# Telescope Time Allocation Tools

## Algorithms & Subsystems

Allison H. Costa, Jeffrey S. Kern, Dana S. Balser, & Mark Whitehead

*last modified: July 24, 2023*

# Contents

# Part I

# Overview

# 1 Introduction

## 1.1 Scope of Document

The intended use of this document is to have a common place to house the information related to the Telescope Time Allocation (TTA) algorithms and subsystems. The intended audience is the implementation team and scientists who want to understand the functional details of the algorithms.

## 1.2 Related Documents

- 688-TTAT-002-MGMT System Concept
- 688-TTAT-004-MGMT System Description v3.0
- 688-TTAT-007-MGMT Subsystem Description
- 688-TTAT-xxx-MGMT TTA Use Cases v0.1

## 1.3 Document Conventions

Several formatting conventions are used in this document for emphasis.

Structures within the system are *italicized*: *Solicitation*, *Facility*, *Capability*, *Proposal*, *Allocation Request*, *Capability Request*, *Capability Parameter Specifications*, *Observation Specification*, *Scan*, *Subscan*, etc.

When referring to a specific field or value in the structure, SMALL CAPITALS are used. For example, *Capability Parameter Specifications* are the parameters that make up a *Capability*. There are types of *Capability Parameter Specifications*, such as SPECTRAL SPECIFICATIONS. More examples that use this format include SOURCE, HARDWARE CONFIGURATION, FIELD SOURCE, CALIBRATION PARAMETERS, and PERFORMANCE PARAMETERS.

To emphasize specific parameters the algorithm uses or calculates, `monospace font` is used. Typically, these parameters are specific components within a structure and refer to information from the *Capability Request*, the *Capability*, or *Specification Constraints* (e.g., `Center Frequency` is a *Capability Request Parameter* and in the category of SPECTRAL SPECIFICATION; `Settle Time` is a *Specification Constraint* that is specific to a *Facility*). It can also be calculated parameters determined by the algorithm, e.g., the `Requested Time`. Title Case can be used in tandem for emphasis.

*`Italicized monospace font`* is used for emphasizing methods in the algorithm that have actions, e.g., *`Requested Time Generator`* calculates the `Requested Time`.

### 1.3.1 Domain Driven Design

intro...

- Consensus Review is not in the domain model, though it is in the TTA member vernacular. The domain entity is called the Proposal Review, which hosts the "Comments to PI" and "Internal Comments".

# 2 How to Navigate this Document

This document is provides the functional details of the subsystems and algorithms for the TTA tools from Solicitation creation to Project Creation and Closeout processes. Figure 2.1 shows the workflow through the process. The Telescope Time Allocation (TTA) Group member initiates a *Solicitation* through Solicitation Configuration. During configuration, the TTA member specifies the parameters of the *Solicitation*, which are described in Chapter 3. During the Call Period of a *Solicitation*, proposers can create and submit *Proposals*. Chapters in Part IV detail the algorithms that translate the user's science request into a system generated request for time.

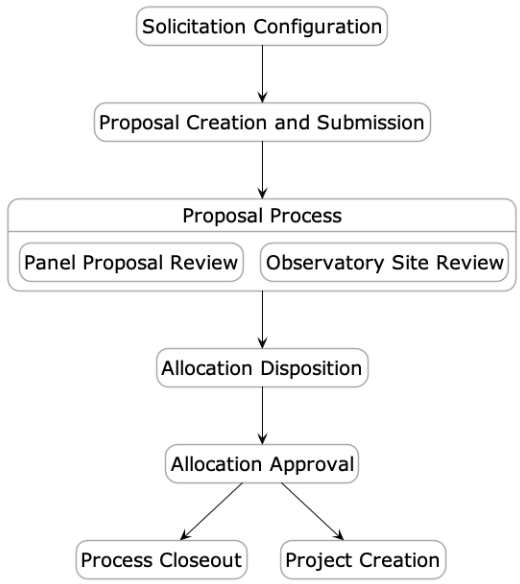



**Figure 2.1:** An abbreviated replica of Figure 6 of 688-TTAT-004-MGMT System Description v3.0.

# Part II

# Solicitation Configuration Subsystem

# 3 Overview of Solicitation Configuration

The TTA member is responsible for configuring a *Solicitation*, which is supported in the Solicitation UI. A *Solicitation* has the following attributes that are specified by the TTA member:

- Call Period
- Execution Period
- Grace Period
- Proposal Process
- Demo Status
- Science Categories
- Notifications
- Facilities
- Capabilities

# 4 Capabilities

A *Capability* corresponds to the different ways a *Facility* may be operated and the resources available. It is convenient to group the *Capabilities* into OBSERVING TYPES. Section B defines the groups as:

- Continuum
- Spectral Line
- Pulsar
- Radar

A *Capability* is composed of parameters which facilitate the proposal creation process; these are called *Capability Parameter Specifications*. The TTA member specifies the *Capability Parameter Specifications* at Solicitation Configuration.

## 4.1 Capability Parameter Specifications

A *Capability* is composed of *Capability Parameter Specifications*. It is convenient to group these parameters; Section 3.1 and Table 2 of the 688-TTAT-004-MGMT System Description v3.0 lists the groups as

- FIELD SOURCE
- SPECTRAL SPECIFICATION
- PERFORMANCE PARAMETERS
- CALIBRATION PARAMETERS

Section C of the 688-TTAT-xxx-MGMT TTA Use Cases v0.1 lists the full set of *Capability Parameter Specifications* by group. A specific subset of *Capability Parameter Specifications* available to a *Capability* is what differentiates one *Capability* from another and dictates how the user's request (i.e., *Capability Request*) can be processed by the TTAT algorithms. In this section, we provide a table of *Capability Parameter Specifications* per *Capability* (Tables 25.2 – 25.41); see Table 4.1 for reference.

**Table 4.1:** Reference for Table Number per *Capability* and *Capability Parameter Specifications* group

| *Capability* | FIELD SOURCE | SPECTRAL SPECIFICATION | PERFORMANCE PARAMETERS | CALIBRATION PARAMETERS |
|---|---|---|---|---|
| GBT Continuum | 25.2 | 25.3 | 25.4 | 25.5 |
| GBT Pulsar | 25.6 | 25.7 | 25.8 | 25.9 |
| GBT Radar | 25.10 | 25.11 | 25.12 | 25.13 |
| GBT Spectral Line | 25.14 | 25.15 | 25.16 | 25.17 |
| VLA Continuum | 25.18 | 25.19 | 25.20 | 25.21 |
| VLA Pulsar | 25.22 | 25.23 | 25.24 | 25.25 |
| VLA Spectral Line | 25.26 | 25.27 | 25.28 | 25.29 |
| VLBA Continuum | 25.30 | 25.31 | 25.32 | 25.33 |
| VLBA Pulsar | 25.34 | 25.35 | 25.36 | 25.37 |
| VLBA Spectral Line | 25.38 | 25.39 | 25.40 | 25.41 |

*Capability Parameter Specifications* can have default inputs, which are noted in Table 4.1. There may also be additional units or inputs available to a *Capability Parameter Specification*. If applicable, a reference is given for an entry in Table 25.1.

# Part III

# Proposal Creation Subsystems

# 5 Capability Requests

The *Capability Request* is composed of the *Capability Request Parameters*, which are the user's response to the *Capability Parameter Specifications*, and they provide the user supplied information about the proposer's request.

A single set of CALIBRATION PARAMETERS define any one *Capability Request*. However, PERFORMANCE PARAMETERS can be specified for each FIELD SOURCE + SPECTRAL SPECIFICATION pair. Table 2 of 688-TTAT-004-MGMT System Description v3.0 provides additional information on the multiplicity of the *Capability Parameter Specifications*.

The *Capability Request* is passed downstream to the algorithms to construct one or more *Observation Specifications*. The remainder of this document provides the details of the algorithms needed to create an *Observation Specification*.

# Part IV

# Proposal Creation Algorithms

# 6 Overview of the Observing Strategy Algorithm

The *Observing Strategy* algorithm generates a normalized data structure called the *Science Target List*, which contains the fundamental user request. Each entry in the *Science Target List* has three components: a Science Target, a *Calibration Strategy*, and a *Scheduling Strategy*.

- *Calibration* and *Scheduling Strategies* are *Capability* specific factories that produce the instructions for the *Observation Planner*.
- A Science Target consists of a SOURCE, a HARDWARE CONFIGURATION, and a `Requested Time`.
  - A SOURCE is derived from a FIELD SOURCE or created for a CALIBRATOR. A SOURCE is a normalized data structure that has
    * a name,
    * a `Pointing Pattern`,
    * a nominal position from the `Pointing Pattern`.
  - A `Pointing Pattern` describes the trajectory of an antenna over the course of an observation of a FIELD SOURCE. Pointing Patterns are *Facility* dependent.
- The HARDWARE CONFIGURATION describes the FRONT-END and BACK-END of a *Facility* that best suits the requested SPECTRAL SPECIFICATION and PERFORMANCE PARAMETERS of the *Capability Request Parameters*. If applicable, it also specifies the Array Configuration or the Array Subset, where an Array Configuration is an antenna pattern of the VLA (e.g., A-array, B-array), and an Array Subset is a one or more antenna stations associated with the VLBA.
- The `Requested Time` is an initial approximation of the time spent by the antenna(s) collecting data that accounts for the requested `RMS Sensitivity`, the `Pointing Pattern`, and additional considerations described in Section 21.3.

Once an Observing Type is specified in the *Allocation Request*, the *Capability* selects the family of algorithms that the *Observing Strategy* implements to make the *Science Target List*. First, the *Observing Strategy* ingests the appropriate *Capability Request Parameters* for the specified Observing Type. The algorithm creates the *Science Target List* and populates each row with a prototype Science Target and *Capability* specific *Calibration* and *Scheduling Strategies*. A prototype Science Target does not have a specified SOURCE, HARDWARE CONFIGURATION, or `Requested Time`: these attributes are created by the internal algorithms of the *Observing Strategy*. The number of entries in the *Science Target List* is determined....

- TBD: How the algorithm handles field sources with similar position, SS, etc
- TBD: How the algorithm treats actual duplicated entries in CR.

The generation of the *Science Target List* is *Capability* specific; the general workflow of the *Observing Strategy* are shown in Figure 6.1. The Chapters 7 – 8 detail the *Capability* specific implementation of the *Observing Strategy* for the GBT, VLA, and VLBA.

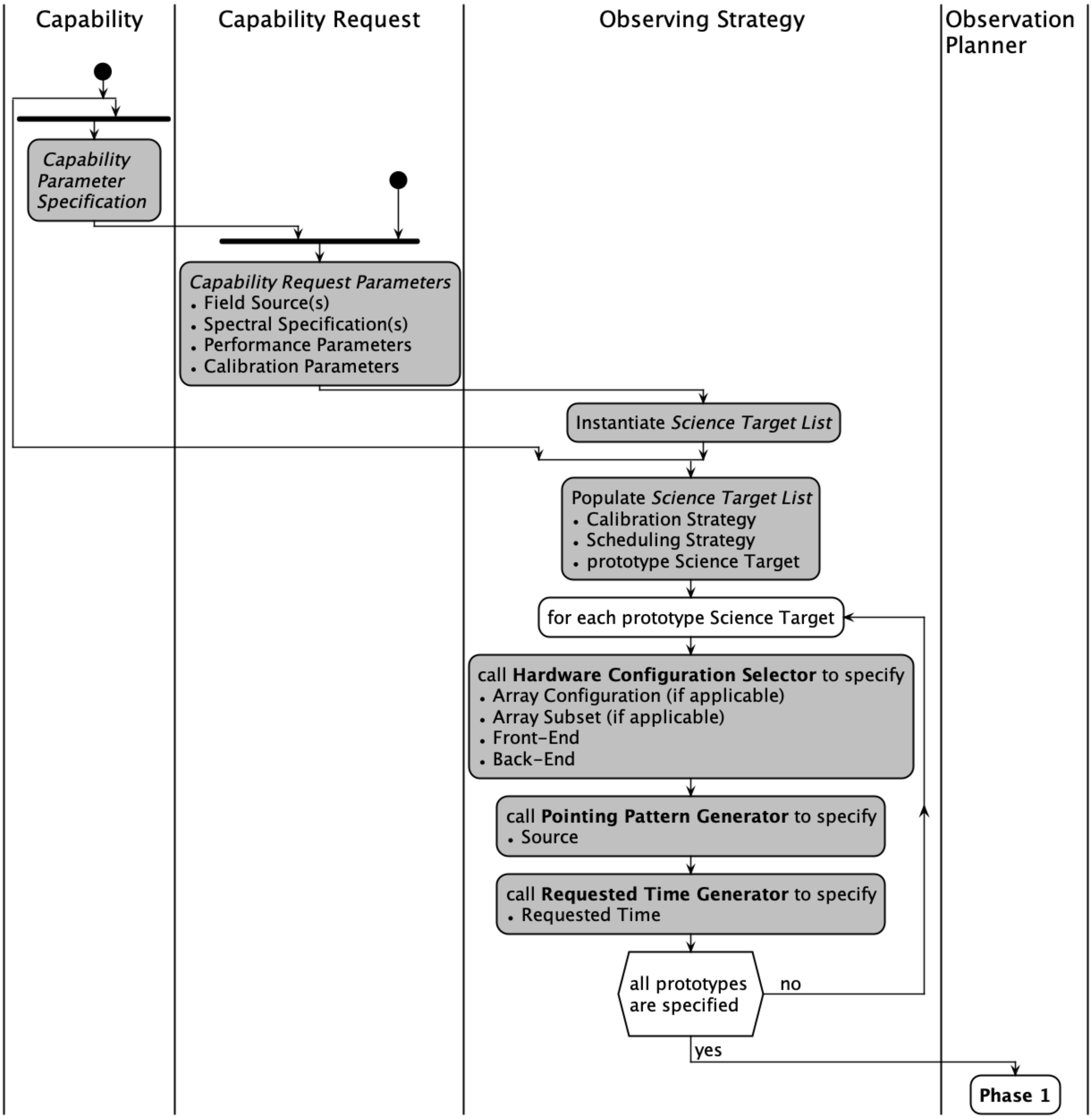



**Figure 6.1:** Overview of the algorithms implemented by the *Observing Strategy.* The black dot (i.e., the "start" symbol in the PlantUML language) in the *Capability Request* column, or "swim lane", represents a user's point of interaction to respond to the *Capability Parameter Specifications.*

# 7 VLA Observing Strategy

**Table 7.1:** VLA Receivers and BACK-ENDS: capabilityConfig.json

| here's a test table |
|---|

For each row in the *Science Target List*, the *Observing Strategy* implements a series of algorithms to specify the SOURCE, HARDWARE CONFIGURATION, and `Requested Time` for a prototype Science Target. The VLA *Observing Strategy* calls the VLA ***`Hardware Configuration Selector`***, which determines the FRONT-END, BACK-END, and VLA Configuration. Next, it calls the VLA ***`Pointing Pattern Generator`***, which specifies the SOURCE and then it calls the ***`Requested Time Generator`***. The steps outlined here are shown in Figure 6.1.

The sections in this chapter provide the details of these algorithms for VLA *Capabilities*. Table 7.2 lists relevant sections for the *Observing Strategy* algorithms, and Table 7.3 presents the same information as Table 7.2 but organized by Observing Type. As an example, if VLA Continuum *Capability* is selected, the *Observing Strategy* uses the algorithms referenced in column 2 of Table 7.3 to construct the *Science Target List*. Whereas, a VLA Pulsar Capability follows column 4.

**Table 7.2:** Overview of VLA *Observing Strategy*.

| Algorithm | § |
|---|---|
| VLA Hardware Configuration | 7.1 |
| VLA Configuration Selector | 7.1.1 |
| VLA FRONT-END Selector | |
| VLA Continuum | 7.1.2 |
| VLA Spectral Line | 7.1.3 |
| VLA Pulsar | 7.1.4 |
| VLA BACK-END Selector | |
| VLA Continuum | 7.1.5 |
| VLA Spectral Line | 7.1.6 |
| VLA Pulsar | 7.1.7 |
| VLA Pointing Pattern Generator | 7.2 |
| VLA Requested Time Generator | 7.3 |

**Table 7.3:** VLA *Observing Strategy* Algorithms by Observing Types

| Algorithm | VLA Continuum § | VLA Spectral Line § | VLA Pulsar § |
|---|---|---|---|
| VLA Hardware Configuration | 7.1 | 7.1 | 7.1 |
| Array Configuration Selector | 7.1.1 | 7.1.1 | 7.1.1 |
| VLA FRONT-END Selector | 7.1.2 | 7.1.2 | 7.1.4 |
| VLA BACK-END Selector | 7.1.5 | 7.1.6 | 7.1.7 |
| VLA Pointing Pattern Generator | 7.2 | 7.2 | 7.2 |
| VLA Requested Time Generator | 7.3 | 7.3 | 7.3 |

## 7.1 VLA Hardware Configuration Selector

The VLA ***Hardware Configuration Selector*** algorithm selects the FONT-END, BACK-END, and VLA Configuration that best matches the *Capability Request Parameters*. An outline of the algorithm is shown in Figure 7.1.

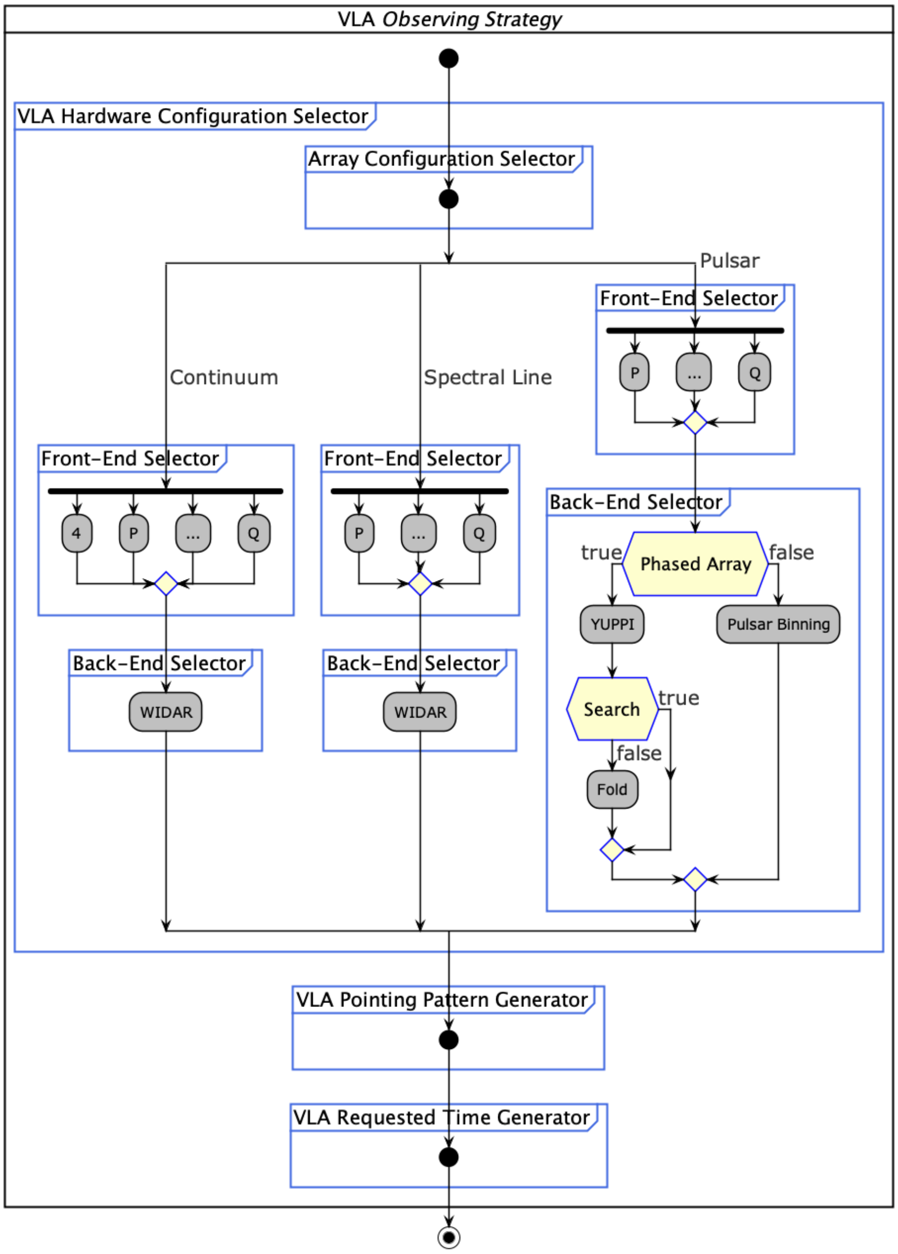



**Figure 7.1:** Diagram of algorithm to select the VLA Configuration, the FRONT-END, and the BACK-END.

### 7.1.1 VLA Configuration Selector

Under Construction.

### 7.1.2 VLA Continuum front-end Selector

Under Construction.

### 7.1.3 VLA Spectral Line front-end Selector

Under Construction.

### 7.1.4 VLA Pulsar front-end Selector

Under Construction.

### 7.1.5 VLA Continuum back-end Selector

Under Construction.

### 7.1.6 VLA Spectral Line back-end Selector

Under Construction.

### 7.1.7 VLA Pulsar back-end Selector

Under Construction.

## 7.2 VLA Pointing Pattern Generator

The VLA ***`Pointing Pattern Generator`*** selects a `Pointing Pattern` in response to the *Capability Request Parameters.* The VLA Pointing Patterns contain the following types:

$$\text{VLA Pointing Patterns}\begin{cases}\text{Single Pointing}\\ \\ \text{Mosaic}\begin{cases}\text{On-the-Fly}\\ \text{Discrete}\begin{cases}\text{Hexagonal}\end{cases}\end{cases}\end{cases}$$

For observers, useful descriptions of these pointing patterns and further documentation are available in Section 24. An observer is presumed to be familiar with the general behavior of the antennas associated with these terms and the implementation team would not require such details to proceed.

The following enumerated list outlines the steps needed to select a `Pointing Pattern`; these steps are diagrammed in Figure 7.2.

1. The condition for a `Single Pointing Pattern` is when the Primary Beam, $\theta_{\rm PB}$, is much greater than $\Omega_{\rm FOV}$:

$$\theta_{\rm PB} > \text{scalar} \times \ max(\Omega_{\rm FOV}), \tag{7.2.1}$$

where scalar = XX, and the Primary Beam is compared to the largest dimension of $\Omega_{\rm FOV}$. The Primary Beam of the telescope is

$$\theta_{\rm PB} = (1.02 + 0.0135\, T_e) \times \frac{c}{\nu} \times \frac{1}{\rm D_{dish}} \text{ rad}, \tag{7.2.2}$$

or

$$\theta_{\rm PB} = 1.25 \frac{c}{\nu} \times \frac{1}{\rm D_{dish}} \text{ rad}$$

where $c$ is the speed of light in m s$^{-1}$, $\nu = \nu_c + \Delta\nu$ is the upper frequency (Hz) of the requested bandwidth.

