \frac{n_{\mathrm{MeerKAT}}(n_{\mathrm{MeerKAT}}-1)}{SEFD_{\mathrm{MeerKAT}}^2}}}$$

where n_{SKA} is the number of SKA antennas, n_{MeerKAT} is the number of MeerKAT antennas, $SEFD_{\mathrm{SKA}}$ is the SEFD computed for an individual SKA antenna, and $SEFD_{\mathrm{MeerKAT}}$ is the SEFD computed for an individual MeerKAT antenna.

We define the telescope sensitivity here as the minimum detectable Stokes I flux (1 σ). This is equal to the noise on the background power, obtained using the radiometer equation $\sigma = SEFD/\sqrt{2Bt}$, corrected for atmospheric absorption:

$$\Delta S_{min} \exp(-\tau_{atm}) = \frac{SEFD_{array}}{\eta_s \sqrt{2Bt}} Jy$$

where:

- ΔS_{min} is the source flux density above the atmosphere
- η_s is the efficiency factor of the interferometer
- B is bandwidth
- t is integration time
- τ_{atm} is the optical depth of the atmosphere towards the target

See implementation for more details.

CHAPTER

THREE

DESIGN

3.1 Public and 'Expert' Users

The calculator is intended for two types of user.

The first type is the ordinary observer who will use the calculator to simply calculate the performance of the telescope when looking at their target object.

The second type is the 'expert' user, who understands the telescope design and wants to test the effect of tweaking some aspect of it. This mode of use is intended for SKA staff. It allows the user to manually edit some of the values which are usually calculated automatically as part of the sensitivity calculation. Say a user wanted to test out how a different array configuration might affect the sensitivity of a given observation. They could manually edit the number of SKA1 and MeerKAT dishes in the array and these would override the numbers that the calculator uses and use the new values in the sensitivity calculation.