2. If Eq (7.2.1) is True, the `Single Pointing Pattern` is used (go to Section 7.2.1).
3. If False, then a mosaic pattern is used, which is either a discrete mosaic or an On-the-Fly (OTF) mapping.
   (a) The algorithm decides between OTF mapping and a discrete mosaic based on two conditions:
      i. *Overhead.* When the `settle time` of the telescope is comparable to the `effective requested time`, $t_{eff}$, the overhead becomes large or

$$t_{eff} < XX \text{s}. \tag{7.2.3}$$

      ii. *Data Rate.* The Data Rate for OTF mapping must be less than XX.
   (b) If either condition is False, a discrete mosaic pattern is used (go to Section 7.2.2).
   (c) If both conditions are True, OTF mapping is used (go to Section 7.2.3).

**Table 7.4:** Summary of Conditions for Figure 7.2

| Text in Figure 7.2 | Condition | Reference |
|---|---|---|
| (1) Condition for Single Pointing | $\theta_{\rm PB} >$ scalar $\times$ $max(\Omega_{\rm FOV})$ | §7.2 |
| (2) Condition for OTF | $t_{eff} <$ XX s AND Data Rate $<$ XX | §7.2 |
| (3) Validate OTF RA Spatial Extent | $\Omega_{\rm RA} >$ XX | §7.2.3 |
| (4) Validate `scan rate` | `scan rate` $< 3$ arcmin s$^{-1}$ | §7.2.3 |
| (5) Validate `Dump Time` | $t_{\rm dump} < 0.6$ s | §7.2.3 |

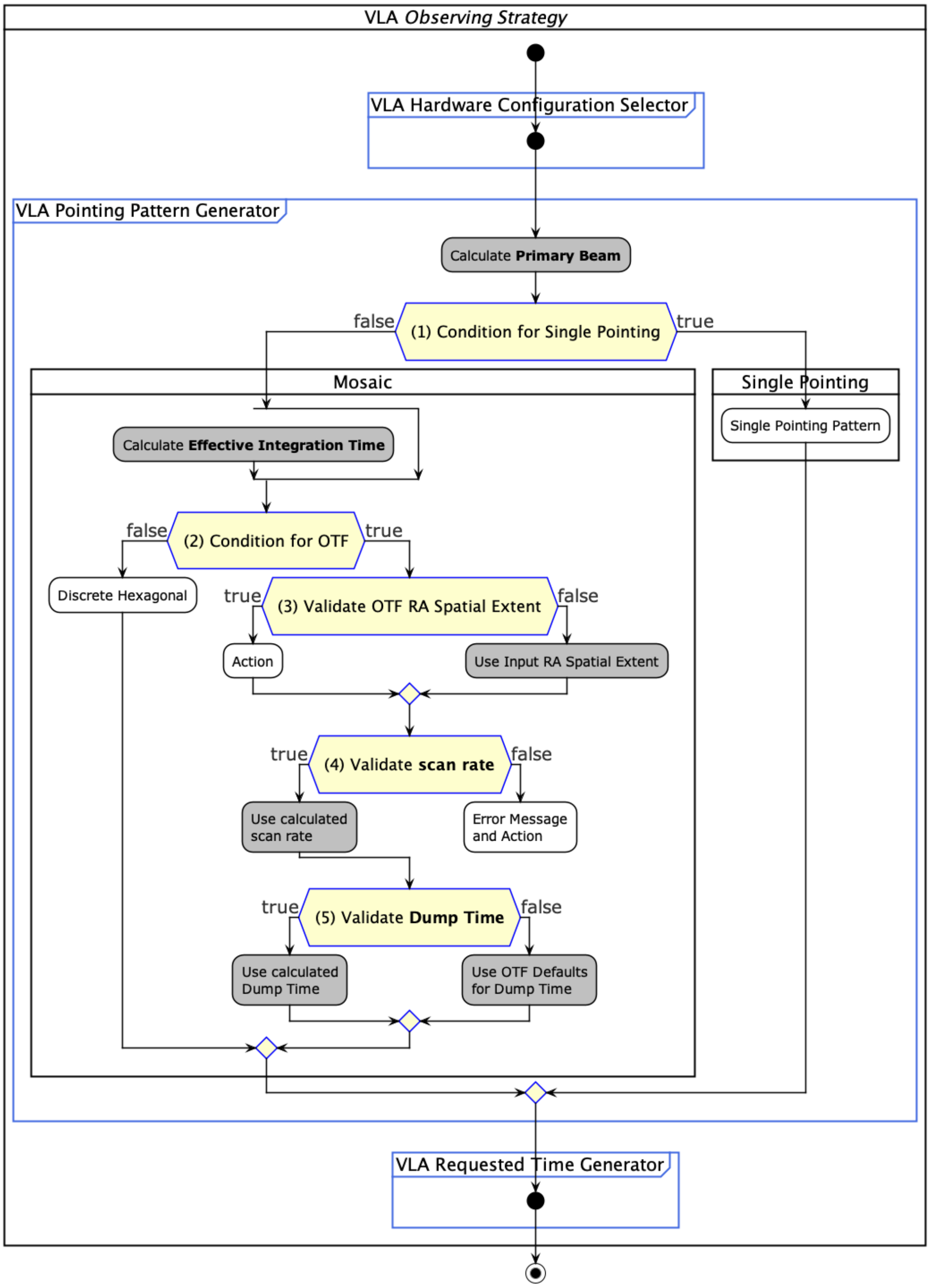



**Figure 7.2:** VLA *Observing Strategy*

### 7.2.1 VLA Single Pointing

The parameters needed to describe a VLA `Single Pointing Pattern` are as follows:

1. Position, which is equivalent to the FIELD SOURCE Position.
2. `Requested Time`

### 7.2.2 VLA Discrete Mosaic

The recommended discrete mosaic pattern for the VLA is a Hexagonal Pattern. The parameters needed to fully describe a Discrete Mosaic Pattern are the following:

1. `position per pointing` – For each pointing, this is a coordinate that tells an antenna where to point. The nominal position of the `Pointing Pattern` is equivalent to the FIELD SOURCE Position.

2. `requested time per pointing` – The time an antenna spends collecting data per pointing. This does not include overhead.

The Hexagonal Pattern is composed of a number of pointings along a line of constant Declination (called a row) to span the angular extent in RA ($\Omega_{\text{RA}}$). The angular extent in Dec ($\Omega_{\text{Dec}}$) is spanned by layering rows. The pattern, which includes the `position per pointing` and the `requested time per pointing`, is constructed with the following steps.

1. The center of the mosaic is equal to the Position in the FIELD SOURCE. Once specified, the `position per pointing` is calculated following the guidelines below.

   (a) The number of pointings needed to span the requested angular extent in RA ($\Omega_{\text{RA}}$) and that in Dec ($\Omega_{\text{Dec}}$) are

   $$\mathrm{n}_{\mathrm{ra}} = \frac{\Omega_{\mathrm{RA}}}{\theta_{hex}} \text{ and}$$

   $$\mathrm{n}_{\mathrm{dec}} = \frac{\Omega_{\mathrm{Dec}}}{\theta_{row}},$$

   respectively, rounded up to the nearest integer. $\theta_{\text{hex}}$ and $\theta_{\text{row}}$ are defined as

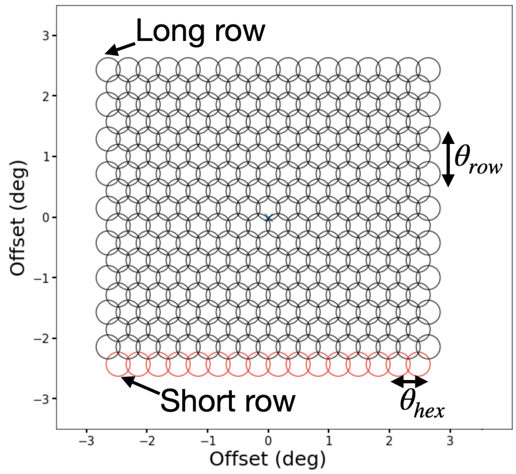



**Figure 7.3:** Example of a VLA Hexagonal Pointing Pattern. The variables, terms, and circles in red are described in the text.

   i. The angular spacing along the rows of the mosaic is

   $$\theta_{\text{hex}} = \frac{\theta_{\text{PB}}}{\text{scalar}},$$

   where

   $$\text{scalar} = \begin{cases} 1.2 \text{ (Nyquist)} \\ \sqrt{2} \\ \sqrt{3}, \end{cases}$$

   and the upper frequency ($\nu_c + \Delta\nu/2$) of the requested bandwidth is used to calculate the Primary Beam, $\theta_{\text{PB}}$. The algorithm uses scalar = XX by default.

   ii. The angular spacing between the rows of the mosaic

   $$\theta_{\text{row}} = \left(\frac{3}{2}\right)\theta_{\text{hex}}.$$

   (b) The pattern is constructed by alternating rows of different lengths, called short rows and long rows. The rows are offset in Declination by $\theta_{row}$. An example of a short row is highlighted in red in Figure 7.3. The offset rows follow these requirements:

   i. Long rows consist of $\mathrm{n}_{\mathrm{ra}}$+1 pointings, each pointing offset in RA by $\theta_{hex}$.

   ii. Short rows consist of $\mathrm{n}_{\mathrm{ra}}$ pointings, each pointing offset in RA by $\theta_{hex}$.

iii. The long and short rows are offset from each other in RA by $\pm\frac{1}{2}\theta_{hex}$ to stagger position of each pointing between the rows, which creates the hexagonal pattern.

2. Each pointing will have the same integration time, so the `requested time per pointing` is calculated as follows.

   (a) Use the VLA Sensitivity Calculator (§21.2.1) to determine the `effective requested time`, $t_{eff}$.

   (b) The `requested time per pointing` is

   $$\texttt{requested time per pointing} = \frac{t_{eff}}{n_{tot}},$$

   where $n_{tot}$ is the `total number of pointings` in the mosaic given by

   $$n_{tot} = n_{\rm dec} \ \times \ (n_{\rm ra} + \frac{1}{2}),$$

   rounded up to the nearest integer.

3. The additional parameters that may be reported on request are

   (a) Mosaic Beam Area:

   $$\Omega_{beam} = 0.5665\ \theta^2_{\rm PB} \tag{7.2.4}$$

   (b) `Requested Time`:

   $$\texttt{Requested Time} = n_{tot} \ \times \ \texttt{requested time per pointing} \tag{7.2.5}$$

   (c) Survey Speed:

   $$\text{Survey Speed} = \frac{\Omega_{beam}}{t_{eff}} \tag{7.2.6}$$

4. Scheduling Notes:

   (a) The observing order is established by the *Observation Planner Phase 3*.

### 7.2.3 VLA OTF

An On-the-Fly mosaic scans the sky in rows (stripes) along a line of constant Declination. The telescope scans continuously along a row (e.g., east-west) and then scans the opposite direction in the next row, which is offset in Declination from the preceding row. To fully describe an OTF pattern, the following parameters are needed:

1. Positions of the first and last pointings of each row.

2. `time per row` – This is the time interval an antenna collects data for a row; this does not include overhead.

The pattern is constructed with the following steps:

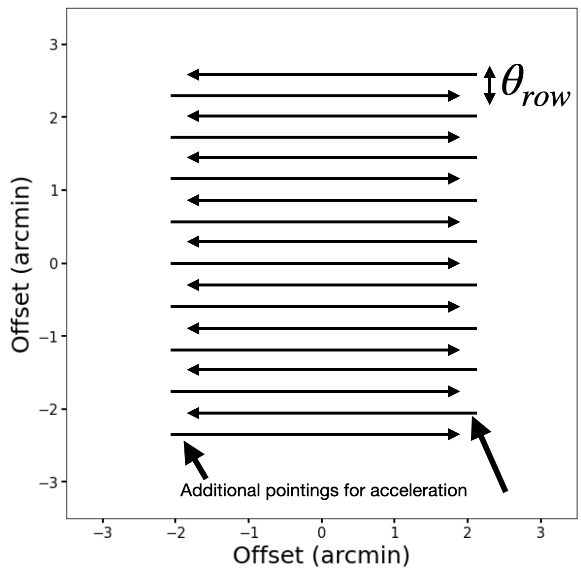



**Figure 7.4:** Example of VLA OTF mapping for a small $\Omega_{ra}$. The offset in Dec is equal to $\theta_{row}$.

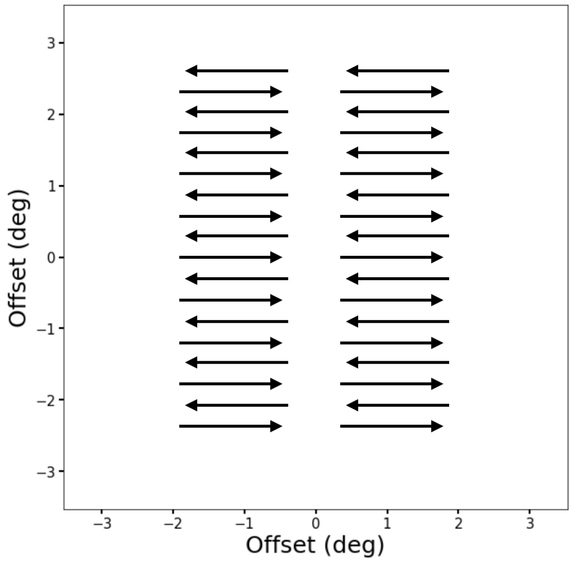



**Figure 7.5:** Example of VLA OTF mapping for a large $\Omega_{ra}$. The offset in RA between the tracks is artificially inflated. The offset in Dec is between rows is equal to $\theta_{row}$.

1. If the angular extent in RA, $\Omega_{\mathrm{ra}}$, is too large, the length of the rows is reduced to maintain flexibility for dynamic scheduling due to elevation concerns. To facilitate the requested coverage by $\Omega_{\mathrm{ra}}$, multiple OTF patterns are constructed instead of a single OTF pattern. Figures 7.4 and 7.5 show examples of OTF pointings for a small and large angular extent, respectively. The condition for splitting up the OTF pattern is when

$$\Omega_{\mathrm{RA}} \;>\; XX, \tag{7.2.7}$$

the algorithm constructs $N$ contiguous OTF sub-patterns with a width of $\Delta_{RA} \sim$ XX, where

$$\Omega_{\mathrm{RA}} = \sum_{i=1}^{N} \Delta_{\mathrm{RA,i}}.$$

   (a) If this condition is met, $\Delta_{\mathrm{RA}}$ should be used in place of $\Omega_{\mathrm{RA}}$ in Steps 2f, 3a, and 3b.

2. The `time per row` is the time the antenna spends collecting data along a row. It does not include the slew and `settle time`. It is calculated in the steps below.

   (a) The `scan rate` is how quickly an antenna will slew across the sky and is given by

$$scan\ rate = \frac{[0.5665\ \theta_{\mathrm{PB}}^2]}{t_{eff} \times \theta_{\mathrm{row}}}. \tag{7.2.8}$$

   The quantity in the square brackets is the beam area, which uses the upper frequency of the requested bandwidth ($\nu_c + \Delta\nu/2$) to calculate Primary Beam, $\theta_{\mathrm{PB}}$. The `effective requested time` is $t_{eff}$ (see Section 21.2.1), and $\theta_{\mathrm{row}}$ is the angular spacing between rows (stripes):

$$\theta_{\mathrm{row}} = \frac{\theta_{\mathrm{PB}}}{\mathrm{scalar}},\ \text{where}$$

$$\mathrm{scalar} = \begin{cases} \sqrt{2} \\ \sqrt{3} \\ 4, \end{cases}$$

   where the default value used by the algorithm is XX.

(b) The algorithm performs a check on the value of the `scan rate`.

i. If the `scan rate` is greater than 3 ? 10 arcmin $s^{-1}$, the algorithm will provide a message to the user prompting an Action.

ii. E.g., The message instructs the user to change the requested `RMS Sensitivity`, the frequency, or bandwidth OR defaults to the value of XX. Figure 7.6 illustrates the relationship between $t_{eff}$, $\theta_{\rm PB}$, and the `scan rate`.

(c) In OTF mapping, the integration time is inherently fast and will effectively function as a `Dump Time` ($t_{dump}$), which is the time interval in which the data is processed by the back-end processing cluster. To prevent beam smearing, at least 10 integrations are needed as the antenna scans a distance equal to the FWHM of the Primary Beam. The calculated `Dump Time`, $t_{\rm dump}$, for OTF mapping is

$$t_{\rm dump} \sim 0.1 \times \frac{\theta_{\rm PB}}{scan\ rate}. \tag{7.2.9}$$

(d) The algorithm performs a check of value of $t_{\rm dump}$, as the phasecenter cannot change faster than 0.6 ? 0.5s.

i. If $t_{dump} < 1$s for 8-bit observing, the default `Dump Time`, $t_{dump,default}$, of 1s is used.

ii. If $t_{dump} < 4$s for 3-bit observing, $t_{dump,default} = 4$s.

(e) If the defaults are used for $t_{dump}$, the Number Of Integrations Per Phasecenter, $n_{integ}$, is

$$n_{integ} = \frac{t_{dump,default}}{t_{\rm dump}}, \tag{7.2.10}$$

otherwise, $n_{integ} = 1$.

(f) The number of pointings in a row (stripe) is

$$n_{\rm stripe} = \frac{\Omega_{ra}}{[scan\ rate\ \times\ n_{integ} \times t_{dump}]} \tag{7.2.11}$$

where the quantity in the square brackets is the angular distance between two phasecenters, $\theta_{point}$.

(g) The `time per row` is then

$$\text{Row Duration} = (n_{stripe} + 1) \times n_{integ}\ \times\ t_{dump}. \tag{7.2.12}$$

The addition of 1 to $n_{stripe}$ is to allow the antenna(s) time to accelerate.

3. The center of the OTF pattern is equal to the Position specified by the FIELD SOURCE. The starting row is the southern most one (Row 0), and is observed east to west. There are $n_{rows}$ in the pattern, so the last row is number Row $n_{rows}$ - 1. Additionally, an extra phasecenter is added to the start of each row to allow for the telescope to accelerate.

(a) For Row 0, the Positions of the first and last pointings of the row are given by

$$(\alpha_{start,0}, \delta_{start,0}) = (\alpha_{position} + \frac{\Omega_{\rm RA}}{2} + \theta_{point},\ \delta_{position} - \frac{\Omega_{\rm Dec}}{2})$$

and

$$(\alpha_{stop,0}, \delta_{stop,0}) = (\alpha_{position} - \frac{\Omega_{\rm RA}}{2}, \delta_{start,0}).$$

(b) Then, the odd rows, which are observed west-to-east, have positions of

$$(\alpha_{start,n}, \delta_{start,n}) = (\alpha_{position} - \frac{\Omega_{\rm RA}}{2} - \theta_{point},\ \delta_{start,n-1} + \theta_{row})$$

and

$$(\alpha_{stop,n}, \delta_{stop,n}) = (\alpha_{position} - \frac{\Omega_{\rm RA}}{2},\ \delta_{start,n})$$

and the even numbered rows, which are observed east-to-west, have start and stop positions of

$$(\alpha_{start,n}, \delta_{start,n}) = (\alpha_{position} + \frac{\Omega_{\rm RA}}{2} + \theta_{point},\ \delta_{position} - \frac{\Omega_{\rm Dec}}{2} + \theta_{row})$$

and

$$(\alpha_{stop,n}, \delta_{stop,n}) = (\alpha_{position} - \frac{\Omega_{\rm RA}}{2},\ \delta_{start,n})$$

until $n = n_{rows}$-1, where

$$n_{rows} = \frac{\Omega_{\rm Dec}}{\theta_{row}}$$

and is the number of rows required to span $\Omega_{\rm Dec}$.

4. There are additional parameters that may be reported for validation purposes or may need to be accessible to other portions of the algorithm.

   (a) Beam Area: Equation 7.2.4

   (b) Survey Speed: Equation 7.2.6

   (c) If Condition 7.2.7 is True, the number of OTF sub-patterns, $N$, and the extent in RA of each sub-pattern, $\Delta_{\rm RA}$.

   (d) Scan Rate: Equation 7.2.8

   (e) `Dump Time` + Validation Check: Equation 7.2.9

   (f) Number of Integrations Per Step: Equation 7.2.10

   (g) `Requested Time`, which does not include overhead:

$$\texttt{Requested Time} = n_{rows} \times \texttt{time per row} \tag{7.2.13}$$

5. OTF Scheduling Notes:

   (a) Scans from east to west move with the sidereal motion while *Scans* from west to east are counter-sidereal. Therefore, for the same on-the-sky angular scan rate, the east-to-west scans will require faster telescope motion. Additionally, observing near the Zenith where the azimuthal rate becomes very high should be avoided.

   (b) For OTF targets close to 34 deg Dec, map well before or well after transit, i.e. close to rise or set. https://science.nrao.edu/facilities/vla/docs/manuals/opt-manual/observation-preparation-tool/scan-modes-obs-modes/on-the-fly-mosaicking

   (c) The `slew` and Settle times needs to be accounted for.

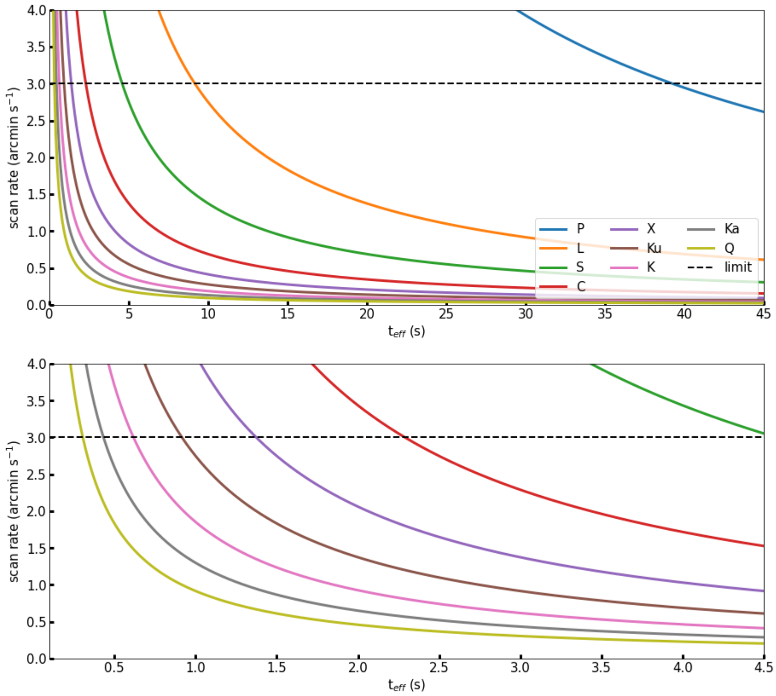



**Figure 7.6:** *Top.* The `scan rate` as a function of different Effective Integration Times and VLA frequency bands, as represented by the different colored curves. *Bottom.* Same as the top plot except the range of the abscissa is truncated to provide better detail.

# 7.3 VLA Requested Time Generator

`RMS Sensitivity` is not the only concern for time on source. Need to fold in other toggles if applicable. This may not be the best place to do so. Can the `Requested Time` from the `Pointing Pattern` be updated here?

1. Look in the System Description for information about this
   (a) uv-coverage for an interferometer
   (b) event occurrence rate
   (c) sensitivity
   (d) "etc"
2. Does there need to be a performance parameters for Dynamic Range or Parallactic Angle?

# 8 GBT Observing Strategy

For each row in the *Science Target List*, the *Observing Strategy* implements a series of algorithms to specify the SOURCE, HARDWARE CONFIGURATION, and `Requested Time` of the prototype Science Target. The GBT *Observing Strategy* calls the GBT ***`Hardware Configuration Selector`***, which determines the FRONT-END and BACK-END. Next, it calls the GBT ***`Pointing Pattern Generator`***, which specifies the SOURCE and then it calls the ***`Requested Time Generator`***. The steps outlined here are shown in Figure 6.1. The sections in this chapter provide the details of these algorithms for GBT *Capabilities*. Table 8.2 lists relevant sections for the *Observing Strategy* algorithms, and Table 8.3 presents the same information as Table 8.2 but organized by Observing Type. As an example, if GBT Continuum *Capability* is selected, the *Observing Strategy* uses the algorithms referenced in column 2 of Table 8.3 to construct the *Science Target List*. Whereas, a GBT Pulsar Capability follows column 4.

**Table 8.1:** GBT Receivers and BACK-ENDS: capabilityConfig.json

here's a test table

**Table 8.2:** Overview of GBT Observing Strategy.

| Algorithm | § |
|---|---|
| GBT Hardware Configuration Selector | |
| GBT FRONT-END Selector | |
| GBT Continuum | 8.1.1 |
| GBT Spectral Line | 8.1.2 |
| GBT Pulsar | 8.1.3 |
| GBT Radar | 8.1.4 |
| GBT BACK-END Selector | |
| GBT Continuum | 8.1.5 |
| GBT Spectral Line | 8.1.6 |
| GBT Pulsar | 8.1.7 |
| GBT Radar | 8.1.8 |
| GBT ***`Pointing Pattern Generator`*** | 8.2 |
| GBT ***`Requested Time Generator`*** | 8.3 |

**Table 8.3:** GBT *Observing Strategy* Algorithms by *Observing Types*

| Algorithm | GBT Continuum § | GBT Spectral Line § | GBT Pulsar § | GBT Radar § |
|---|---|---|---|---|
| GBT FRONT-END Selector | 8.1.1 | 8.1.1 | 8.1.3 | 8.1.4 |
| GBT BACK-END Selector | 8.1.5 | 8.1.6 | 8.1.7 | 8.1.8 |
| GBT ***`Pointing Pattern Generator`*** | 8.2 | 8.2 | 8.2 | 8.2 |
| GBT ***`Requested Time Generator`*** | 8.3 | 8.3 | 8.3 | 8.3 |

## 8.1 GBT Hardware Configuration Selector

An observing type is decided upstream; the available observing types are continuum, spectral line, pulsar, or radar. The algorithm then flows according to the logic shown in Figure 8.1.

### 8.1.1 GBT Continuum front-end Selector

Under Construction.

### 8.1.2 GBT Spectral Line front-end Selector

Under Construction.

### 8.1.3 GBT Pulsar front-end Selector

Under Construction.

### 8.1.4 GBT Radar front-end Selector

Under Construction.

### 8.1.5 GBT Continuum back-end Selector

Under Construction.

### 8.1.6 GBT Spectral Line back-end Selector

Under Construction.

### 8.1.7 GBT Pulsar back-end Selector

Under Construction.

### 8.1.8 GBT Radar back-end Selector

Under Construction.

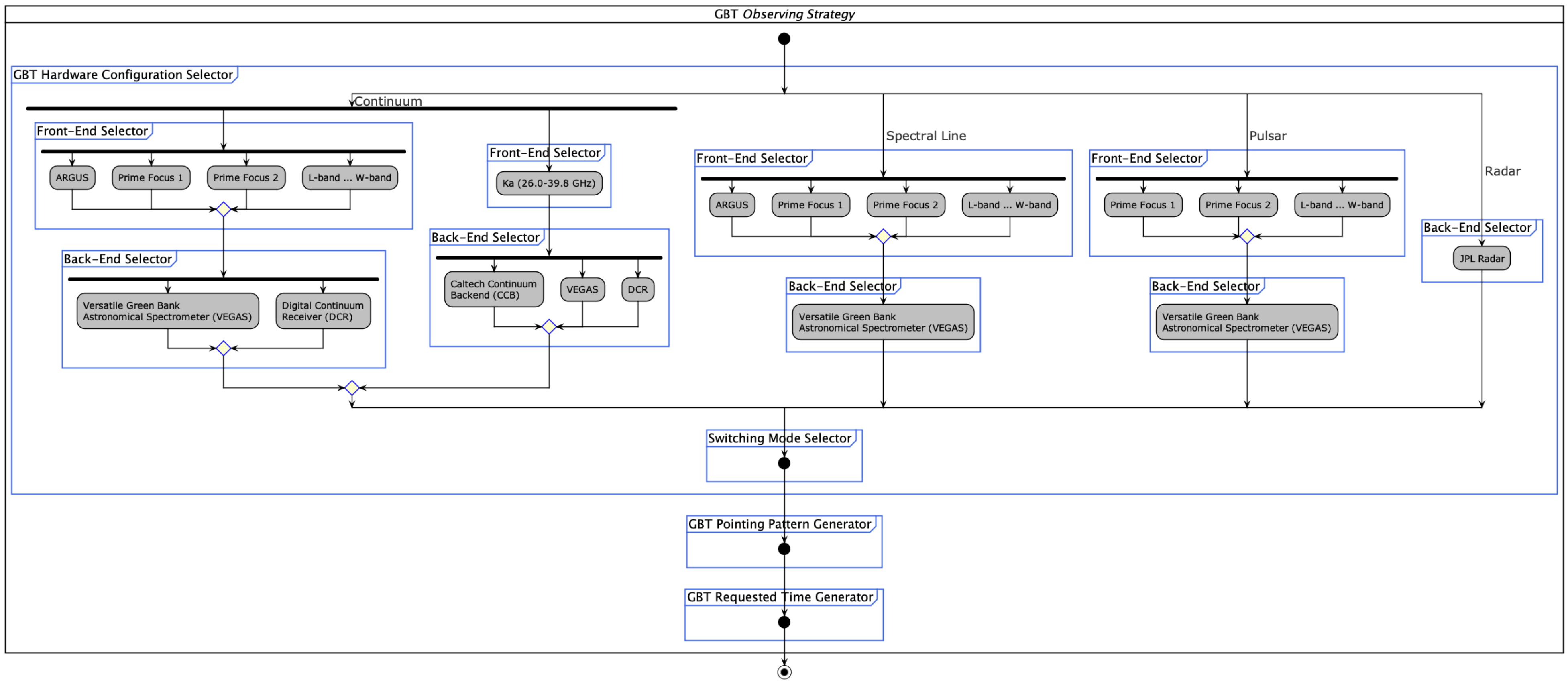



**Figure 8.1:** Diagram of algorithm to select the FRONT-END for the GBT and the BACK-END.

## 8.2 Pointing Patterns for the GBT

The standard `Pointing Patterns`[1] available to the algorithm for the GBT include the following

$$\text{GBT Pointing Patterns}\begin{cases}\text{Single Pointing} \\ \text{Map}\begin{cases}\text{OTF}\begin{cases}\text{RAlongmap} \\ \text{RAlongmapWithReference} \\ \text{Declatmap} \\ \text{DeclatmapWithReference} \\ \text{Daisy}\end{cases} \\ \text{Discrete}\begin{cases}\text{PointMap} \\ \text{PointMapWithReference}\end{cases}\end{cases}\end{cases}$$

For observers, useful descriptions of these pointing patterns and further documentation are available in Section 24. An observer is presumed to be familiar with the general behavior of the antennas associated with these terms and the implementation team would not require such detail to proceed.

The condition for Single Pointing is when the Primary Beam[2], $\theta_{\mathrm{PB}}$, is much greater than the `Field of View`. The Primary Beam is compared to the largest dimension of $\Omega_{\mathrm{FOV}}$.

$$\theta_{\mathrm{PB}} > \mathrm{scalar} \times \; max(\Omega_{\mathrm{FOV}}), \tag{8.2.1}$$

where scalar = XX. The Primary Beam, or Half Power Beam Width (HPBW), of the GBT is

$$\theta_{\mathrm{PB}} = (1.02 + 0.0135\, T_e) \times \frac{c}{\nu} \times \frac{1}{\mathrm{D_{dish}}} \text{ rad},$$

or

$$\theta_{\mathrm{PB}} = 1.25 \frac{c}{\nu} \times \frac{1}{\mathrm{D_{dish}}} \text{ rad}$$

where $c$ is the speed of light in m s$^{-1}$, $\nu = \nu_c + \Delta\nu$ is the highest frequency in Hz. The remaining variables are defined in Chapter 8.

If condition (8.2.1) is not met, i.e., the Primary Beam is smaller than the `Field of View`, then a Map pattern is considered: a Discrete map or OTF mapping. The algorithm decides between OTF mapping and a Discrete map based on two conditions.

[1] The GBT Observing Guide (GBTog) calls these Scan Types

[2] This is the Half Power Beam Width (HPBW).

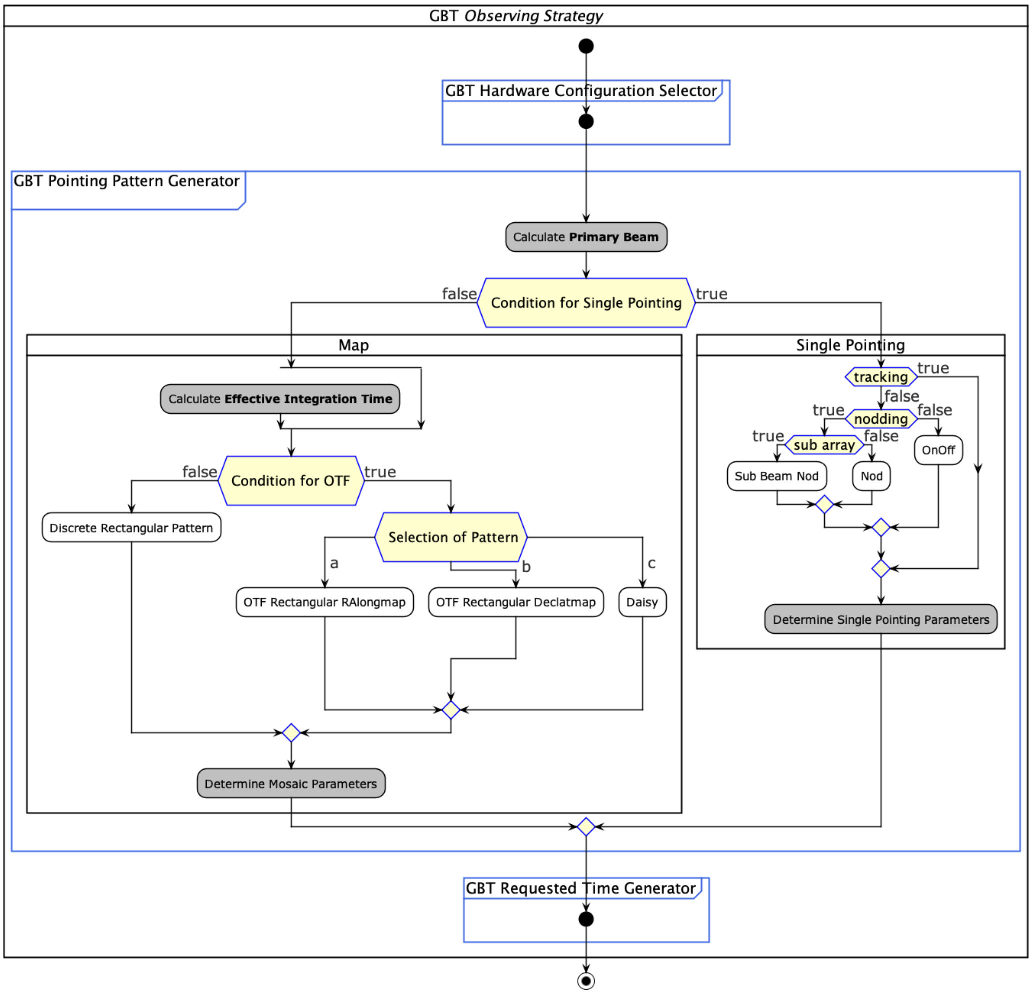



**Figure 8.2:** GBT *Observing Strategy*

| Condition for Figure 8.2 | | Section |
|---|---|---|
| Condition for Single Pointing | $\theta_{\mathrm{PB}} > \mathrm{scalar} \times \ max(\Omega_{\mathrm{FOV}})$ | 8.2 |
| Condition for OTF | ??? | 8.2 |
| Selection of Discrete Pattern | ??? | 8.2.2 |
| Tracking | ??? | 8.2.1 |
| Nodding | ??? | 8.2.1 |
| Sub array | ??? | 8.2.1 |

### 8.2.1 GBT Single Pointing

High Frequency receivers have two beams. If the source is not extended and doing position switched observations, use Astrid Nod() procedure

### 8.2.2 GBT Discrete Map

Under Construction.

### 8.2.3 GBT OTF

Under Construction.

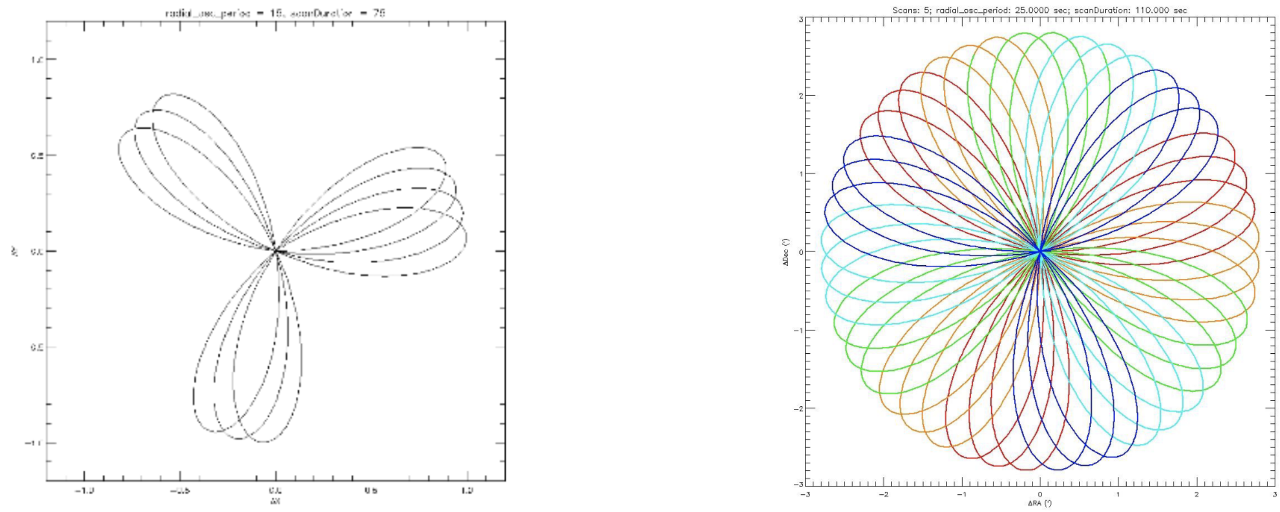
#### 8.2.3.1 GBT RAlongmap and Declatmap

Under Construction.

#### 8.2.3.2 GBT Daisy Map

A Daisy map scans continuously around a central point as shown in Figure 8.3. This pattern is discussed in detail in Section 6.4.3.7 of *Observing with the Green Bank Telescope*[3].

[3] https://science.nrao.edu/facilities/gbt/observing/GBTog.pdf




**(a)** **(b)**

**Figure 8.3:** GBT Daisy maps at 5 radial oscillations (a) and 22 radial oscillations (b), which is a closed pattern.

## 8.3 GBT Requested Time

1. Spectral lines towards strong continuum sources need careful configuration.
2. weak broad spectral lines (wider than $\sim$ 100 MHz) towards a strong continuum emission (more than 1/10th the system temperature), then need to consider double position switching.
3. need to find proper IF balance – BalanceOnOff() in section 6.4.1

# 9 Overview of Calibration and Scheduling Strategies

*Calibration* and *Scheduling Strategies* are *Capability* specific factories that produce instructions for the *Observation Planner*. These instructions are fundamentally prescriptions for calibrating and scheduling Science targets to produce *Observation Specifications* that reflect the user's fundamental science request and adhere to best practices for observing. These factories belong to the *Science Target List* but they do not produce the instructions until called in the *Observation Planner*.

A *Calibration Strategy* contains Partition and Calibration Plans. A Partition Plan consists of instructions for partitioning the *Science Target List*, called Partition Instructions, and a Calibration Plan consists of Observing Instructions which are instructions on how to calibrate the Science Targets. All *Capabilities* require at least one Partition Plan and one Calibration Plan; the composition of a plan is *Capability* specific. There are similarities between the *Capabilities*, but each Observing Type (e.g., continuum, spectral, pulsar) is considered independently.

A *Scheduling Strategy* contains one or more Scheduling Plan. This is the prescription for creating a *Scan List*, which is the order in which the Observing Targets are to be observed.

Chapters 10 – 15 detail of the strategies available to the algorithm per *Capability*, as *Calibration* and *Scheduling Strategies* are dependent on the *Capability* (see Table 9.1). The implementation details of the Partition Instructions and Observing Instructions are in Chapters 17 and 18.

**Table 9.1:** Overview of Calibration and Scheduling Strategies.

| Capability | Sections for Calibration and Scheduling Strategies |
|---|---|
| GBT Continuum | 10.1, 13.1 |
| GBT Spectral Line | 10.2, 13.2 |
| GBT Pulsar | 10.3, 13.3 |
| GBT Radar | 10.4, 13.4 |
| VLA Continuum | 11.1, 14.1 |
| VLA Spectral Line | 11.2, 14.2 |
| VLA Pulsar | 11.3, 14.3 |
| VLBA Continuum | 12.1, 15.1 |
| VLBA Spectral Line | 12.2, 15.2 |
| VLBA Pulsar | 12.3, 15.3 |

## 9.1 Overview of Partition Plans and Partition Instructions

Because each row of the *Science Target List* includes a *Calibration Strategy*, a Partition Plan is built per Science Target but the details of the Partition Instructions can depend on a holistic assessment of a subset or the entirety of the *Science Target List*. The Partition Plan is meant to be a set of rules that can dynamically adjust to the state of partitioning and the Science Targets.

There are many Partition Instructions available to a *Capability*. Some Partition Instructions can be *Facility* specific, for example, an Array Configuration PI is only available to the VLA. Other PIs may be available to all *Facilities*, but the details of their behavior is *Capability* specific. There are two types of PIs: Hierarchical Partition Instructions and Dynamical Partition Instructions, which have distinct algorithmic implementations.

### 9.1.1 Hierarchical Partition Instructions

Hierarchical PIs set the criteria for evaluating the dissimilarity between a pair of Science Targets. A Hierarchical PI will define a custom metric, which can be a comparison operator, a mathematical expression, or set of if-then statements. The *Capability* sets the criteria for segmentation, and the Hierarchical PI translates into a numerical value, called a threshold, for the *Observation Planner* algorithm to implement.

The following sections provide a general overview of the intent, motivation, constraints, and requirements of the Hierarchical PIs. The *Capability* specific treatment of these PIs are discussed in later chapters, which are referenced in the sections below. The algorithmic implementation details are given in Chapter 17.1.1.

#### 9.1.1.1 Array Configuration Partition Instruction

The Array Configuration Partition Instruction specifies the threshold and custom metric to divide the *Science Target List* to create groups that contain one unique VLA Configuration. This is exclusive to VLA *Capabilities*, which have HARDWARE CONFIGURATIONS that consist of a FRONT-END, BACK-END, and a Array Configuration. One unique Array Configuration is permitted per *Observation Specification*. This PI does not have *Capability* specific constraints on its behavior, meaning it behaves the same for any VLA *Capability*:

- The threshold for this PI is 0.5;
- The custom metric is a comparison operator that compares the Array Configuration of any pair of Science Targets in a specific subset. The value the custom metric returns is 0 if the pair of Array Configurations are equal and 1 if they are not.

#### 9.1.1.2 Array Subset Partition Instruction

The Array Subset Partition Instruction specifies the threshold and custom metric to divide the *Science Target List* to create groups that are defined by the Array Subsets in their HARDWARE CONFIGURATIONS. This is exclusive to VLBA *Capabilities*, which have HARDWARE CONFIGURATIONS that consist of a FRONT-END, BACK-END, and an Array Subset. One or more Array Subsets are permitted in a group if they have 50% of the stations in common. This PI does not have *Capability* specific constraints on its behavior, meaning it behaves the same for any VLBA *Capability*:

- The threshold for this PI is 0.5;
- The custom metric is a comparison operator that divides Science Targets into groups based on a shared Array Subset. The value the custom metric returns is 0 if the pair of the Array Subsets are equal and 1 if they are not.

#### 9.1.1.3 Calibration Parameter Partition Instruction

The Calibration Parameter Partition Instruction specifies the threshold and custom metric to divide the *Science Target List* by CALIBRATION PARAMETER. The CALIBRATION PARAMETERS are specified in the *Capability Request* as either True or False, and all the FIELD SOURCES that are associated with the *Capability Request* will share those values of the CAPABILITY REQUEST PARAMETERS. Recall that for any *Allocation Request*, multiple *Capability Requests* can exist; however, only one *Science Target List* is associated with an *Allocation Request*. It is possible that a *Science Target List* has mix of values for the CALIBRATION PARAMETERS. Only one unique value (e.g., True or False) of this CALIBRATION PARAMETER is permitted in a group, as it has significant consequences how the Science Targets are calibrated. This PI has *Capability* specific constraints on its behavior, as different *Capabilities* may have different CALIBRATION PARAMETERS. The *Capability* sets

- the threshold;
- the custom metric, which evaluates pairs of Science Targets and a subset, or the entirety, of the *Science Target List* using a set of if-then rules. The value the custom metric returns will range between 0 and 1.

#### 9.1.1.4 Distance Partition Instruction

The Distance Partition Instruction specifies the threshold and custom metric to divide the *Science Target List* with respect to angular separation between a group of Science Targets. It may be efficient in both observing and scheduling considerations to have relatively compact associations of Science Targets. The angular extent of a cluster of sources can also be important to the calibration of the data.

- The *Capability* sets the threshold, which may also depend on the `scheduling priority` and the HARDWARE CONFIGURATION of the Science Targets.
- The custom metric, which calculates and returns the angular separation between a pair of Science Targets, is the same for any *Capability*. It is the Vincenty Formula[1], which is, at the time of writing this document, the formula used by the Python package Astropy[2] version 5.1[3] to perform a similar calculation.

The *Capability* specific details are given in Chapters 10, 11, and 12.