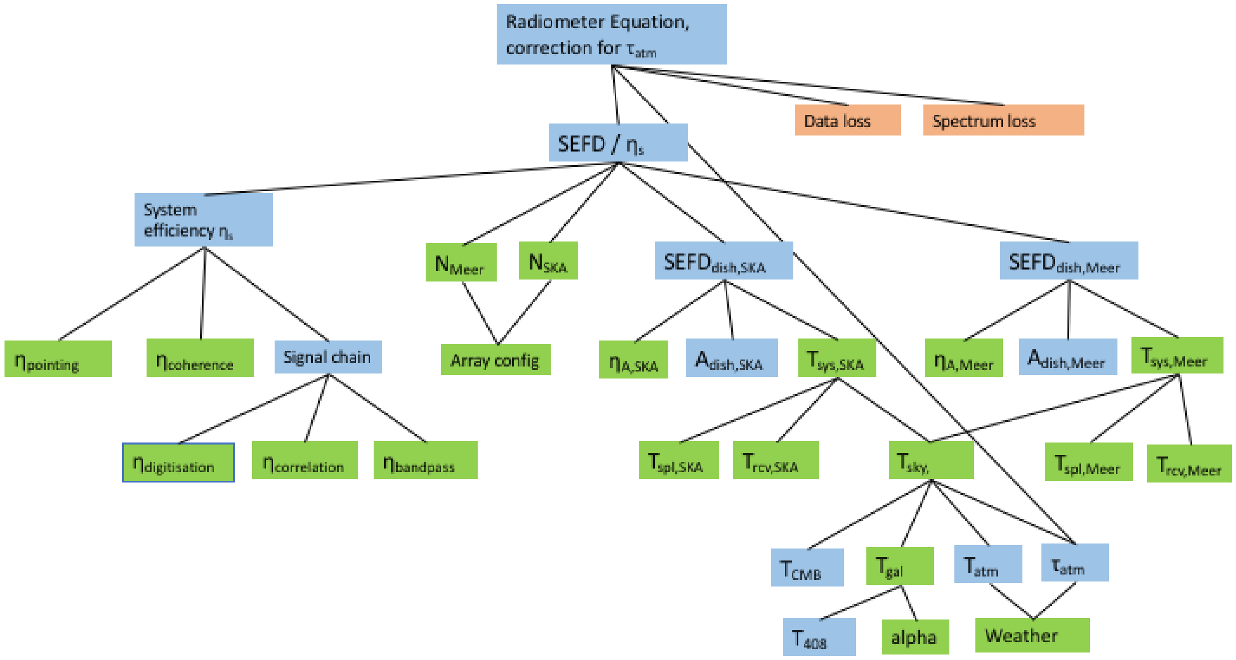


Figure 1 . Flowchart diagram showing the dependencies of the variables used in the sensitivity calculation.

3.2 Technologies

TODO

3.3 High Level Architecture

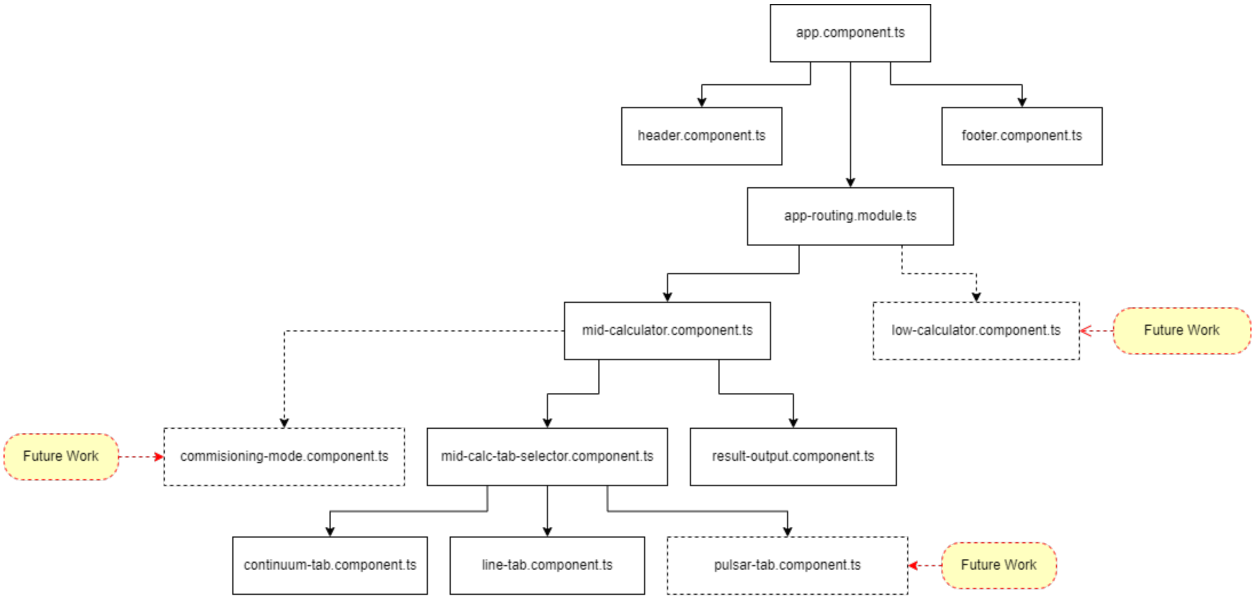

app.component.ts
header.component.ts
footer.component.ts
app-routing.module.ts
mid-calculator.component.ts
low-calculator.component.ts
Future Work
Future Work
commisioning-mode.component.ts
mid-calc-tab-selector.component.ts
result-output.component.ts
continuum-tab.component.ts
line-tab.component.ts
pulsar-tab.component.ts
Future Work

Figure 2 . Flowchart diagram showing the different parent and child angular components that make up the UI structure