#### 9.1.1.5 Frequency Partition Instruction

The Frequency Partition Instruction specifies the threshold and custom metric to divide the *Science Target List* with respect to the requested FRONT- and BACK-ENDS of the Science Targets. It is possible, and common, to have multifrequency observations in a single *Observation Specification* but the best practices associated with a *Capability* may restrict which frequencies are grouped together. For example, high frequency observations are rarely scheduled with lower frequency observations because the weather constraints are considerable in the former. In addition to specifying which frequency bands can be grouped, it is necessary to specify the prioritization of this grouping. For example, a Ka-band (33 GHz) Science Target could be allowed in a group with an X-band (9 GHz) Science Target, but if there is also a Science Target requesting C-band (5 GHz), the priority may be to group together the X- and C-band Science Targets and let the Ka-band Science Target group separately. The *Capability* specifies

- the threshold;
- the custom metric, which evaluates pairs of Science Targets and a subset, or the entirety, of the *Science Target List* using sets of comparison operators. The value the custom metric returns will range between 0 and 1.

#### 9.1.1.6 Priority Partition Instruction

The Priority Partition Instruction specifies the threshold and custom metric to divide the *Science Target List* by `scheduling priority` to create groups with one, unique `scheduling priority`. Only one unique `scheduling priority` is permitted per *Observation Specification*. Unless specified otherwise by the *Observing Strategy*, the algorithm assumes a `scheduling priority` of A-rank to produce clusters

[1] https://en.wikipedia.org/wiki/Great-circle_distance
[2] http://www.astropy.org
[3] https://docs.astropy.org/en/v5.1/api/astropy.coordinates.SkyCoord.html

of Science Targets that best balance the observing practices of the *Facility* and the user intent reflected in the *Capability Request*. This PI does not have *Capability* specific constraints on its behavior, meaning it behaves the same for any *Capability* and thus any *Facility*:

- The threshold is set to 0.5.
- The custom metric is a comparison operator that divides Science Targets into groups based on a common `scheduling priority`. The value the custom metric returns is 0 if the pair of Science Targets have equivalent *scheduling priorities* and 1 if they do not.

### 9.1.2 Dynamical Partition Instructions

For a particular partitioning goal, Dynamical Partition Instructions provide the *Capability* specific criteria that reflect recommended observing practices, *Capability* specific concerns, and *Specification Constraints*. The subsections here define the conditions the algorithm needs for partitioning. Chapters 10, 11, and 12 give the *Capability* specific values of these parameters, and the *Observation Planner Phase 1* (Chapter 17) provides the implementation of them.

#### 9.1.2.1 Calibrator Partition Instructions

Under Construction.

#### 9.1.2.2 Hour Angle Partition Instruction

The Hour Angle PI defines the `observing window` for one or a group of Science Targets. The `observing window` is the time when

- the altitude of a SOURCE is above the `minimum elevation` and
- the altitude of a SOURCE is below the `maximum elevation`.

The PI defines these *Capability* specific criteria:

- the `minimum elevation` that an antenna may operate above, which could be determined by the *Specification Constraint* or the *Capability*;
- the `maximum elevation` that an antenna may operate below, which could be determined by the *Specification Constraint* or the *Capability*;
- the nominal location of the *Facility*, which can be many locations if, for example, the *Facility* is the VLBA.

With these parameters, the Hour Angle PI also dictates the algorithmic treatment of the `observing window`.

#### 9.1.2.3 Time Partition Instruction

A Time Partition Instruction provides the specifications for the *Observing Planner Phase 1* to partition Science Targets with respect to time. For a subset of Science Targets, the partitioning is constrained by

1. the `minimum elevation` that an antenna may operate above, which could be determined by the *Specification Constraint* or the *Capability*;
2. the `maximum elevation` that an antenna may operate below, which could be determined by the *Specification Constraint* or the *Capability*;

3. the nominal location of the *Facility*, which can be many locations if, for example, the *Facility* is the VLBA;

4. the `minimum duration` of a *Subscan*, which is a *Specification Constraint*;

5. a `minimum duration` per `Repeat Count`, which prioritizes groups whose constituents have longer `Partition Requested Time` over groups with large Repeat Counts and small `Partition Requested Time` per Science Target. This parameter functions to optimize the clustering; it is not a *Specification Constraint*, and it is not fundamentally motivated but chosen to address the tendency of the algorithm to make groups with larger than optimal Repeat Counts. A discussion on this parameter is available in Section 22.2.

6. The `maximum duration` per `Repeat Count` is the maximum allowed length of time of any single execution of an *Observation Specification*, excluding all overhead associated with calibration (e.g., Phase Referencing, Flux Density, Pointing, etc). While the *Capability* sets this criterion, the `scheduling priority` may be a factor. This criterion is necessary to accommodate the dynamic scheduling of the *Facilities* and to create clusters that the span reasonable time frames.

7. The total `Setup Time` associated with a cluster should be less than or equal to a fraction of the `maximum duration` per `Repeat Count`. The *Capability* sets the value of this criterion. This criterion is necessary to prioritize clusters that are not overhead dominated. This parameter functions to optimize the clustering; it is not a *Specification Constraint*, and it is not fundamentally motivated but chosen to address the tendency of the algorithm to make overhead dominated clusters. A discussion on this parameter is available in Section 22.2.

## 9.2 Overview of Calibration Plans and Observing Instructions

The *Calibration Strategy* factory produces the appropriate Calibration Plan for a group of Science Targets; this group can include all of the Science Targets in the *Science Target List* or a subset. The implementation of the Calibration Plan depends only on the Observing Instructions it contains. There are many Observing Instructions available to a *Capability*.

It is necessary to introduce concepts and terms before defining the components of a Calibration Plan, and Figure 9.1 illustrates the hierarchy of the time related concepts defined here. Note unless explicitly stated, these definitions are limited to the construction and implementation of Calibration Plans. Due to the limitations of the English language, there are shared words with different functional definitions spread throughout this document despite best efforts to be considerate of this exact situation.

**Calibration Plan Definitions and Concepts**

- `Acquisition Time` is the time an antenna spends taking data in a *Subscan*.
- `antenna slew time` is the time is takes for an antenna to move on the sky between two positions.
- `Hardware Configuration Overhead` is the time needed for hardware changes (e.g., changing receivers).
- `settle time` is the time an antenna needs to stabilize? after it has moved.
- `Setup Time` (§ 21.1) is the sum of the `antenna slew time` + `settle time` + `Hardware Configuration Overhead`.
- *Subscan* is the specification of the shortest, contiguous block of time over which an antenna is taking data. Each *Subscan* consists of
  - a SOURCE;
  - a HARDWARE CONFIGURATION;

- an `Acquisition Time`;
- a `Setup Time`;
- an antenna trajectory, as derived upstream by the `Pointing Pattern`;
- a scientific intent, which is called a `Subscan Intent` (Table 9.2).

- `Subscan Prototype` is a *Subscan* that has a SOURCE, a HARDWARE CONFIGURATION, and `Subscan Intents` but does not include an `Acquisition Time` or a `Setup Time`.

- `Scan Intent` is a tag that describes the scientific purpose of a set of *Subscans* (e.g., a flux, phase, or bandpass calibration, a pointing, or an observation of a Science Target). A single scan can have multiple Scan Intents. See Table 9.2.

- *Scan* is a group of *Subscans* that share `Scan Intent`. All *Scans* have at least one *Subscan*.

- `maximum duration` has different contextual definitions. It is
  - the maximum length for any single *Scan* including all associated `Setup Times` on an Observing Target;
  - the maximum length of time allowed for a *Subscan*, Scan, all *Subscans*, or all *Scans*.

- Maximum `Acquisition Time` is the maximum time of any single *Scan* on an Observing Target.

- `Requested Time` is the time specified for a Science Target in the *Science Target List*.

- `Science Target Integration Time`(s) is
  - the sum of the `Acquisition Times` for all *Subscans* on a Science Target with `Subscan Intent` ON_SOURCE and associated with a `Scan Intent` of OBSERVE_TARGET. This is greater than or equal to the `Requested Time` when all these *Subscans* are complete.
  - the sum of all `Acquisition Times` for all *Subscans* for all Science Targets with `Subscan Intent` ON_SOURCE and associated with a `Scan Intent` of OBSERVE_TARGET (*Scan List* level).

- Observing Target is a generalization of a Science Target to include Calibrators, such that all Science Targets are Observing Targets, but not all Observing Targets are Science Targets. Observing Targets have
  - a HARDWARE CONFIGURATION,
  - a SOURCE.

- `Time on Observing Target`(s) has different contextual definitions. It is
  - the sum of the `Acquisition Times` for all *Subscans* of this Observing Target. This can be greater than or equal to the `Requested Time` for Science Targets; it can be greater than or equal to the `Science Target Integration Time`.
  - the sum of all `Acquisition Times` for all *Subscans* for all Observing Targets (*Scan List* level).

- Duration has different contextual definitions.
  - Generally, a duration is the total time of a *Subscan*, Scan, all *Subscans*, or all *Scans*. It includes overhead.
  - Specifically, it is the total time of all *Scans*, and Duration = `Setup Time` + `Time on Observing Targets`.

- Overhead has different contextual definitions.
  - Generally, the overhead is any time an antenna is not collecting data on a Science Target.
  - Specifically, Overhead = Duration - `Science Target Integration Times`

**Table 9.2:** Table of Scan and `Subscan Intents`

| Intent | Comment | GBT | VLA | VLBA |
|---|---|---|---|---|
| UNSPECIFIED | *Subscan*; Scan | ✓ | ✓ | |
| ON_SOURCE | *Subscan* | ✓ | ✓ | |
| OFF_SOURCE | *Subscan* | ✓ | ✓ | |
| OBSERVE_TARGET | Scan | ✓ | ✓ | |
| CALIBRATE_AMPLI | Scan | | ✓ | |
| CALIBRATE_BANDPASS | Scan | | ✓ | |
| CALIBRATE_FLUX | Scan | | ✓ | |
| CALIBRATE_FOCUS | Scan | ✓ | | |
| CALIBRATE_PHASE | Scan | | ✓ | |
| CALIBRATE_POINTING | Scan | | ✓ | |
| CALIBRATE_POL_LEAKAGE | Scan | | ✓ | |
| CALIBRATE_POL_ANGLE | Scan | | ✓ | |
| SYSTEM_CONFIGURATION | Scan | | ✓ | |

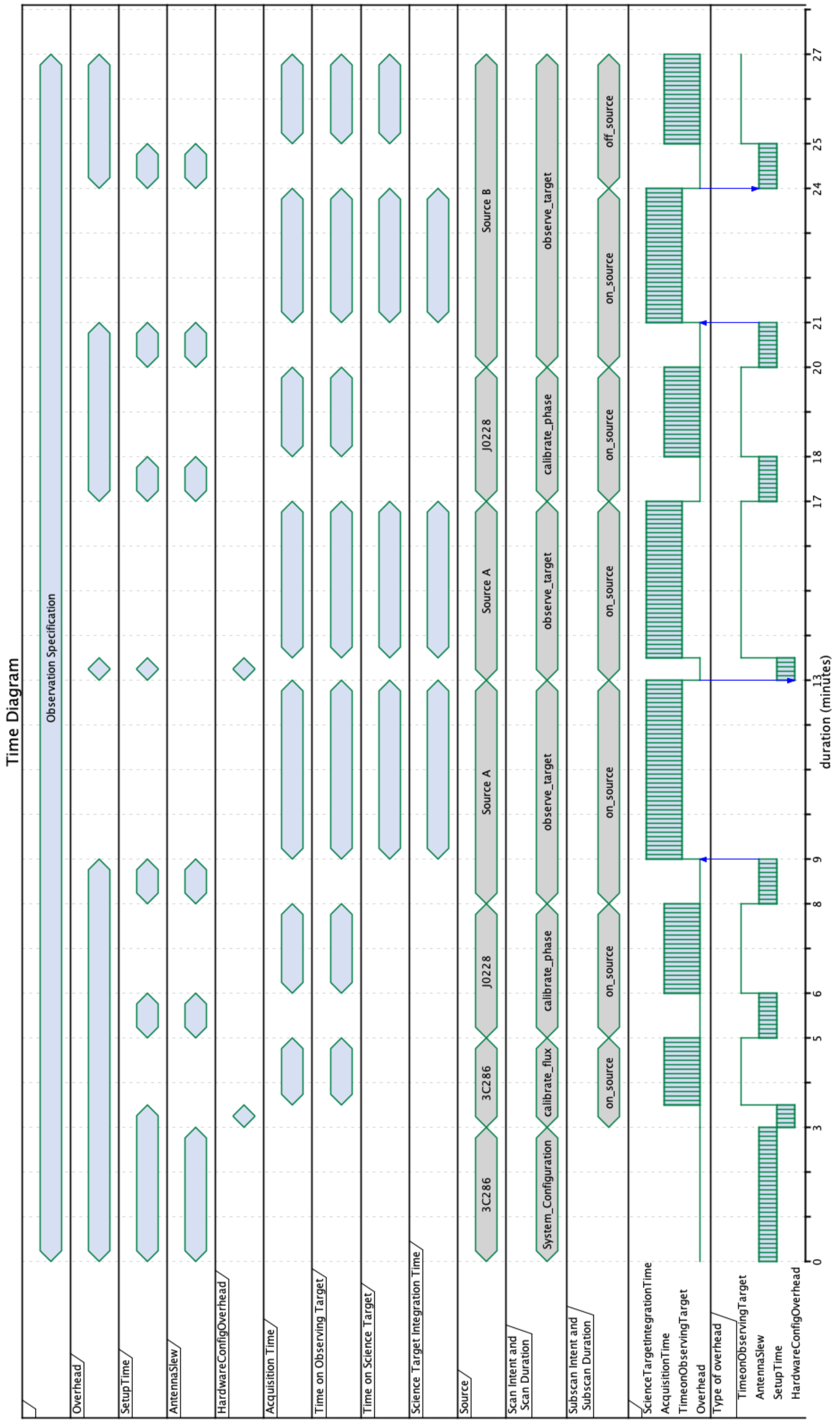



**Figure 9.1:** Hierarchy and concepts of time in the *Observation Planner*.

Finally, we can define the Calibration Plan. A Calibration Plan is expressed as a set of Observing Instructions, which are designed to operate on a set of Science Targets. There are a great number of Observing Instructions (OIs) available to a *Capability*. In this section, we describe the OIs in the context of each *Facility* and leave the details of the *Capability* specific treatment for Chapters 10, 11, and 12.

### 9.2.1 Observing Instructions

Observing Instructions encapsulate information about observations that need to be made, which includes

- how long the observations are required to be observed for,
- information about the ordering of *Scans* and *Subscans*,
- how often the observations are required to be observed.

Observing Instructions have two primary responsibilities in the system. Given their internal state, a list of scans that have already been scheduled and a list of upcoming scans, the OI determines whether or not scans on it's Observing Target(s) should added to the observation. The second responsibility is given a list of previously scheduled *Scans* and a `maximum duration` that may be scheduled, the OI returns a list of scans which advance the state of the Observing Instruction toward completion or satisfy calibration requirements. These manifest as the actions of "Determining if Observation is Required" and "Generating a List of Scans". The TTA System has a set of OIs; the following sections define a non-exhaustive list of the required OIs that include Science Observing Instructions, which are correlated with the *Science Target List*, and Calibrator Observing Instructions, which contain information about the Observing Target that is required for calibration.

OIs are subject to prerequisites, called Prerequisite Observing Instructions. These are actions that must be done before the OI. A Prerequisite Observing Instruction triggers the inclusion of a particular Calibrator OI or Science OI. The prototypical example for this is the Pointing Calibration Observing Instruction. Table 9.3 briefly summarizes the OIs, and Figure 9.2 provides a illustrative guide for the hierarchy.

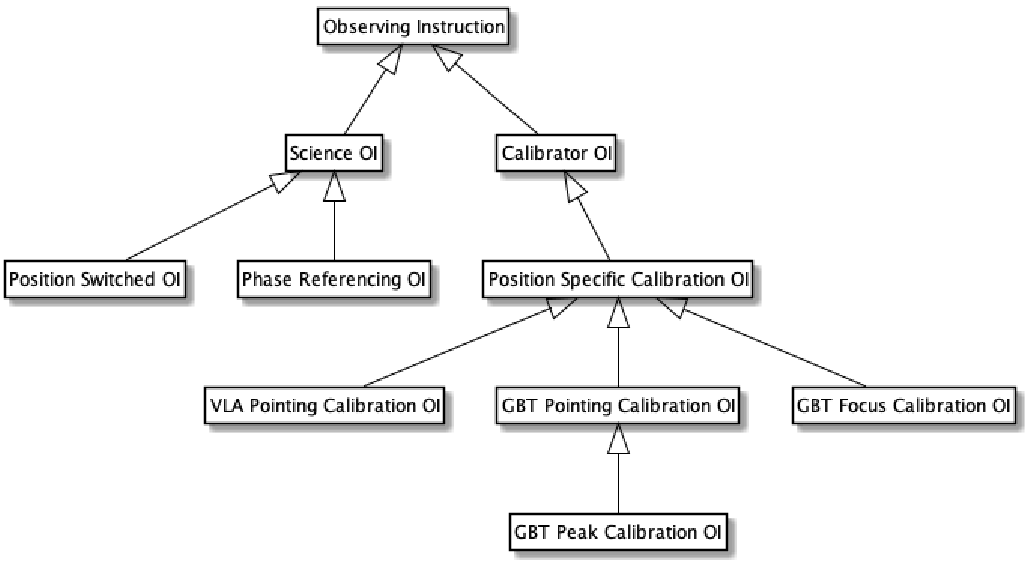



**Figure 9.2:** Hierarchy of the Observing Instructions.

**Table 9.3:** Summary of Observing Instructions

| Name | Summary | Section |
|---|---|---|
| Science OI | Encapsulates the observation of a Science Target | 9.2.2 |
| Position Switching OI | A Science OI for moving between on-source and off-source positions the purposes of calibration. | 9.2.2.1 |
| Phase Referencing OI | A Science OI for implementing the phase referencing technique for interferometry. | 9.2.2.2 |
| Calibrator OI | Encapsulates the observation of a calibrator | 9.2.3 |
| Position Specific Cal OI | Instructions for observations pertaining to a distance on the sky over which an OI is valid. | 9.2.3.1 |
| VLA Pointing Cal OI | A Position Specific OI for VLA pointing calibrations. | 9.2.3.1.1 |
| GBT Pointing Cal OI | A Position Specific OI for GBT pointing calibrations. | 9.2.3.1.2 |
| GBT Focus Cal OI | A Position Specific OI for GBT focus calibrations | 9.2.3.1.3 |

### 9.2.2 Generic Science Observing Instructions

Science Observing Instructions encapsulate the observation of a single row of the *Science Target List*. Science OIs contain the following types of information:

- A list of `Subscan Prototypes` to be realized within each Scan. Each `Subscan Prototype` will have an associated weight, which specifies how the `Requested Time` is distributed (as `Acquisition Time`) amongst the *Subscans* that share a common Observing Target. The default weight is 1.
- The `Requested Time` is the total time that data should be acquired on the Science Target. It does not include time spent moving to the Science Target or time spent on calibration.
- The Maximum Scan may not be defined, but in many cases, it is useful to define a maximum length for any single Scan on an Observing Target.
- A list of Prerequisite Observing Instructions (e.g., Calibrator Observing Instructions) that must be checked to see if they need to be observed prior to the Science OI.

There are multiple types of Science OIs (e.g., Position Switching OI, Phase Referencing OI). The inheritance tree is Observing Instruction:: Science OI, so the types of OIs will each perform the actions of

- Determining if Observation is Required
  - If the `Science Target Integration Time` in previously scheduled *Scans* for this OI is greater than or equal to the `Requested Time` , then this OI does not require observation; otherwise it does.
- Generating a List of Scans
  - Given the set of *Scans* that have previously been scheduled and the information about the Scan that is about to be scheduled for this OI, determine if any of the Prerequisite OIs need to be scheduled. If so, the *Scans* they generate should be inserted at the beginning of the returned list of scans.
  - Science OIs produce a single OBSERVE_TARGET Scan each time they are called. The Scan should be as long as possible subject to the following constraints.
    - The total time of the scan, excluding `Setup Time`, should not exceed the Maximum Scan, if it is specified.

    * The duration of the scan, including overheads, should not exceed the `maximum duration` specified when the list of scans is requested.

  The `Time on Observing Target`, (`Scan Duration - Setup Time`) should be distributed among the *Subscans* according to their specified weights.

#### 9.2.2.1 Science OI: Position Switching Observing Instruction

A Position Switching Observing Instruction is a specific type of Science OI that takes as input

- the position of the Science Target, called the on-source position;
- an off-source position used for calibration;
- a HARDWARE CONFIGURATION;
- the `Requested Time` ;
- a Cycle Time, which is the maximum time to complete one full cycle of on-source and off-source observations.

The Position Switching Observing Instruction has two `Subscan Prototypes` of equal weight. The first `Subscan Prototype` has the off-source position, a HARDWARE CONFIGURATION, and a `Subscan Intent` of OFF_SOURCE. The second has the on-source position, the HARDWARE CONFIGURATION, and the `Subscan Intent` of ON_SOURCE.

The inheritance tree is OI:: Science OI:: Position Switching Observing Instruction, so this OI will perform the actions of

- Determining if Observation is Required
  - The conditions for determining if an observation is required for a Position Switched Observing Instruction are the same as for a Science Observing Instruction.
- Generating a List of Scans
  - In general, the list of scans is the same as for the Science OI; however, the scan, including any overhead, must always be less than or equal to the specified Cycle Time.

#### 9.2.2.2 Science OI: Phase Referencing Observing Instruction

A Phase Referencing Observing Instruction implements the phase referencing technique for interferometry. The Phase Referencing Observing Instruction takes as input

- a Calibrator Observing Instruction,
- one or more Science Observing Instructions,
- a Cycle Time, which is the maximum permitted time between subsequent observations of the Calibrator Observing Instruction.

The inheritance tree is OI:: Science OI:: Phase Referencing Observing Instruction, so this OI will perform the actions of

- Determining if Observation is Required
  - Observation of a Phase Referencing Observing Instruction is required if any of the Science OIs require observation. Note that a Phase Referencing Observing Instruction may not require observation although its Calibrator Observing Instruction will report that it requires observation.

- Generating a list of Scans
  - As with other OI, any Prerequisite Observing Instructions that require observation should be inserted into the returned list of scans prior to the *Scans* generated by this OI.
  - The list of scans generated by the Phase Referencing Observing Instruction should start with a scan created by the Calibrator Observing Instruction. Note that if this scan is identical to the last scan in the input list of scans, the new scan should be omitted.
  - The objective now is to schedule as much time on the Science Targets before the Cycle Time is reached and another scan from the Calibrator Observing Instruction must be scheduled. The available time shall be scheduled such that
    * The Science OIs are prioritized inversely to their completion fraction. Here, the completion fraction is the ratio of the `Science Target Integration Time` already in previous *Scans* to the `Requested Time` of the Science OI.
    * The average time, excluding overhead, for each Science OI is greater than the total overhead for the scan. If this is not the case, remove the lowest priority Science Observing Instructions until it is.
    * The total `Acquisition Time` of the *Scans* should be allocated proportionally to each OI's remaining time (`Requested Time` – previous `Acquisition Time`).
    * Each Science OI should account all of the time allocated to it even if it requires multiple *Scans* to do so; in this case, all of the *Scans* of a Science Observing Instruction should have equal `Acquisition Times`.
  - Finally, another scan from the Calibrator Observing Instruction should be included as the final scan in the returned list of scans.

### 9.2.3 Generic Calibrator Observing Instruction

Calibrator Observing Instructions differ from Science OIs in that `Requested Time` mostly does not affect their need to be observed. The simplest form is the generic Calibrator Observing Instruction which requires

- a list of one or more `Subscan Prototypes`;
- a list of `Scan Intents`; note that each *Scan* has all of these intents;
- the `Acquisition Time` for each *Subscan*;
- the Repeat Time, which is how often this calibration must be repeated (This could be a time or a flag designating that the calibrator should always be observed or should be observed exactly once);
- a list of Prerequisite Observing Instructions (probably Calibrator Observing Instructions) which must be checked to see if they need observed prior to this Calibrator Observing Instruction.

The inheritance tree is OI:: Calibrator OI, so this OI will perform the actions of

- Determining if Observation is Required
  - Whether a Calibrator Observing Instruction needs to be observed is based on the Repeat Time.
    * If the flag designating that it should always be observed is set, then the response should be True.
    * If the flag designating that it should be observed exactly once is set, then the response should be True if and only if the Calibrator has never been observed, otherwise False.

    * If a repeat duration is specified, the response should be True if it has never been observed or if the duration since last observation is greater than or equal to the specified Repeat Time, otherwise False.

- Generating a List of Scans
  - As with other Observing Instructions, any Prerequisite OIs that require observation should be added to the beginning of the return list of scans.
  - Calibrators generally produce a single scan (aside from any prerequisites) with the specified set of Scan Intents. Each `Subscan Prototype` in the input list should be scheduled with the specified `Acquisition Time`. `Setup Times` should be calculated based on the information in the `Subscan Prototype`.

#### 9.2.3.1 Calibrator OI: Position Specific Calibration Observing Instruction

Some forms of calibration are only valid near the region of the sky where the measurement is made. A Position Specific Calibrator Observing Instruction is a type of Calibrator Observing Instruction that takes as input

- the Maximum Region of Validity, which is the distance on the sky over which this Observing Target does not need to be re-observed.

The inheritance tree is OI:: Calibrator OI:: Position Specific Calibration OI, so this OI will perform the actions of

- Determining if Observation is Required
  - The conditions for determining if an observation is required for a Position Specific Calibration Observing Instruction are the same as for a Calibrator OI. If any of those conditions are True, then the response should be True. However even if the response to all of the usual calibrator conditions is False, a Position Specific Calibration Observing Instruction requires observation if any of the next `Subscan Prototypes` have a position that is greater than the Region of Validity from where this calibrator was last observed. Note that it is important to consider the rotation of the Earth while making this calculation.
- Generating a List of Scans
  - No additional modification of this behavior is required beyond that for the Calibration Observing Instruction.

##### 9.2.3.1.1 Position Specific Calibration OI: VLA Pointing Calibration Observing Instruction

A VLA Pointing Calibration Observing Instruction is a type of Position Specific Calibration Observing Instruction which requires only the specification of a SOURCE.

The inheritance tree is OI:: Calibrator OI:: Position Specific Calibration OI:: VLA Pointing Calibration Observing Instruction, so this OI will perform the actions of

- Determining if Observation is Required
  - No additional modification of this behavior is required beyond that for the Position Specific Calibration Observing Instruction.
- Generating a List of Scans
  - The VLA Pointing Calibration is actually composed of 5 *Subscans* but the on-line system provides a single scan shortcut that represents it as a single scan.

- The returned list of scans contains a single scan with a `Scan Intent` consisting only of CALIBRATE_POINTING. The scan should contain a single *Subscan* with an `Acquisition Time` of 150 seconds, the specified FIELD SOURCE, and the specific HARDWARE CONFIGURATION of "X-Band Pointing."
- Overhead should be calculated as usual when moving to the SOURCE.

#### 9.2.3.1.2 Position Specific Calibration OI: GBT Pointing Calibration Observing Instruction

The inheritance tree is OI:: Calibrator OI:: Position Specific Calibration OI:: GBT Pointing Calibration Observing Instruction, so this OI will perform the actions of

##### 9.2.3.1.2.1 GBT Pointing Calibration OI: GBT Peak Calibration Observing Instruction

The inheritance tree is OI:: Calibrator OI:: Position Specific Calibration OI:: GBT Pointing Calibration OI:: GBT Peak Calibration Observing Instruction, so this OI will perform the actions of

#### 9.2.3.1.3 Position Specific Calibration OI: GBT Focus Calibration Observing Instruction

The inheritance tree is OI:: Calibrator OI:: Position Specific Calibration OI:: GBT Focus Calibration Observing Instruction, so this OI will perform the actions of

# 10 Calibration Strategies for GBT Capabilities

The following sections discuss the GBT *Capability* specific Calibration Strategies. Qualitative introductions of Partition Plans and Calibration Plans are in Chapters 9.1 and 9.2, respectively, and it is assumed the reader has some familiarity with the motivation, constraints, and function of the components of the plans as described in these chapters. The implementation details of the algorithm are discussed in Chapters 17 and 18. The sections here detail the *Capability* specific behavior of the plans. A discussion of the merit of the best practices underpinning the GBT Calibration Strategies is available in Section 23.4.

## 10.1 GBT Continuum *Calibration Strategy*

Under Construction.

## 10.2 GBT Spectral Line *Calibration Strategy*

The GBT Spectral Line *Capability* is defined in the 688-TTAT-004-MGMT System Description v3.0. Section 3.3 of the 688-TTAT-xxx-MGMT TTA Use Cases v0.1 detail the GBT Spectral Line use case that motivates the *Calibration Strategy* described here.

### 10.2.1 GBT Spectral Line Partition Plan

A GBT Spectral Line Partition Plan is composed of the following Partition Instructions.

- A Priority PI, which does not have *Capability* specific attributes (§ 9.1.1.6);
- A Calibration Parameter PI, which does have *Capability* specific attributes (§9.1.1.3);
- A Frequency PI, which does have *Capability* specific attributes (§ 9.1.1.5);
- A Distance PI, which does have *Capability* specific attributes (§ 9.1.1.4);
- An Hour Angle PI, which does have *Capability* specific attributes (§ 9.1.2.2);
- A Time PI, which does have *Capability* specific attributes (§ 9.1.2.3);
- A Calibrator PI, which does have *Capability* specific attributes (§ 9.1.2.1).

The subsections below give the details of the Partition Instructions if they have *Capability* specific behavior.

#### 10.2.1.1 GBT Spectral Line Calibration Parameter PI: Threshold and Custom Metric

> **Note**
>
> The details of this PI are under review. These details and the overall behavior of this PI need vetting by the Telescope Subsystem Scientist for the GBT.

For the GBT Spectral Line *Capability*, the CALIBRATION PARAMETERS of Polarization Calibration, Test Source, and Flux Density Calibration may have non unique states for a set of Science Targets.

#### 10.2.1.2 GBT Spectral Line Frequency PI: Threshold and Custom Metric

> **Note**
>
> The details of this PI are under review and need vetting by the Telescope Subsystem Scientist for the GBT.

For the Frequency PI, the *Capability* sets the threshold value. The *Capability* sets the threshold value to 0.95. The *Capability* also defines the custom metric, which evaluates a pair of Science Targets and assigns a numerical value to the pairing that quantifies the dissimilarity of the attributes being compared. In this case, that attribute is the frequency band associated with the HARDWARE CONFIGURATION. An advantage of defining multiple thresholds is it is possible to define a single custom metric that will partition according to best practices without needing to be tailored explicitly for any one list of Science Targets. The Frequency PI custom metric for the GBT Spectral Line *Capability* is primarily motivated by weather considerations; the metric is a straightforward comparison operator to enforce this partitioning scheme.

1. For a pair of HARDWARE CONFIGURATIONS, if there is one unique requested band, the custom metric returns a value of 0;
2. For a pair of HARDWARE CONFIGURATIONS that include PF1:342MHz, PF1:450MHz, PF1:600MHz, PF1:800MHz, PF2, L-, S-, or C-band, the metric returns 0.5;
3. For a pair of HARDWARE CONFIGURATIONS that include X-, Ku-, Ka-, or Q-band, the metric returns 0.5;
4. For a pair of HARDWARE CONFIGURATIONS that include KFPA, W1:MM-F1, W2:MM-F2, W3:MM-F3, W4:MM-F4, Argus, or Mustang, the metric returns 0.5;
5. Otherwise, the metric returns 1.

#### 10.2.1.3 GBT Spectral Line Distance PI: Threshold and Custom Metric

> **Note**
>
> The details of this PI are under review, specifically the values of the threshold and their relation to frequency. These details and the overall behavior of this PI need vetting by the Telescope Subsystem Scientist for the GBT.

For the Distance PI, the *Capability* sets the threshold value. It is efficient for scheduling if the clusters have an angular sizes that reflect calibration recommendations. The PI assesses a set of Science Targets using the if-then rules to select the most restrictive threshold value from the following set:

- Selecting a Pointing Observing Target that is within 10° of the Science Targets;

#### 10.2.1.4 GBT Spectral Line Hour Angle PI

> **Note**
>
> The details of this PI are under review and needs vetting by the Telescope Subsystem Scientist for the GBT.

The Hour Angle PI specifies the following parameters for the GBT Spectral Line *Capability*.

1. The `minimum elevation` is the elevation that an antenna operates above. The PI selects the largest value from the conditions below for those that are appropriate for the set of Science Targets that the PI is assessing.

    (a) The *Specification Constraint* of the *Facility* for this parameter is +5°;

2. The `maximum elevation` is the elevation that an antenna operates below. The PI selects the smallest value from the conditions below for those that are appropriate for the set of Science Targets that the PI is assessing.

    (a) The *Specification Constraint* of the *Facility* for this parameter is +86°;

3. The nominal location of the GBT, which the algorithm uses a longitude of 79.8397°, a latitude of 38.4330°, and an elevation above sea level of 807.43 meters;

With these parameters, the Hour Angle PI defines the `observing window` for one or a group of Science Targets. For a single Science Target, the PI transforms the specified coordinate information of the SOURCE into the local Horizontal Coordinate system of the GBT. This is facilitated by the Astropy package, particularly astropy.coordinates.EarthLocation and astropy.coordinates.SkyCoord. For an entire sidereal day, the algorithm calculates the Azimuth and Altitude of the Science Target. The `observing window` is the sidereal times for which the Science Target has an Altitude above the specified `minimum elevation` and below the `maximum elevation`.

For a group of Science Targets, the coordinate information of the group is defined as the circular mean using the Astropy package astropy.stats.circmean. This is the coordinate that is transformed into the local Horizontal Coordinate information for the group. The definition of the `observing window` is unchanged.

#### 10.2.1.5 GBT Spectral Line Time PI

> **Note**
>
> The details of this PI are under review and needs vetting by the Telescope Subsystem Scientist for the GBT.

The Time PI defines the following parameters for the GBT Spectral Line *Capability*.

1. The `minimum elevation` is the elevation that an antenna operates above. The PI selects the largest value from the conditions below for those that are appropriate for the set of Science Targets that the PI is assessing.

2. The `minimum elevation` is the elevation that an antenna operates above. The PI selects the largest value from the conditions below for those that are appropriate for the set of Science Targets that the PI is assessing.

    (a) The *Specification Constraint* of the *Facility* for this parameter is +5°;

3. The `maximum elevation` is the elevation that an antenna operates below. The PI selects the smallest value from the conditions below for those that are appropriate for the set of Science Targets that the PI is assessing.

    (a) The *Specification Constraint* of the *Facility* for this parameter is +86°;

4. The nominal location of the GBT, which the algorithm uses a longitude of 79.8397°, a latitude of 38.4330°, and an elevation above sea level of 807.43 meters;

5. The `minimum duration` of a *Subscan* is 20 seconds;

6. The `minimum duration per repeat count` is 10 minutes.

7. The `maximum duration per repeat count` is the maximum allowed length of time of any single execution of an *Observation Specification*, excluding overhead. The PI selects the smallest value from the conditions below for those that are appropriate for the set of Science Targets that the PI is assessing.
   (a) For a `scheduling priority` of A-rank, the value is 6 hours.
   (b) For a `scheduling priority` of B-rank, the value is 5 hours.
   (c) For a `scheduling priority` of C-rank, the value is 2 hours.
8. The total `Setup Time` allowed for a cluster must be less than 50% of the `maximum duration per repeat count`.

### 10.2.2 GBT Spectral Line Calibration Plan

The GBT Spectral Line Calibration Plan is as follows. It acts on a set of Science Targets provided by the *Observation Planner*.

- If each of the Science Targets has a `Pointing Pattern` equal to Single Pointing, create a Position Switching Observing Instruction (§ 9.2.2.1) with the Science Target and HARDWARE CONFIGURATION. This OI takes the `Requested Time` as input. The position of the off-source region should be offset from the Science Target by XX, representing an upper limit on set up time due to telescope motion.

  We could just hard code an upper overhead into the Position Switching Observing Instruction; alternatively, we could specify the real off-source position.

- The Cycle Time of the Position Switched Observing Instruction is determined by the receiver (FRONT-END) in use according to Table 20.1.
- For each unique HARDWARE CONFIGURATION, create a Calibrator Observing Instruction (§ 9.2.3) with the single `Scan Intent` of CALIBRATE_FLUX and a single `Subscan Intent` of ON_SOURCE. The HARDWARE CONFIGURATION should be the same as for the Science Target.
- Create a GBT Pointing Calibrator OI (§ 9.2.3.1.2) and a GBT Focus Calibrator OI (§ 9.2.3.1.3) for each unique HARDWARE CONFIGURATION and set them as prerequisites for the Calibrator and Position Switched Observing Instructions that share the HARDWARE CONFIGURATION.

## 10.3 GBT Pulsar *Calibration Strategy*

Under Construction.

## 10.4 GBT Radar *Calibration Strategy*

Under Construction.

# 11 Calibration Strategies for VLA Capabilities

The following sections discuss the VLA Capability specific Calibration Strategies. Qualitative introductions of Partition Plans and Calibration Plans are in Chapters 9.1 and 9.2, respectively, and it is assumed the reader has some familiarity with the intent, motivation, constraints, and function of the components of the plans as described in these chapters. The implementation details of the algorithm are discussed in Chapters 17 and 18. The sections here detail the *Capability* specific behavior of the plans. A discussion of the merit of the best practices underpinning the VLA Calibration Strategies is available in Section 23.2.

## 11.1 VLA Continuum *Calibration Strategy*

The VLA Continuum *Capability* is defined in the 688-TTAT-004-MGMT System Description v3.0 Section 3.1.1 as "observations of emission that is continuous over a large frequency span (e.g., blackbody, free-free, synchrotron, etc.). Such observations require large bandwidths." This *Capability* includes polarimetric observations, which is specified by the selection of the CALIBRATION PARAMETER polarization calibration. Sections 3.1 and 3.2 of the 688-TTAT-xxx-MGMT TTA Use Cases v0.1 detail two VLA Continuum use cases that motivate the *Calibration Strategy* for both polarimetric and non-polarimetric observations, respectively.

### 11.1.1 VLA Continuum Partition Plan

A VLA Continuum Partition Plan is composed of the following Partition Instructions.

- A Priority PI, which does not have *Capability* specific attributes (§ 9.1.1.6);
- An Array Configuration PI, which does not have *Capability* specific attributes (§9.1.1.1);
- A Calibration Parameter PI, which does have *Capability* specific attributes (§9.1.1.3);
- A Frequency PI, which does have *Capability* specific attributes (§ 9.1.1.5);
- A Distance PI, which does have *Capability* specific attributes (§ 9.1.1.4);
- An Hour Angle PI, which does have *Capability* specific attributes (§ 9.1.2.2);
- A Time PI, which does have *Capability* specific attributes (§ 9.1.2.3);
- A Calibrator PI, which does have *Capability* specific attributes (§ 9.1.2.1).

The subsections below give the details of the Partition Instructions if they have *Capability* specific behavior.

#### 11.1.1.1 VLA Continuum Calibration Parameter PI: Threshold and Custom Metric

> **Note**
>
> The details of this PI are under review. These details and the overall behavior of this PI need vetting by the Telescope Subsystem Scientist for the VLA.

For the VLA Continuum *Capability*, the CALIBRATION PARAMETER Polarization Calibration may be non unique for a set of Science Targets.

#### 11.1.1.2 VLA Continuum Frequency PI: Threshold and Custom Metric

> **Note**
>
> The details of this PI are under review, specifically the if-then rules of the threshold selection, the values of the threshold, the if-then rules in the custom metric, and the values that are returned by the metric. These details and the overall behavior of this PI need vetting by the Telescope Subsystem Scientist for the VLA.

For the Frequency PI, the *Capability* sets the threshold value. One value of the threshold is insufficient to dynamically partition the Science Targets however, so a multitude of values for the threshold are defined and can be accessed by the PI via if-then statements. The PI determines which threshold to select by assessing a set of Science Targets; it follows these if-then rules to select the appropriate threshold:

1. If the HARDWARE CONFIGURATIONS of a partitioned *Science Target List* are only associated with one or more frequency bands of 4-, L-, S-, C-, or X-band and the number of unique frequency bands is $\leq 3$, the threshold is set to 1.

2. If the HARDWARE CONFIGURATIONS of a partitioned *Science Target List* are only associated with one or more frequency bands of 4-, L-, S-, C-, or X-band and the number of unique frequency bands is $> 3$, the threshold is set to 0.65.

3. If the HARDWARE CONFIGURATIONS of a partitioned *Science Target List* are only associated with one or more frequency bands of Ku-, K-, Ka-, or Q-band and the number of unique frequency bands is $\leq 2$, the threshold is set to 1.

4. If the HARDWARE CONFIGURATIONS of a partitioned *Science Target List* are only associated with one or more frequency bands of Ku-, K-, Ka-, or Q-band and the number of unique frequency bands is equal to 3, the threshold is set to 0.85.

5. If the HARDWARE CONFIGURATIONS of a partitioned *Science Target List* are only associated with one or more frequency bands of Ku-, K-, Ka-, or Q-band and the number of unique frequency bands is $> 3$, the threshold is set to 0.65.

6. If the HARDWARE CONFIGURATIONS of a partitioned *Science Target List* are associated with one or more frequency bands of 4-, L-, S-, C-, or X-band **and** one or more bands of Ku-, K-, Ka-, or Q-band and the number of unique frequency bands is $\leq 3$, the threshold is set to 0.85.

7. If the HARDWARE CONFIGURATIONS of a partitioned *Science Target List* are associated with one or more frequency bands of 4-, L-, S-, C-, or X-band **and** one or more bands of Ku-, K-, Ka-, or Q-band and the number of unique frequency bands is $> 3$, the threshold is set to 0.65.

The threshold alone does not determine the partitioning. The *Capability* defines the custom metric, which evaluates a pair of Science Targets and assigns a numerical value to the pairing that quantifies the dissimilarity of the attributes being compared. In this case, that attribute is the frequency band associated with the HARDWARE CONFIGURATION. An advantage of defining multiple thresholds is it is possible to define a single custom metric that will partition according to best practices without needing to be tailored explicitly for any one list of Science Targets. The Frequency PI custom metric for the VLA Continuum *Capability* is composed of if-then rules, which are implemented in the following priority queue[1].

1. For a pair of HARDWARE CONFIGURATIONS, if there is one unique requested band, the custom metric returns a value of 0.

[1] These are preliminary filters on frequency to demonstrate the flexibility of the algorithm

2. For a set of Science Targets, if there are $\leq 3$ unique requested bands and the bands only consist of pairing between 4-, L-, S-, C-, or X-band, then for any pair of HARDWARE CONFIGURATIONS, the custom metric returns a value of 0.5.

3. For a set of Science Targets, if there are $> 3$ unique requested bands and the bands only consist of pairing between 4-, L-, S-, C-, or X-band, then the priority queue follows these constraints:

   (a) If the pair of HARDWARE CONFIGURATIONS includes C- and X-band, the custom metric returns 0.3;

   (b) If the pair of HARDWARE CONFIGURATIONS includes C-band (X-band) and another band that is not X-band (C-band) yet X-band (C-band) is in the set of Science Targets, the metric returns 0.7;

   (c) If the pair of HARDWARE CONFIGURATIONS includes L- and S-band, the custom metric returns 0.3;

   (d) If the pair of HARDWARE CONFIGURATIONS includes L-band (S-band) and another band that is not S-band (L-band) yet S-band (L-band) is in the set of Science Targets, the metric returns 0.7;

   (e) If the pair of HARDWARE CONFIGURATIONS only consists of pairings between 4-, P-, and L-, the metric returns 0.3.

   (f) Otherwise, the metric returns 0.2 for any pair of HARDWARE CONFIGURATIONS.

4. For a set of Science Targets, if there are $\leq 2$ unique requested bands and the bands only consist of pairings between Ku-, K-, Ka-, or Q-band, then for any pair of HARDWARE CONFIGURATIONS, the custom metric returns 0.5.

5. For a set of Science Targets, if there are more than 2 but less than 5 unique requested bands and the bands only consist of pairings between Ku-, K-, Ka-, or Q-band, then the priority queue follows these constraints:

   (a) If Q-band is included in the pair of HARDWARE CONFIGURATIONS, then the custom metric returns 0.9.

   (b) Otherwise, the metric returns 0.2 for any pair of HARDWARE CONFIGURATIONS.

6. For a set of Science Targets, if there are more than 4 unique requested bands and the bands only consist of pairings between Ku-, K-, Ka-, or Q-band, then the priority queue follows these constraints:

   (a) If Q-band is included in the pair of HARDWARE CONFIGURATIONS, then the custom metric returns 0.9.

   (b) If the pair of HARDWARE CONFIGURATIONS includes Ka- and K-band, the custom metric returns 0.3;

   (c) If the pair of HARDWARE CONFIGURATIONS includes Ka-band (K-band) and another band that is not K-band (Ka-band) yet K-band (Ka-band) is in the set of Science Targets, the metric returns 0.7;

   (d) If the pair of HARDWARE CONFIGURATIONS includes Ka-band, the metric returns 0.7.

   (e) If the pair of HARDWARE CONFIGURATIONS includes K- and Ku-band, the custom metric returns 0.3.

   (f) If the pair of HARDWARE CONFIGURATIONS includes Ku-band (K-band) and another band that is not K-band (Ku-band) yet K-band (Ku-band) is in the set of Science Targets, the metric returns 0.7.

7. For a set of Science Targets that include X-band and only elements in [Ku- or K-band], then the metric returns 0.6 for any pair of HARDWARE CONFIGURATIONS.

8. If the pair of HARDWARE CONFIGURATIONS includes a pairing between a band in 4-, P-, L-, S-, C-, or X-band and an band in Ku-, K-, Ka-, or Q-band, the metric returns 0.8.

9. Otherwise, the metric returns 0.2 for any pair of HARDWARE CONFIGURATIONS.

For polarimetric observations, there is no change to the custom metric, the conditions on the threshold, and the threshold values.