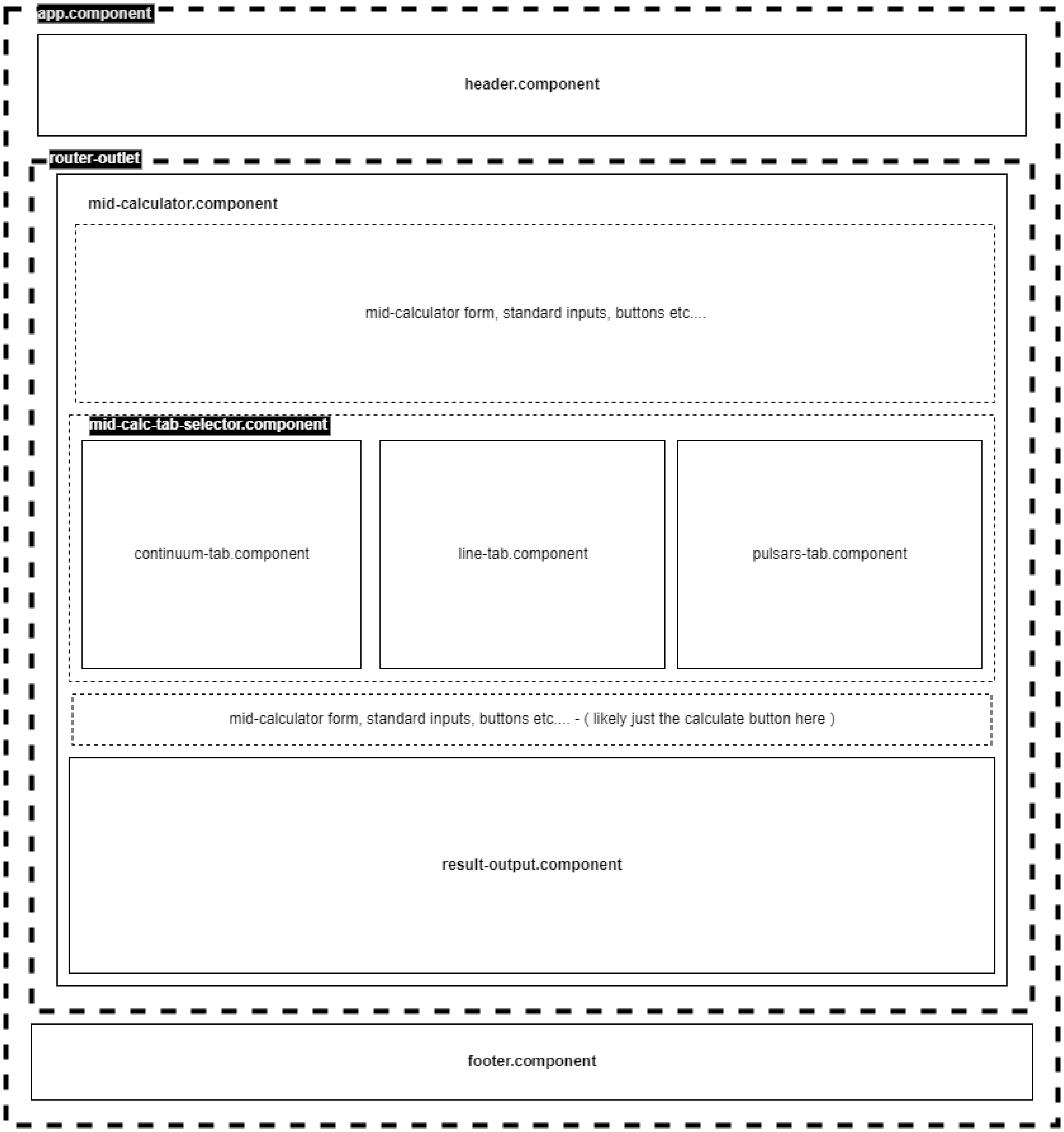

app.component
header.component
router-outlet
mid-calculator.component
mid-calculator form, standard inputs, buttons etc....
mid-calc-tab-selector.component
continuum-tab.component
line-tab.component
pulsars-tab.component
mid-calculator form, standard inputs, buttons etc.... - (likely just the calculate button here)
result-output.component
footer.component

Figure 3 . High level diagram showing the different parent and child angular components that make up the UI structure in a nested format

3.4 Low Fidelity Prototype

Figure 4 . A Low Fidelity prototype that is built in InVision Studio. This is a screenshot from that prototype showing the design of how the UI could look pre-development.