#### 11.1.1.3 VLA Continuum Distance PI: Threshold and Custom Metric

> **Note**
>
> The details of this PI are under review, specifically the values of the threshold and their relation to frequency. These details and the overall behavior of this PI need vetting by the Telescope Subsystem Scientist for the VLA.

For the Distance PI, the *Capability* sets the threshold value. It is efficient for scheduling if the clusters have an angular extents that reflect calibration recommendations[2]. For a VLA Continuum *Capability*, these include

- Selecting a Pointing Observing Target that is within 10° of the Science Targets;
- Selecting a Phase Reference Observing Target that is within a specified angular distance, which is dependent on the HARDWARE CONFIGURATION. Typically, this is $< 10°$ for high frequency observations and $\sim 15°$ otherwise.

The PI assesses a set of Science Targets using the if-then rules to select the most restrictive threshold value.

#### 11.1.1.4 VLA Continuum Hour Angle PI

The Hour Angle PI specifies the parameters for the VLA Continuum *Capability*.

1. The `minimum elevation` is the elevation that an antenna operates above. The PI selects the largest value from the conditions below.
   (a) The *Specification Constraint* of the *Facility* for this parameter is +8°;
   (b) For HARDWARE CONFIGURATIONS that request frequency ranges between 4- and X-band, this value is +8°;
   (c) For HARDWARE CONFIGURATIONS that request frequency ranges above K-band, this value is +20°;
   (d) If the CALIBRATION PARAMETER Polarization Calibration is selected, this value is +30°.

2. The `maximum elevation` is the elevation that an antenna operates below. The PI selects the smallest value from the conditions below.
   (a) The *Specification Constraint* of the *Facility* for this parameter is +80°;
   (b) For HARDWARE CONFIGURATIONS that request frequency ranges between 4- and X-band, this value is +85°;
   (c) For HARDWARE CONFIGURATIONS that request frequency ranges above K-band, this value is +80°;
   (d) If the CALIBRATION PARAMETER Polarization Calibration is selected, this value is +80°.

[2] https://science.nrao.edu/facilities/vla/docs/manuals/obsguide/calibration

3. the nominal location of the VLA, which the algorithm uses a longitude of 107.6184°, a latitude of 34.0784°, and an elevation above sea level of 2124 meters;

With these parameters, the Hour Angle PI defines the `observing window` for one or a group of Science Targets. For a single Science Target, the PI transforms the specified coordinate information of the SOURCE into the local Horizontal Coordinate system of the VLA. This is facilitated by the Astropy package, particularly astropy.coordinates.EarthLocation and astropy.coordinates.SkyCoord. For an entire sidereal day, the algorithm calculates the Azimuth and Altitude of the Science Target. The `observing window` is the sidereal times for which the Science Target has an Altitude above the specified `minimum elevation` and below the `maximum elevation`.

For a group of Science Targets, the coordinate information of the group is defined as the circular mean using the Astropy package astropy.stats.circmean. This is the coordinate that is transformed into the local Horizontal Coordinate information for the group. The definition of the `observing window` is unchanged[3].

#### 11.1.1.5 VLA Continuum Time PI

> **Note**
>
> The details of this PI are under review and needs vetting by the Telescope Subsystem Scientist for the VLA.

The Time PI defines the following parameters for the VLA Continuum *Capability*.

1. The `minimum elevation` is the elevation that an antenna operates above. The PI selects the largest value from the conditions below for those that are appropriate for the set of Science Targets that the PI is assessing.
   (a) The *Specification Constraint* of the *Facility* for this parameter is +8°;
   (b) For HARDWARE CONFIGURATIONS that request frequency ranges between 4- and X-band, this value is +8°;
   (c) For HARDWARE CONFIGURATIONS that request frequency ranges above K-band, this value is +20°;
   (d) If the CALIBRATION PARAMETER Polarization Calibration is selected, this value is +30°.
2. The `maximum elevation` is the elevation that an antenna operates below. The PI selects the smallest value from the conditions below for those that are appropriate for the set of Science Targets that the PI is assessing.
   (a) The *Specification Constraint* of the *Facility* for this parameter is +80°;
   (b) For HARDWARE CONFIGURATIONS that request frequency ranges between 4- and X-band, this value is +85°;
   (c) For HARDWARE CONFIGURATIONS that request frequency ranges above K-band, this value is +80°;
   (d) If the CALIBRATION PARAMETER Polarization Calibration is selected, this value is +80°.
3. the nominal location of the VLA, which the algorithm uses a longitude of 107.6184°, a latitude of 34.0784°, and an elevation above sea level of 2124 meters;
4. the `minimum duration` of a *Subscan* is 20 seconds;
5. the `minimum duration per repeat count` is 20 minutes;

[3] Though, it could be smarter

6. The `maximum duration per repeat count` is the maximum allowed length of time of any single execution of an *Observation Specification*, excluding overhead. The PI selects the smallest value from the conditions below for those that are appropriate for the set of Science Targets that the PI is assessing.
   (a) For a `scheduling priority` of A-rank, the value is 4 hours.
   (b) For a `scheduling priority` of B-rank, the value is 3 hours.
   (c) For a `scheduling priority` of C-rank, the value is 2.5 hours.
7. The total `Setup Time` allowed for a cluster must be less than 50% of the `maximum duration per repeat count`.

### 11.1.2 VLA Continuum Calibration Plan

The VLA Continuum Calibration Plan is as follows. It acts on a set of Science Targets provided by the *Observation Planner*.

- For each Science Target, create a Science Target Observing Instruction with the SOURCE, HARDWARE CONFIGURATION, and `Requested Time` (§ 9.2.2) of the Science Target.
- For each Phase Partition Instruction Group, create a Phase Referencing Observing Instruction. The Cycle Time should be set according to Table 20.1.
- For each Phase Referencing Observing Instruction, select an appropriate calibrator (§ 21.3) and create a Calibrator Observing Instruction (§ 9.2.3) that has Scan Intents of CALIBRATE_AMPLI and CALIBRATE_PHASE and is flagged as always requiring observation. The `Acquisition Time` should be the maximum of
  - a time based on required signal to noise ratio and the flux density of the source;
  - 180 seconds.
- Select Observing Targets appropriate for the intents of CALIBRATE_FLUX and CALIBRATE_BANDPASS for each unique HARDWARE CONFIGURATION (§ 21.3). Create Calibrator Observing Instructions with the execute "once flag" set for each of the selected calibrators. The `Acquisition Time` should be the maximum of
  - a time based on required signal to noise ratio and the flux density of the source;
  - 180 seconds.
- For each Polarization Partition Instruction Group, select an appropriate Observing Target for the intents of CALIBRATE_POL_LEAKAGE and create a Calibrator Observing Instruction. The `Acquisition Time` should be the maximum of
  - a time based on required signal to noise ratio and the flux density of the source;
  - 180 seconds.
- For each Pointing Partition Instruction Group, create a VLA Pointing Observing Instruction 9.2.3.1.1. Set this as a prerequisite for all Observing Instructions that have HARDWARE CONFIGURATIONS above 15 GHz. The `Acquisition Time` should be the maximum of
  - a time based on required signal to noise ratio and the flux density of the source;
  - 180 seconds.

## 11.2 VLA Spectral Line *Calibration Strategy*

Under Construction.

## 11.3 | VLA Pulsar *Calibration Strategy*

Under Construction.

# 12 Calibration Strategies for VLBA Capabilities

The following sections discuss the VLBA *Capability* specific Calibration Strategies. Qualitative introductions of Partition Plans and Calibration Plans are in Chapters 9.1 and 9.2, respectively, and it is assumed the reader has some familiarity with the intent, motivation, constraints, and function of the components of the plans as described in these chapters. The implementation details of the algorithm are discussed in Chapters 17 and 18. The sections here detail the *Capability* specific behavior of the plans. A discussion of the merit of the best practices underpinning the VLBA Calibration Strategies is available in Section 23.3.

## 12.1 VLBA Continuum *Calibration Strategy*

Under Construction.

### 12.1.1 VLBA Continuum Partition Plan

A VLBA Continuum Partition Plan is composed of the following Partition Instructions.

- A Priority PI, which does not have *Capability* specific attributes (§ 9.1.1.6);
- An Array Subset PI, which does not have *Capability* specific attributes (§9.1.1.2);
- A Calibration Parameter PI, which does have *Capability* specific attributes (§9.1.1.3);
- A Frequency PI, which does have *Capability* specific attributes (§ 9.1.1.5);
- A Distance PI, which does have *Capability* specific attributes (§ 9.1.1.4);
- An Hour Angle PI, which does have *Capability* specific attributes (§ 9.1.2.2);
- A Time PI, which does have *Capability* specific attributes (§ 9.1.2.3);
- A Calibrator PI, which does have *Capability* specific attributes (§ 9.1.2.1).

The subsections below give the details of the Partition Instructions if they have *Capability* specific behavior.

#### 12.1.1.1 VLBA Continuum Calibration Parameter PI: Threshold and Custom Metric

> Note
>
> The details of this PI are under review. These details and the overall behavior of this PI need vetting by the Telescope Subsystem Scientist for the VLBA.

For the VLBA Continuum *Capability*, the CALIBRATION PARAMETERS of Polarization Calibration and Flux Density Calibration may have non unique states for a set of Science Targets.

#### 12.1.1.2 VLBA Continuum Frequency PI: Threshold and Custom Metric

> **Note**
>
> The details of this PI are under review and need vetting by the Telescope Subsystem Scientist for the VLBA.

For the Frequency PI, the *Capability* sets the threshold value. The *Capability* sets the threshold value to 0.95. The *Capability* also defines the custom metric, which evaluates a pair of Science Targets and assigns a numerical value to the pairing that quantifies the dissimilarity of the attributes being compared. In this case, that attribute is the frequency band associated with the HARDWARE CONFIGURATION. An advantage of defining multiple thresholds is it is possible to define a single custom metric that will partition according to best practices without needing to be tailored explicitly for any one list of Science Targets. The Frequency PI custom metric for the VLBA Continuum *Capability* is primarily motivated by weather considerations; the metric is a straightforward comparison operator to enforce this partitioning scheme.

1. For a pair of HARDWARE CONFIGURATIONS, if there is one unique requested band, the custom metric returns a value of 0;
2. For a pair of HARDWARE CONFIGURATIONS that include 90cm, 50cm, or 21cm, the metric returns 0.5;
3. For a pair of HARDWARE CONFIGURATIONS that include 13 cm, the metric returns 0.5;
4. For a pair of HARDWARE CONFIGURATIONS that include 6cm, 4cm, 2cm, 1cm, 7mm, or 3mm, the metric returns 0.5;
5. Otherwise, the metric returns 1.

#### 12.1.1.3 VLBA Continuum Distance PI: Threshold and Custom Metric

> **Note**
>
> The details of this PI are under review, specifically the values of the threshold and their relation to frequency. These details and the overall behavior of this PI need vetting by the Telescope Subsystem Scientist for the VLBA.

For the Distance PI, the *Capability* sets the threshold value. It is efficient for scheduling if the clusters have an angular sizes that reflect calibration recommendations. The PI assesses a set of Science Targets using the if-then rules to select the most restrictive threshold value from the following set:

- Selecting a Pointing Observing Target that is within 15° of the Science Targets;

#### 12.1.1.4 VLBA Continuum Hour Angle PI

The Hour Angle PI specifies the parameters for the VLBA Continuum *Capability*.

1. The `minimum elevation` is the elevation that an antenna operates above. The PI selects the largest value from the conditions below.
   (a) The *Specification Constraint* of the *Facility* for this parameter is +8°;
   (b) For HARDWARE CONFIGURATIONS that request frequency ranges between 4- and X-band, this value is +8°;

(c) For HARDWARE CONFIGURATIONS that request frequency ranges above K-band, this value is +20°;

(d) If the CALIBRATION PARAMETER Polarization Calibration is selected, this value is +30°.

2. The `maximum elevation` is the elevation that an antenna operates below. The PI selects the smallest value from the conditions below.

   (a) The *Specification Constraint* of the *Facility* for this parameter is +80°;

   (b) For HARDWARE CONFIGURATIONS that request frequency ranges between 4- and X-band, this value is +85°;

   (c) For HARDWARE CONFIGURATIONS that request frequency ranges above K-band, this value is +80°;

   (d) If the CALIBRATION PARAMETER Polarization Calibration is selected, this value is +80°.

3. There is not a single location for the VLBA but the VLBA operates in GST, so nominally the location is Greenwich (longitude = -0.001545°, latitude = 51.477928°, elevation = 46 meters). However when calculating the `observing window` or similar parameters, the location of each station is considered independently and then converted to GST. The algorithm uses https://science.nrao.edu/facilities/vlba/docs/manuals/oss/sites to source the locations of the stations.

With these parameters, the Hour Angle PI defines the `observing window` for one or a group of Science Targets. For the VLBA, the `observing window` must account for the wide spread nature of the VLBA stations, so each station is considered independently at first. For each VLBA station, the PI transforms the specified coordinate information of a SOURCE into the local Horizontal Coordinate system for that station. This is facilitated by the Astropy package, particularly astropy.coordinates.EarthLocation and astropy.coordinates.SkyCoord. For an entire sidereal day, the algorithm calculates the Azimuth and Altitude of the Science Target. The sidereal times for which the Science Target has an Altitude above the specified `minimum elevation` and below the `maximum elevation` is transformed into Greenwich Sidereal Time (GST) and recorded as a `station observing window` for this Science Target at this station.

Once a `station observing window` is defined for the specified stations, the algorithm uses a comparison operator to set the `observing window` equal to the times (in GST) for which the Science Target is available for observation at all the specified stations. An illustration of this approach is shown in Figure 12.1.

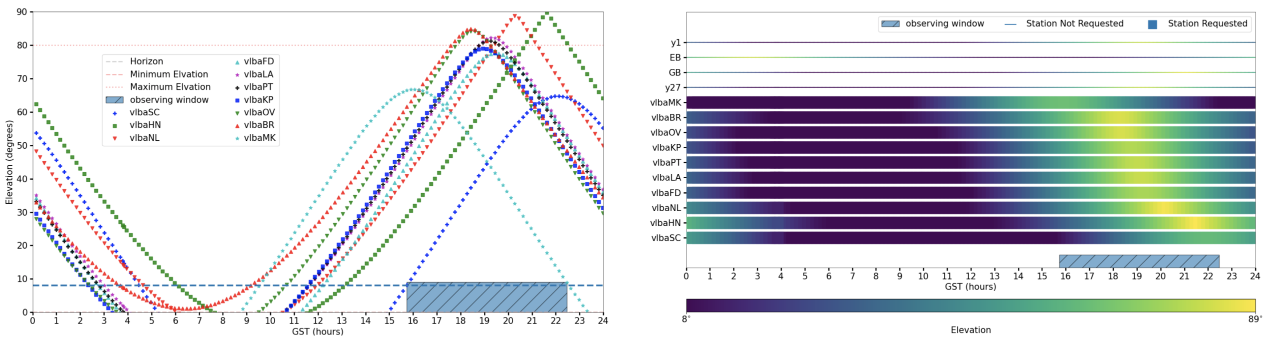



**Figure 12.1:** (*left*) Plot of Elevation vs GST for a SOURCE. Each curve represents a different VLBA station. The hash blue bar shows the `observing window` for the source. The dashed blue line represents the `minimum elevation` and the dotted red line is the `maximum elevation`. (*right*) Bar plot of the elevation vs GST for each VLBA station for a SOURCE. The color map is the elevation of the source; if the station is not requested, it is shown as a line. The `observing window` is the hash blue bar.

For groups of Science Targets, the coordinate information of the group is defined as the circular mean using the Astropy package astropy.stats.circmean. This is the coordinate that is transform into the local Horizontal Coordinate information for the group for each station.

#### 12.1.1.5 VLBA Continuum Time PI

> **Note**
>
> The details of this PI are under review and needs vetting by the Telescope Subsystem Scientist for the VLBA.

The Time PI defines the following parameters for the VLBA Continuum *Capability*.

1. The `minimum elevation` is the elevation that an antenna operates above. The PI selects the largest value from the conditions below for those that are appropriate for the set of Science Targets that the PI is assessing.
   (a) The *Specification Constraint* of the *Facility* for this parameter is +8°;
   (b) For HARDWARE CONFIGURATIONS that request frequency ranges in bands between 90cm and 2cm, this value is +8°;
   (c) For HARDWARE CONFIGURATIONS that request frequency ranges in bands above 2cm, this value is +20°;
   (d) If the CALIBRATION PARAMETER Polarization Calibration is selected, this value is +30°.
2. The `maximum elevation` is the elevation that an antenna operates below. The PI selects the smallest value from the conditions below for those that are appropriate for the set of Science Targets that the PI is assessing.
   (a) The *Specification Constraint* of the *Facility* for this parameter is +80°;
   (b) For HARDWARE CONFIGURATIONS that request frequency ranges in bands between 90cm and 2cm, this value is +85°;
   (c) For HARDWARE CONFIGURATIONS that request frequency ranges in bands above 2cm, this value is +80°;
   (d) If the CALIBRATION PARAMETER Polarization Calibration is selected, this value is +80°.
3. There is not a single location for the VLBA but the VLBA operates in GST, so nominally the location is Greenwich (longitude = -0.001545°, latitude = 51.477928°, elevation = 46 meters). However when calculating the `observing window` or similar parameters, the location of each station is considered independently and then converted to GST. The algorithm uses https://science.nrao.edu/facilities/vlba/docs/manuals/oss/sites to source the locations of the stations.
4. The `minimum duration` of a *Subscan* is 20 seconds;
5. The `minimum duration per repeat count` is 5 minutes;
6. The `maximum duration per repeat count` is the maximum allowed length of time of any single execution of an *Observation Specification*, excluding overhead. The PI selects the smallest value from the conditions below for those that are appropriate for the set of Science Targets that the PI is assessing.
   (a) For a `scheduling priority` of A-rank, the value is 4 hours.
   (b) For a `scheduling priority` of B-rank, the value is 3 hours.
   (c) For a `scheduling priority` of C-rank, the value is 2.5 hours.
7. The total `Setup Time` allowed for a cluster must be less than 50% of the `maximum duration per repeat count`.

## 12.2 VLBA Spectral Line Calibration Strategy

Under Construction.

## 12.3 VLBA Pulsar Calibration Strategy

Under Construction.

# 13 | Scheduling Strategies for GBT Capabilities

## 13.1 | GBT Continuum *Scheduling Strategy*

Under Construction.

## 13.2 | GBT Spectral Line Scheduling Strategy

Under Construction.

## 13.3 | GBT Pulsar Scheduling Strategy

Under Construction.

## 13.4 | GBT Radar Scheduling Strategy

Under Construction.

# 14 | Scheduling Strategies for VLA Capabilities

## 14.1 | VLA Continuum Scheduling Strategy

Under Construction.

## 14.2 | VLA Spectral Line Scheduling Strategy

Under Construction.

## 14.3 | VLA Pulsar Scheduling Strategy

Under Construction.

# 15 | Scheduling Strategies for VLBA Capabilities

## 15.1 | VLBA Continuum Scheduling Strategy

Under Construction.

## 15.2 | VLBA Spectral Line Scheduling Strategy

Under Construction.

## 15.3 | VLBA Pulsar Scheduling Strategy

Under Construction.

# 16 Overview of Observation Planner

The *Observation Planner* algorithm converts the *Science Target List* into one or more *Observation Specifications*. The *Observation Planner* contains the three parts that facilitate the conversion of the *Science Target List*: *Phase 1*, *Phase 2* and *Phase 3*. These phases utilize the *Calibration Strategy* and the *Scheduling Strategy*, which are part of the *Science Target List*.

# 17 | Observation Planner Phase 1

*Phase 1* applies two stages to the partitioning of the *Science Target List* to form clusters of Science Targets that can then be scheduled. Prior to the *Phase 1*, no partitioning has occurred to the *Science Target List*. In the Initial Partitioning stage, the algorithm groups Science Targets by HARDWARE CONFIGURATIONS, spatial, and CALIBRATION PARAMETERS. In this chapter, we define a cluster as being composed of one or more Science Targets and one that has been evaluated as a suitable collection. The usage of the word "cluster" in this section maintains this definition, whereas the term "group" means a collection of one or more Science Targets whose state as an optimal cluster is unknown.

It is necessary to introduce the concept for a `repeat count` and a `Partition Requested Time`. As the `Requested Time` of a Science Target can be larger than the time the Science Target is available in the sky, it is natural to instead schedule multiple observations of the Science Target. The `Partition Requested Time`, the `repeat count`, and the `Requested Time` are related such that

$$\texttt{Partition Requested Time} \times \texttt{repeat count} \geq \text{Requested Time.} \tag{17.0.1}$$

It is required that Equation 17.0.1 be true, and the `repeat count` is required to be a positive integer. In the algorithm, the `repeat count` is rounded up when calculated. Once a cluster is formed, the `Partition Requested Time` is the initial estimate for the amount of time an antenna will collect data on the specific Science Target in a single *Observation Specification*. A `repeat count` is specified for Science Targets by the *Phase 1* algorithm and inherited by clusters and *Observation Specifications*.

In Final Partitioning, the algorithm assess the groups for efficiency and it can combine these groups if certain criteria are met. The algorithm evaluate groups to determine if they are optimal clusters. An optimal cluster is (insipidly) referred to here as a "Good Cluster", which is defined as having the following attributes:

1. There may be many clusters but any one Science Target belongs to exactly one cluster.

2. All of the constituents of a cluster are required to have the same `repeat count`.

3. For all Science Targets in a cluster, the `Partition Requested Time` of any one Science Target cannot be longer than the length of time that SOURCE is available for observation, called the `observing window`. Specifically, the `observing window` is the time when

   - the altitude of a SOURCE is above the `minimum elevation` and
   - the altitude of a SOURCE is below the `maximum elevation`.

   The Hour Angle Partition Instruction specifies the `observing window` (§ 9.1.2.2).

4. The `observing window` associated with a cluster may not be oversubscribed. It is a necessary to validate that the `Partition Requested Time` for each Science Target in a group is feasible, given the `Partition Requested Times` of the other group members and the `observing window`[1].

5. A cluster's total `Partition Requested Time` must be less than or equal to the `maximum duration`. The Time Partition Instruction specifies this value (§ 9.1.2.3).

6. A Science Target's `Partition Requested Time` cannot be less than the `minimum duration` of a *Subscan*. The Time Partition Instruction specifies this value (§ 9.1.2.3).

7. A cluster's total `Partition Requested Time` is greater than or equal to 90% of the `maximum duration`. This criterion is necessary to prioritize the formation of clusters that are not underfull. It is not fundamentally motivated by the *Capability*. It is a constraint on the behavior of the algorithm and is determined through tests on the algorithm, see Section 22.2.

[1]This portion of the algorithm could be upgraded to make smarter groups. Using the circular mean to dictate the `observing window` is easy and computationally inexpensive for the moment. Other treatments may be better and should be investigated for efficacy.

8. The total `Setup Time` associated with a cluster is less than or equal to 10% of the `maximum duration`. This criterion is necessary to prioritize the formation of clusters that are not overhead dominated. It is not fundamentally motivated by the *Capability*. It is a constraint on the behavior of the algorithm and is determined through tests on the algorithm, see Section 22.2.

9. The Science Targets in the cluster, on average, have `Partition Requested Times` greater than a specified `minimum duration` (e.g., 20 minutes) for the Science Targets with `Requested Times` greater than the `minimum duration`. The Time Partition Instruction specifies this value (§ 9.1.2.3). This prioritizes groups whose constituents have longer `Partition Requested Time` over groups with large REPEAT COUNTS and small `Partition Requested Time` per Science Target. This parameter functions to optimize the clustering; it is not a *Specification Constraint*, and it is not fundamentally motivated but chosen to address the tendency of the algorithm to make groups with large REPEAT COUNTS and small `Partition Requested Times`. A discussion on this parameter is available in Section 22.2. This parameter is only considered when a `Requested Time` is greater than the specific `minimum duration`; otherwise, this criteria would be impossible to implement.

The algorithm will also define sets of Science Targets within a cluster that can share calibrators, for example a Phase Referencing calibrator. [Posit] At the conclusion of the *Phase 1*, the clusters of Science Targets map to *Observation Specifications* and the clusters should be treated independently of one another in the *Observation Planner Phase 2* and 3. Chapters 10, 11, 13, and 14 provide a functional and science motivated overview of the Partition Instructions per *Capability*.

## 17.1 Phase 1 Algorithm

Here, we present the algorithmic implementation details of the *Phase 1* algorithm.

1. At the start of *Phase 1*, the algorithm calls the *Calibration Strategy* factory for each the Science Targets to produce Partition Plans.

2. For each Partition Instruction in the Partition Plan, it creates a Partition Instruction Group index and sets them equal to -1 for each Science Target. This functionality allows the algorithm to assess the state of partitioning. For example, it can determine if all Science Targets have been evaluated using the Distance PI constraints if each Science Target has been assigned a Distance Partition Instruction Group. This enables the algorithm to recall a specific Partition Instruction Group when necessary (e.g., partitioning to determine common groups for Phase Reference Calibrators). As partitioning progresses, a partitioned set of Science Targets is assigned a common Partition Instruction Group index, indicating that they belong to the same group for the PI.

3. The algorithm creates an Initial Partition Group and a Final Partition Group index for each Science Target and sets them equal to -1 for each Science Target.

4. The algorithm sets the `Partition Requested Time` for each Science Target to be equal to its `Requested Time` and sets the `repeat count` per Science Target equal to 1.

5. The algorithm initiates Initial Partitioning, as described in Section 17.1.1. At the end of Initial Partitioning, the algorithm has assigned Science Targets to groups, and this information is stored in the Initial Partition Group index.

6. The algorithm then initiates Final Partitioning, as described in Section 17.1.2. At the end of Final Partitioning, the algorithm has assigned Science Targets to clusters. This partitioning information is recorded in the Final Partition Group index. The algorithm has also assigned subsets of Science Targets within a Final Partition Group to Calibration PI Groups.

7. The *Phase 1* algorithm is now complete; the Final Partition Groups have a 1-1 mapping with *Observation Specifications*.

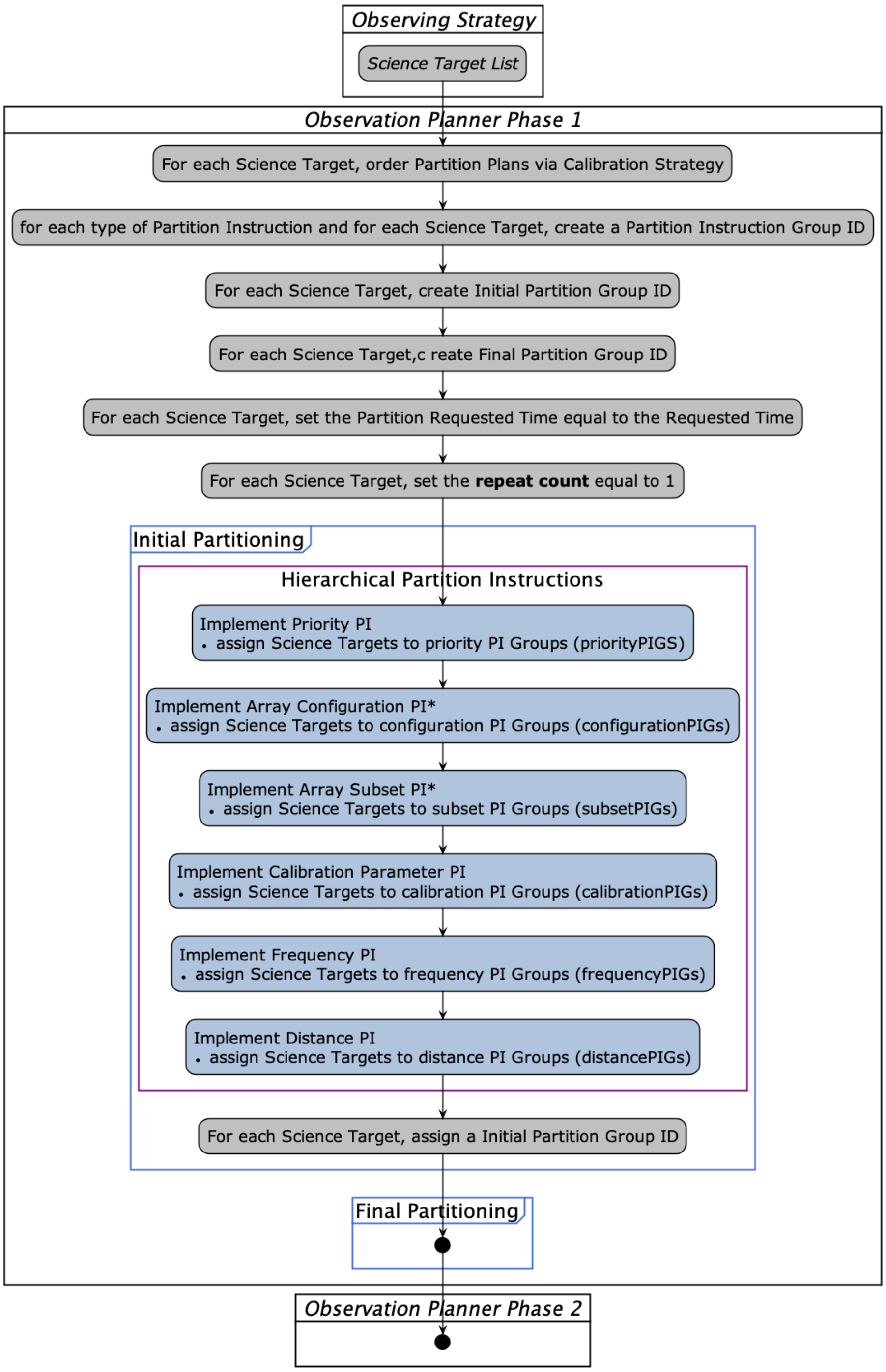



**Figure 17.1:** Diagram of the Phase 1: Initial Partitioning algorithm.

### 17.1.1 Initial Partitioning

In Initial Partitioning, the algorithm evaluates the Partition Plan. The first PI is a Hierarchical PI, so the algorithm follows the procedure described in Section 17.1.1.1, using the custom metric and threshold specified by the Hierarchical PI. Then for the resulting groups, the algorithm assigns the Science Targets to a Partition Instruction Group. For example, if the Partition Instruction is a Priority PI, each Science Target is assigned to a Priority PI Group.

The number of partitioned groups increases as partitioning progresses and grouped Science Targets are assigned to a common PI Group. Once the algorithm forms a PI Group, it will treat each group independently for the next Partition Instruction. For example, if there are $N$ Priority PI Groups, and the next PI is a Frequency PI, the algorithm will implement the Frequency PI for each of the $N$ groups independently.

The algorithm tracks the state of partitioning through the Partition Instruction Group indexes. When all Science Targets are associated with PI Groups, i.e., when all of the PI Group indexes are not equal to -1, the algorithm will set the Initial Partition Group index. It does so by copying the last Partition Instruction Group assignments to the Initial Partition Group index. This is then the end of Initial Partitioning.

#### 17.1.1.1 Implementation of Hierarchical Partition Instructions

The algorithm utilizes Scipy's Hierarchy[2], Linkage[3], and fcluster[4] to perform the partitioning. The algorithm will

- Construct a linkage matrix using scipy.cluster.hierarchy.linkage, which takes as inputs
  - a set of Science Targets,
  - a custom metric that defines a dissimilarity function,
  - a method that specifies the linkage algorithm to use in calculating the dissimilarity value. The default method is "complete".
- Form flat clusters based on the linkage matrix using scipy.cluster.hierarchy.fcluster, which takes as inputs
  - a linkage matrix,
  - a criterion for forming flat clusters,
  - a threshold for forming clusters.

The use the criterion of "distance", which "forms flat clusters so that the original observations in each flat cluster have no greater a cophenetic distance than [the threshold]"[4]. The use of distance here is not necessarily the angular distance between astronomical objects; it is the dissimilarity between two objects for a given metric.

The clusters formed by fcluster consist of a hierarchical list of nodes. The nodes are the input Science Targets and the clusters to which the Science Targets belong. Commonly, a Dendrogram tree is shown to visualize the relationship between the nodes. In Figure 17.2, the tree shows the hierarchical relationship between Nodes

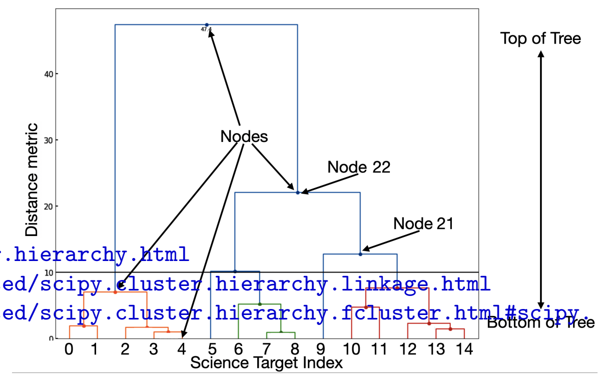



**Figure 17.2:** Example Dendrogram tree for 15 Science Targets using angular distance on the sky as the custom metric. The solid black line at y = 10 cuts the tree, yielding 5 clusters.

[2] https://docs.scipy.org/doc/scipy/reference/cluster.hierarchy.html

[3] https://docs.scipy.org/doc/scipy/reference/generated/scipy.cluster.hierarchy.linkage.html

[4] https://docs.scipy.org/doc/scipy/reference/generated/scipy.cluster.hierarchy.fcluster.html#scipy.cluster.hierarchy.fcluster

for a list of Science Targets based on the angular separation between the SOURCES. The distance between nodes is determined by the choice of the distance function and the linkage algorithm in scipy.cluster.hierarchy.linkage. The solid, horizontal black line represents a cut across the tree; the partitioning occurs where the cut intersects with the branches of tree. The height at which the cut is made is determined by the value of the threshold for forming clusters in scipy.cluster.hierarchy.fcluster.

### 17.1.2 Final Partitioning

In Final Partitioning, the algorithm performs three tasks.

1. It enforces the definition of what constitutes an optimal cluster (§ 17.1.2.1).
2. It defines the distribution of clusters per *Observation Specification* (§ 17.1.2.2).
3. It specifies subsets within clusters that have common Calibration Plans (§ 17.1.2.3).

The algorithm facilitates these actions via if-then rules and in the order listed here. The constraints are provided by the Partition Instructions. Figure 17.3 shows the workflow of the *Phase 2* algorithm.

#### 17.1.2.1 Enforcing Optimal Clusters

The algorithm attempts to balance efficiency, recommended observing practices, and the science requirements when evaluating and refactoring groups of Science Targets. The Initial Partition Groups are an initial estimate of optimal groups, but they are not necessarily optimal clusters. The algorithm will evaluate the Initial Partition Groups and refactor them if necessary.

Of the Hierarchical Partition Instructions, the Distance PI is the most arbitrary. The distance threshold is roughly motivated by best practices for calibration and by simulating clustering with different source distributions (see § 23.4). Depending on the number of sources and their distribution on the sky, the optimal distance threshold can vary substantially in that sometimes it is more efficient to have fewer clusters with sources that are spread across the sky versus many small clusters. An experienced observer would decide which is preferable: overhead due to the antennas slewing across the sky or overhead due to many System Configuration *Subscans*, for example.

The algorithm is designed to automate this decision process in a decision tree. It initially implements the Distance PI with a "best guess" threshold, assesses the consequences of that threshold, and refactors the partitioning. To do so, the algorithm requires knowledge of the state of partitioning before the Distance PI was implemented. For all *Capabilities*, the Frequency PI is directly upstream of the Distance PI in the Partition Plan, so the Frequency PI Group represents the state of partitioning before the Distance PI was implemented. In Final Partitioning, the algorithm considers groups of Science Targets as defined by the Initial Partition Group, the Frequency PI Group, and the Distance PI Group to automate the creation of optimal clusters.

For each Initial Partition Group, the algorithm performs the following actions.

1. The algorithm uses the Hour Angle Partition Instruction to define the `observing window` for the group of Science Targets.
2. The algorithm compares the sum of the `Partition Requested Time` for the group to the `observing window` or the `maximum duration per repeat count`, whichever is smaller.
   (a) If the total `Partition Requested Time` of the group is greater than either, the algorithm triggers a refactoring of the group. See point 4.

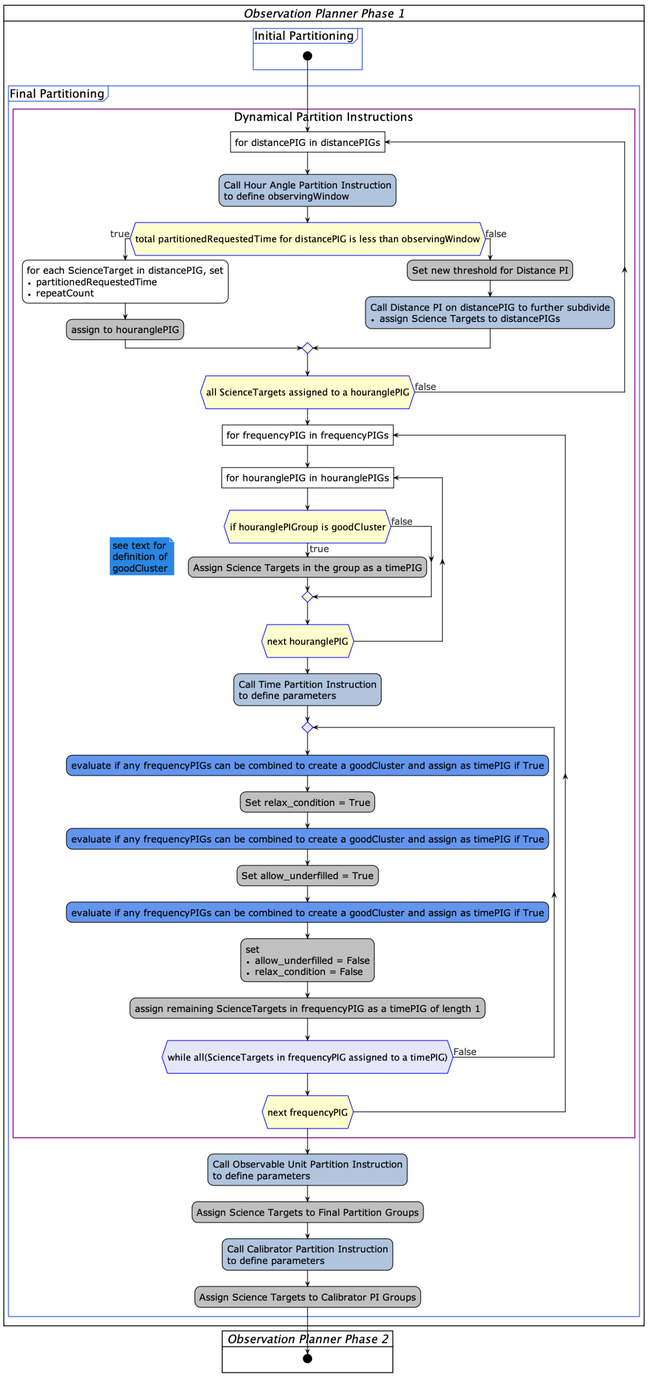



**Figure 17.3:** Final Partitioning

(b) If the total `Partition Requested Time` of the group is less than `observing window` or the `maximum duration per repeat count`, the algorithms proceeds to Point 6.

3. If a Science Target within the group has a `Partition Requested Time` that is less than the `minimum duration` of a *Subscan*, the algorithm proceeds to Point 5.

4. If a refactoring of a group is initiated, the algorithm will proceed as follows.

   (a) The `repeat count` of each Science Target in the group is set equal to the sum of the `Partition Requested Time` of the group divided by the `observing window` or the `maximum duration per repeat count`, whichever is smaller. There is one unique `repeat count` in a group; the algorithm sets the `Partition Requested Time` per Science Target such that Equation (17.0.1) is satisfied. The algorithm then reevaluates Point 2. If the total `Partition Requested Time` of the group is greater than `observing window` or the `maximum duration per repeat count`, the algorithm proceeds to Point 5.

   (b) If the total `Partition Requested Time` of the group is less than `observing window` or the `maximum duration per repeat count`, the algorithms proceeds to Point 6.

5. The algorithm calls the Distance Partition Instruction, defines a new threshold for partitioning, and reevaluates the partitioning.

   (a) Description of how it chooses new threshold.

   (b) This will overwrite the Distance PI Group index for the Science Targets.

   (c) The algorithm sets the `Partition Requested Time` for each Science Target in the group to be equal to its `Requested Time` and sets the `repeat count` per Science Target equal to 1. It

   (d) The algorithm then reevaluates Point 2, which can trigger Point 4. If the total `Partition Requested Time` of the group is still greater than `observing window` or the `maximum duration per repeat count`, the algorithm repeats Point 5. Note, this may ultimately result in a Science Target being partitioned into a group of size 1.

   (e) If the total `Partition Requested Time` of the group is less than `observing window` or the `maximum duration per repeat count`, the algorithms proceeds to Point 6.

6. The algorithm assigns the Science Targets in the group to an Hour Angle PI Group.

Once each Initial Partition Group has been iterated through such that the Science Targets are assigned to one or more Hour Angle PI Groups, the algorithm now evaluates each Hour Angle PI Group to determine if it is an optimal cluster. For each Hour Angle PI Group, evaluates the Hour Angle PI Group against the "Good Cluster" criteria in Chapter 17.

If the Hour Angle PI Group is a "Good Cluster", ...

If it is not a "Good Cluster", the algorithm attempts to refactor the partitioning in the following ways.

1. If a group's total `Partition Requested Time` is less than 90% of the `maximum duration`, the group is underfull. The algorithm will identify the Science Targets that share a common Frequency PI Group with the group[5]. The algorithm form a "test group" by adding a group that is closest in distance to the initial group. The test group is evaluated as a "Good Cluster".

   (a) If it is one, the Science Targets in the test group are assigned to a Time Partition Instruction Group.

   (b) If it is not, the test group is discarded. The algorithm will attempt to add the next closest group in distance, if it shares a common Frequency PI Group with the initial group. The algorithm will iterate on this process until either the test group is evaluated as a "Good Cluster", or the potential test groups have been exhausted. If the former, a new Initial Partition Group is evaluated. If the latter, the algorithm proceeds to Point 2.

[5] Recall that the Partition Plan sets the order of the Hierarchical PIs and that order will affect the outcome of the partitioning

2. If a group is underfull and no combination of groups that share a common Frequency PI Group are "Good Clusters", the algorithm follows a superset of rules that govern which of the constraints can be disregarded as a consequence of its state. This feature is important to intelligently assemble clusters. Consider the example of 2 good clusters and 1 severely overhead dominated cluster described earlier in this section. Specifically, the algorithm can be allowed to relax two specific criteria:

    (a) A cluster's total `Partition Requested Time` is less than or equal to a 1.2 times the `maximum duration`;

    (b) A cluster's total `Partition Requested Time` is greater than or equal to a 70% of the `maximum duration`.

    The algorithm is designed to prioritize Point 2a when assembling clusters over Point 2b. Once the algorithm allows these conditions to relax, it will reevaluate the group as described at Point 1.

3. If the algorithm has relaxed the criteria in Points 2a and 2b and not all of the Science Targets are assigned to a Time PI Group, the algorithm can assign a Science Target as its own cluster.

#### 17.1.2.2 Mapping Clusters to *Observation Specifications*

For each Time PI Group....
...write to Final Partition Group.

#### 17.1.2.3 Identifying Common Calibration Plans in a Cluster

For each Final Partition Group....
....write to Calibration Partition Instruction Group

# 18 Observation Planner Phase 2

From *Phase 1*, the *Science Target List* has been partitioned into clusters. This information is stored in the Final Partition Group. The *Phase 2* algorithm inherits the `Partition Requested Time`, the `repeat count`, and the Calibrator PI Group associated with each Science Target from *Phase 1*. For each Final Partition Group, the *Phase 2* algorithm calls the Calibration Strategies for the Science Targets associated with the group. This produces the Calibration Plans. [Posit] The Science Targets in a given Final Partition Group have similar Calibration Plans, so they will map to a single *Observation Specification*.

## 18.1 Phase 2 Algorithm

Here, we present the algorithmic implementation details of the *Phase 2* algorithm. The algorithm treats each Final Partition Group independently.

1. The algorithms calls the *Calibration Strategy* factory for each Science Target in a Final Partition Group to produce Calibration Plans (Chapter 9.2).

# 19 | Observation Planner Phase 3

Scan Sequencing Under Construction.

# Part V

# Auxiliary Proposal Creation Algorithms and Documentation

# 20 Specification Constraints

| Table 3.1: Cycle times in minutes by configuration and frequency band | | | | |
|---|---|---|---|---|
| Band (Frequency Range) | Array Configuration and Cycle Time in Minutes | | | |
| | A | B | C | D |
| 4 (54-86 MHz) | 30 | 30 | 30 | 30 |
| P (224-480 MHz) | 30 | 30 | 30 | 30 |
| L (1-2 GHz) | 15 | 15 | 15 | 15 |
| S (2-4 GHz) | 15 | 15 | 15 | 15 |
| C (4-8 GHz) | 8 | 10 | 10 | 10 |
| X (8-12 GHz) | 8 | 10 | 10 | 10 |
| Ku (12-18 GHz) | 6 | 7 | 8 | 8 |
| K (18-26.5 GHz) | 4 | 5 | 6 | 6 |
| Ka (26.5-40 GHz) | 3 | 4 | 5 | 6 |
| Q (40-50 GHz) | 2 | 3 | 4 | 5 |

**Figure 20.1:** Cycle Times in Minutes by Configuration and Band

**Table 20.1:** Placeholder Table of Cycle Times

| Band | Receiver | Maximum Cycle Time |
|---|---|---|

# 21 Calculations

## 21.1 Setup Time

The `Setup Time` is time spent not collecting data. This includes the time it takes for an antenna to slew between one source to another (ordered), the time it takes for the antenna to settle to a nominal state to resume collecting data (settle time), and the time it takes for any changes in the HARDWARE CONFIGURATIONbetween the previous observing state and the next observing state to take affect (`Hardware Configuration Overhead`).