CHAPTER

FOUR

DEVELOPMENT PLAN

4.1 PI13 Iteration 1

The suggested development plan for iteration 1 of the Angular Sensitivity Calculator is described in the plan diagram in Figure 1.

The main key point for development is to have two Angular developers working in parallel on the application to complete non blocking tasks.

The first 3 steps are core tasks that must be performed in order to get the large majority of the sensitivity calculator hanging together.

By specifying the tasks for each developer in step 1 to 3, it will prevent each developer waiting too long on something from the previous step being completed.

The first 6 tasks have been split out as fairly as possible in terms of size and complexity, so each developer will be busy and not overloaded.

For example, the Mid-Calculator Component and Mid-Calc-Tab-Selector Component depends on some sort of core SCSS styles/UX to be available for consistent look and feel for the componments so these must wait on step 2, but the Header Compomnent can probably be its own entity for the majority and tidy up with some parent level CSS at the end so it can be developed in parallel to the SCSS.

The final steps are not attached to either developer but are still in some sort of priority by steps to drive focus. When a developer reaches step 4 and beyond, they can pick up any task in that step (as long as its not been picked up already by the other dev). This also would allow a third developer to come onboard and pick up these later tasks if we need support for deadlines.

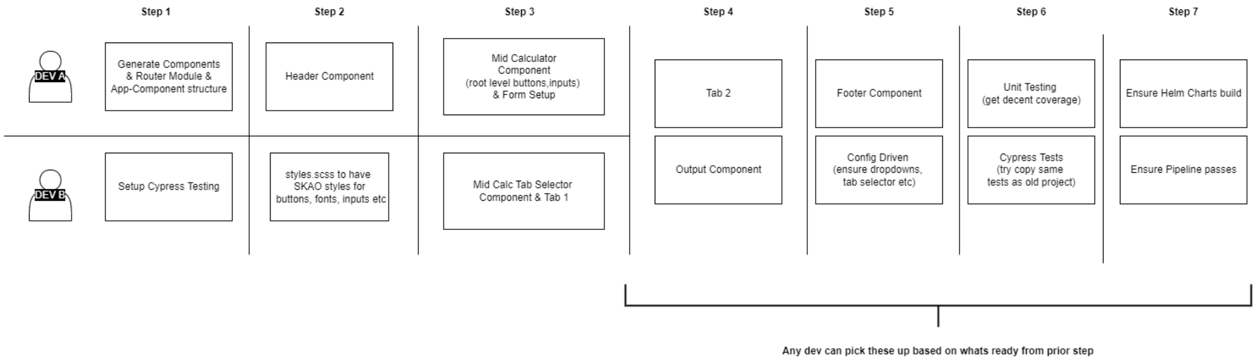

Step 1
Step 2
Step 3
Step 4
Step 5
Step 6
Step 7
DEV A
Generate Components & Router Module & App-Component structure
Header Component
Mid Calculator Component (root level buttons,inputs) & Form Setup
Tab 2
Footer Component
Unit Testing (get decent coverage)
Ensure Helm Charts build
DEV B
Setup Cypress Testing
styles.scss to have SKAO styles for buttons, fonts, inputs etc
Mid Calc Tab Selector Component & Tab 1
Output Component
Config Driven (ensure dropdowns, tab selector etc)
Cypress Tests (try copy same tests as old project)
Ensure Pipeline passes
Any dev can pick these up based on whats ready from prior step

Figure 1 . Diagram shows the flow of work for two developers working in parallel.

Risks :

Availability in Sprint 4/5 is a risk, as these are already bloated sprints, so we have noted as a team we could descope Cypress testing to PI14 if required.