- Antenna Slew Calculation (VLA, GBT...)
- Table of settle times
- Table of `Hardware Configuration Overhead`s

## 21.2 Sensitivity Calculators

### 21.2.1 VLA Sensitivity Calculator

## 21.3 Catchall: Needs to be written

- How to select an appropriate Phase Calibrator (Calibration Strategy)
- Select Calibrators in General

## 21.4 Antenna Motion

Toy code exists but needs written up.

# 22 Partitioning

## 22.1 Example of Initial Partitioning

In Initial Partitioning, each Hierarchical Partition Instruction (PI) is applied to previously partitioned groups of Science Targets independently, so the groups may be even furthered broken down as the partitioning progresses. In this section, we present uses cases to illustrate this fragmentation.

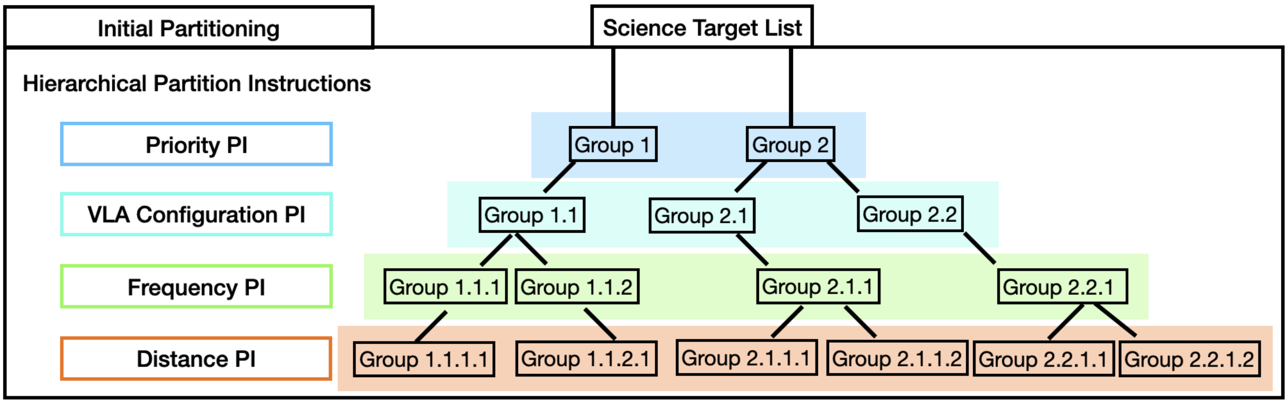



**Figure 22.1:** Example Initial Partitioning for two SOURCES and two HARDWARE CONFIGURATIONS.

Figure 22.1 shows how a *Science Target List* for a VLA *Capability* may fragment depending on the details of the Hierarchical PIs. The *Capability* defines the custom metrics and thresholds for the PIs.

- The Priority PI partitions the *Science Target List* into two groups: Group 1 and 2.
- The VLA Configuration PI
  - does not result in partitioning Group 1 further but records the implementation of the PI by labeling the group as Group 1.1;
  - partitions Group 2 into two groups: Group 2.1 and 2.2.
- The Frequency PI
  - partitions Group 1.1 into two groups: Group 1.1.1 and Group 1.1.2;
  - does not result in further partitioning of Group 2.1 but records the implementation of the PI by labeling the group as Group 2.1.1;
  - does not result in further partitioning of Group 2.2 but records the implementation of the PI by labeling the group as Group 2.2.1.
- The Distance PI
  - does not result in further partitioning of Group 1.1.1 but records the implementation of the PI by labeling the group as Group 1.1.1.1;
  - does not result in further partitioning of Group 1.1.2 but records the implementation of the PI by labeling the group as Group 1.1.2.1;
  - partitions Group 2.1.1 into two groups: Group 2.1.1.1 and Group 2.1.1.2;
  - partitions Group 2.2.1 into two groups: Group 2.2.1.1 and Group 2.2.1.2.

At the end of Initial Partitioning, there are 6 groups of Science Targets, and Final Partitioning begins.

## 22.2 Comparison of Partitioning Algorithm to Archival Observations

> **Note**
>
> This section is unfinished.

The partitioning algorithm in Section 17.1.1 is built to be deterministic, to prioritize *Facility* specific best observing practices, and to reflect observing practices that can deliver usable science data. One easy measure of the success of the partitioning algorithm is to compare the contents of existing Execution Blocks[1] (EBs) to the ones generated by the algorithm. In this section, we present a comparison of clusters created by users, which are available in the NRAO's Science Archive, and the clusters that the algorithm creates from the same proposal.

The Science Archive contains the proposal meta and observing data needed to reconstruct Science Targets and a *Science Target List* for each proposal. Per *Scan*, this includes the name of target, the coordinate information, the minimum and maximum frequency associated with the *Scan*, the VLA Configuration (if applicable), the date and time of the*Scan*, the `Scan Intent`, the `Subscan Intent(s)`, and the `duration` of the *Scan* and *Subscan(s)*. We selected from the archive highly ranked GBT, VLA, and VLBA proposals between observing semesters 19B and 20B. There were 241 GBT, 186 VLA, and 33 VLBA proposals in this selection for a total of 460 proposals. The `total duration` of a proposal was calculated by summing the durations of the EBs, and the Overhead was calculated by subtracting the sum of the `Requested Time` per Science Target from the `total duration`.

We reversed engineered Observing Targets for each proposal that consists of a SOURCE, HARDWARE CONFIGURATION, and `Requested Time`. Through coordinate matching and matching the name of the target, we defined SOURCES. As proposals can have one to several hundred (in cases of surveys) sources, this process is automated and we did not attempt to identify observing that included mosaics. We discuss this later and how it may affect our analysis. The minimum and maximum frequency associated with each *Scan* were used to define a center frequency and frequency range, which were mapped to the frequency bands of a *Facility*. Together with the FONT-END, which is also listed per *Scan* in the archive data, and the Array Configuration (if applicable), we defined HARDWARE CONFIGURATIONS and paired with them with the source information.

Using the `Scan Intent`, `Subscan Intent`, `Scan Duration`, and `Subscan Duration`, we estimated a total integration time per SOURCE and HARDWARE CONFIGURATION pair by summing up the *Subscans* with `Subscan Intents` of ON_SOURCE for *Scans* with the `Scan Intent` of OBSERVE_TARGET. The `Scan Intents` and `Subscan Intents` also differentiate between *Scans* for calibration and for science data acquisition in most cases, so we could identify the Science Targets in the list of Observing Targets. For Science Targets, the integration time was then treated as the `Requested Time`.

The *Science Target List* then consists the Science Targets and *Calibration* and *Scheduling Strategies* factories. We use this *Science Target List* as the input for the Initial Partitioning in the *Observation Planner Phase 1*. It is important to note that neither the *Science Target List* nor the algorithm has prior information about of how the user specified the clustering. This *Science Target List* is identical in the information it contains to one generated by the *Observing Strategy*, which requires information from the *Capability Request*(s).

In Initial Partitioning, the parameters of the individual Partition Instructions are dynamically generated by the *Calibration Strategy* factory for the Science Target and the set of Science Targets under consideration; their behavior is described in the relevant sections of Chapter 9 for each *Facility*. After Initial Partitioning, the first of the three steps of Final Partitioning was invoked. The first step enforces the definition of what constitutes an optimal cluster (§ 17.1.2.1). It is not the final mapping of clusters to *Observation Specifications*, but it is designed to be relatively close to the final mapping. For the purposes of this comparison, the state of partitioning at this point is considered the final state of the clusters, and we refer to the clusters that were generated by the algorithm as "modeled".

[1] Not all of the Execution Blocks for GBT proposals are available in the Science Archive. We mined the GBT archive directly to extract the necessary data when needed.

The user defined clusters can be reconstructed from the EBs. We inferred how the user intended to cluster their Science Targets by their membership in the EBs. We assign a common cluster ID to the Science Targets are members of a single EB. A set of Science Targets can only have one cluster ID. Generally, we find that users did not intermingle sources across EBs. This is likely because the Scheduling Blocks were created with a source group and a `Repeat Count` was set so that the same *Scan List* was executed multiple times. This is a standard mode of observing, particularly for the VLA. This is not always the case, and we discuss the consequences of this later. In the few cases where there are multiple cluster IDs that can be assigned to a Science Target, we assign the cluster ID that is associated with the cluster the source more commonly belongs to. Once the cluster IDs are assigned, we also determine a `Repeat Count` per cluster. We refer to these clusters, which are intended to reflect the user's intent, as "reconstructed".

The `total time per proposal` can be calculated for the reconstructed and modeled proposals. For the reconstructed proposals, this is relatively straightforward and is the sum of the durations of the EBs. For the modeled proposals, there is an additional step needed to arrive at a `total time per proposal`. At this stage of development, the algorithm does not provide a `total time per proposal` yet; it can only estimate a `Requested Time` per Science Target and thus a total `Requested Time` per proposal. This is similar to the raw output of existing Sensitivity Calculators; these calculators use a scalar to roughly transform the total `Requested Time` to a total time; we refer to this scalar as the `overhead factor`. We employ the same `overhead factors` (temporarily) to arrive at a `total time per proposal`. The values of the `overhead factor`, which are *Capability* and HARDWARE CONFIGURATION dependent, are summarized in Table 22.1.

Figures 22.2a, 22.2b, and 22.2c show the `total time per proposal` for the GBT, VLA, and VLBA, respectively, and Figure 22.2d shows all the *Facilities* in one plot. Each figure compares the reconstructed `total time per proposal` against the modeled `total time per proposal`. Generally, the reconstructed and modeled total time are similar, which indicates that the algorithm and the constraints on it can reasonably reproduce what users actually request.

There are notable disagreements though. The glaring ones are

- the systematic offset in Figure 22.2a;
- the trend at shorter durations for the algorithm to give more time to a proposal than the user did;
- the trend at longer durations for the algorithm to give less time to a proposal than the user did.

These three cases generally reflect how we calculate `total time per proposal` in the algorithm. As noted above, we estimate the total time by multiplying the sum of the `Requested Times` by an `overhead factor`. This scalar value is, at times, a poor approach to estimate the total time. Particularly for GBT proposals, a different value than what we have specified for the scalar is needed entirely for some proposals.

We can check this interpretation by calculating the total time from the reconstructed *Scan List* in a different manner. Currently, the reconstructed `total time per proposal` is estimated from the timestamps of the *Scan Lists*. This is the `total time per proposal` presented in Figures 22.2a – 22.2d. However, we can calculate a total time by summing up the `Requested Time` from the reconstructed *Scan Lists* and multiplying by a `overhead factor`. This is similar to how we calculate the total time for the modeled proposals.

Figure 22.2e compares this new approach to determining the total time for the reconstructed proposals against the original way. Generally, the short duration proposals require more time than the `overhead factor` accounts for, and the long duration proposals need less time than the `overhead factor` would give, though the affect is not as prominent. For the VLA in particular, this behavior is understood and explicitly called out in documentation for observing, so it is unsurprising to see it manifest here.

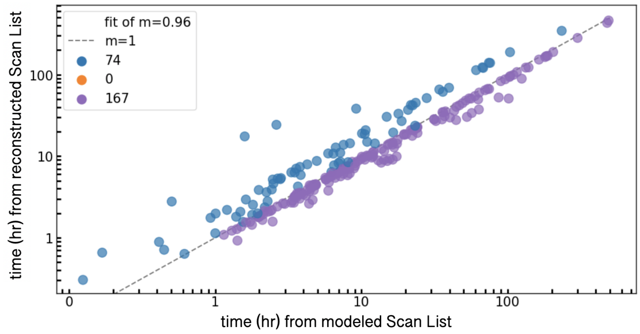
Figure 22.2f compares the new approach to determining the total time for the reconstructed proposals against the modeled `total time per proposal`. The reduction of scatter from the case of perfect agreement supports of interpretation of a poorly performing `overhead factor` being a significant driver of the disagreement. This is especially true for shorter duration proposals.

The remaining inconsistencies require a different explanation. It is important to understand these cases for a few reasons. First, we would like to understand the behavior of the algorithm and modify it as needed to produce reasonable clusters. Second, the parameters constraining the algorithm may be the driver of poor results, and we may need additional input from the user in the *Capability Request* to fully specify the request. Thus, we investigated the algorithm's results on a per proposal basis to determine what caused the algorithm to produce different results than that of the user.

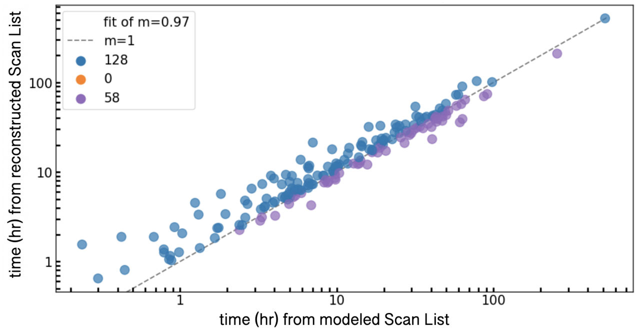
Generally, we found that the algorithm has trouble reproducing a few specific observing styles/-modes.

- Large surveys that have hundreds of sources scattered across the sky but with small `Requested Times` associated with each source are difficult to reproduce with the current constraints (e.g., `maximum duration` and limits to overhead). After adjusting these constraints, the algorithm can reproduce a reasonable set of *Observation Specifications*. This indicates that we may need to provide the user a way to specify this kind of observing scheme in the *Capability Request*, which would trigger a different set of constraints on the algorithm than we currently implement.
- Observations of pulsars are not very well reproduced by the algorithm yet. We have a mechanism in the *Capability Request* to capture this intent from the user. We need to iterate on the constraints in this case to improve the output of the algorithm.
- Observations that have mosaic or on-the-fly schema tend to be poorly reproduced by the algorithm sometimes. This may also be a consequence of the limited amount of data we have in reconstructing the *Science Target List*, as it is not always an obvious property of a proposal from a *Scan List*. When we create *Science Target List* directly from the *Capability Request*, this case may disappear as we have a mechanism to capture the need for mapping already.

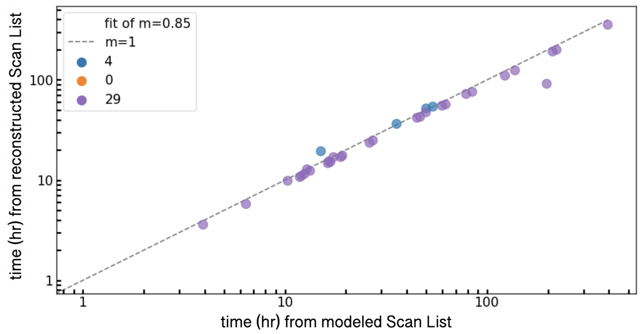
**Table 22.1:** `overhead factors` per *Facility*



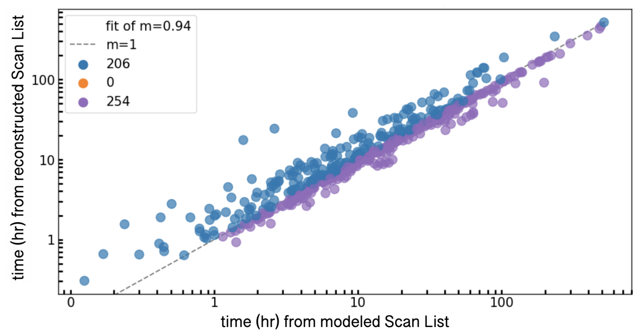

**(a)** GBT: Total Time per Proposal



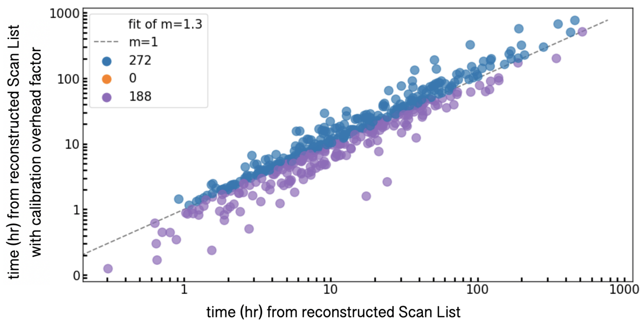

**(b)** VLA: Total Time per Proposal



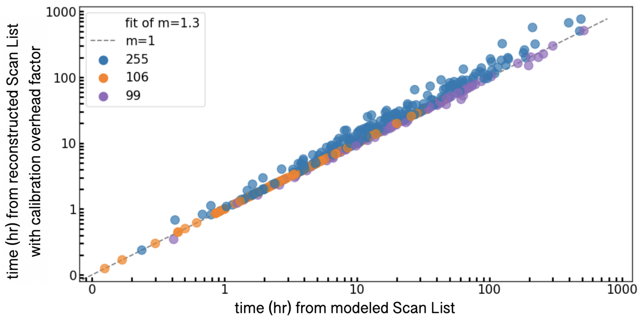

**(c)** VLBA: Total Time per Proposal




**(d)** Total Time per Proposal for all *Facilities*




**(e)**




**(f)**

**Figure 22.2:** **(a)** – **(c)**: Comparison per *Facility* of the `total time per proposal` from the reconstructed *Scan Lists* (ordinate) and from the modeled *Scan List* (abscissa). The dashed line represents the case of perfect agreement. The colors of the symbols illustrate that side of the perfect agreement the datum is on: blue for Y > X, purple for Y < X, and orange for Y = X. The legend gives the slope of the best fit line. **(d)** Comparison of the `total time per proposal` for all the *Facilities*. **(f)** The ordinate is the `total time per proposal` as determined by the sum of reconstructed compRequested Times per Science Target × `overhead factor` (see the text for a further discussion). The abscissa is the `total time per proposal` from the reconstructed *Scan List*. **(g)** The ordinate is the `total time per proposal` as determined by the sum of reconstructed compRequested Times per Science Target × `overhead factor`. The abscissa is the `total time per proposal` from the modeled *Scan List*.

# 23 Motivating Best Practices

## 23.1 Best Practices for the GBT

Under Construction.

## 23.2 Best Practices for the VLA

Here, we motivate and describe the underlying assumptions and practices built into the Partition Plan and Calibration Plan. Under Construction.

## 23.3 Best Practices for the VLBA

Under Construction.

## 23.4 Best Practices for Partitioning (Observation Planner Phase 1)

Under Construction.

# 24 | Pointing Patterns

## 24.1 | GBT Pointing Patterns

Under Construction.

## 24.2 | VLA Pointing Patterns

Under Construction.

# 25 Capability Parameter Specification Inputs

**Table 25.1:** Reference for Input and Unit Groups in Table 4.1.

| Name | Expanded Inputs |
|---|---|
| Coordinate-1 | ICRS, Galactic, Horizontal |
| Angle-1 | degree, hour |
| Angle-2 | degree |
| Angle-3 | degree, arcminute, arcsecond |
| Shape-1 | point, ellipse, rectangle |
| Velocity-1 | km $s^{-1}$ |
| Velocity-2 | LSRK |
| Doppler-1 | Radio, Optical, Redshift |
| Parallax-1 | mas |
| PM-1 | mas $yr^{-1}$ |
| FluxDensity-1 | mK, K, Jy, mJy |
| Frequency-1 | GHz, MHz, kHz, Hz |
| Frequency-2 | GHz, MHz, kHz, Hz, km $s^{-1}$ |
| Intensity-1 | mK, K, mJy, $\mu$Jy, Jy |
| Intensity-2 | mK $beam^{-1}$, K $beam^{-1}$, $mJybeam^{-1}$, $\mu Jybeam^{-1}$, $Jybeam^{-1}$ |

**Table 25.2:** GBT Continuum FIELD SOURCE.

| *Capability Parameter Specification* | Default Input | Table 25.1 Reference |
|---|---|---|
| Name | N/A | N/A |
| Coordinate System | ICRS | Coordinate-1 |
| Position Longitude | 0 degree | Angle-1 |
| Position Latitude | 0 degree | Angle-2 |
| Field of View Shape | point | Shape-1 |
| FOV Longitude | 0 degree | Angle-3 |
| FOV Latitude | 0 degree | Angle-3 |
| Radial Velocity | 0 km $s^{-1}$ | Velocity-1 |
| Velocity Reference Frame | LSRK | Velocity-2 |
| Doppler Type | Radio | Doppler-1 |
| Parallax | 0 mas | Parallax-1 |
| Proper Motion Longitude | 0 mas $yr^{-1}$ | PM-1 |
| Proper Motion Latitude | 0 mas $yr^{-1}$ | PM-1 |
| Peak Continuum Flux Density | 0 Jy | FluxDensity-1 |

**Table 25.3:** GBT Continuum SPECTRAL SPECIFICATION.

| *Capability Parameter Specification* | Default Input | Table 25.1 Reference |
|---|---|---|
| Name | N/A | N/A |
| Center Frequency | 0 GHz | Frequency-1 |
| Bandwidth | 0 km $s^{-1}$ | Frequency-2 |
| Spectral Resolution | 0 km $s^{-1}$ | Frequency-2 |

**Table 25.4:** GBT Continuum PERFORMANCE PARAMETERS.

| *Capability Parameter Specification* | Default Input | Table 25.1 Reference |
|---|---|---|
| RMS Sensitivity | 0 K | Intensity-1 |

**Table 25.5:** GBT Continuum CALIBRATION PARAMETERS.

| *Capability Parameter Specification* | Boolean Default |
|---|---|
| Flux Density | False |
| Test Source | False |
| Polarization | False |

**Table 25.6:** GBT Pulsar FIELD SOURCE.

| *Capability Parameter Specification* | Default Input | Table 25.1 Reference |
|---|---|---|
| Name | N/A | N/A |
| Coordinate System | ICRS | Coordinate-1 |
| Position Longitude | 0 degree | Angle-1 |
| Position Latitude | 0 degree | Angle-2 |
| Field of View Shape | point | Shape-1 |
| FOV Longitude | 0 degree | Angle-3 |
| FOV Latitude | 0 degree | Angle-3 |
| Radial Velocity | 0 km $s^{-1}$ | Velocity-1 |
| Velocity Reference Frame | LSRK | Velocity-2 |
| Doppler Type | Radio | Doppler-1 |
| Parallax | 0 mas | Parallax-1 |
| Proper Motion Longitude | 0 mas $yr^{-1}$ | PM-1 |
| Proper Motion Latitude | 0 mas $yr^{-1}$ | PM-1 |
| Peak Continuum Flux Density | 0 Jy | FluxDensity-1 |

**Table 25.7:** GBT Pulsar SPECTRAL SPECIFICATION.

| *Capability Parameter Specification* | Default Input | Table 25.1 Reference |
|---|---|---|
| Name | N/A | N/A |
| Center Frequency | 0 GHz | Frequency-1 |
| Bandwidth | 0 km $s^{-1}$ | Frequency-2 |
| Spectral Resolution | 0 km $s^{-1}$ | Frequency-2 |

**Table 25.8:** GBT Pulsar PERFORMANCE PARAMETERS.

| *Capability Parameter Specification* | Default Input | Table 25.1 Reference |
|---|---|---|
| RMS Sensitivity | 0 K | Intensity-1 |

**Table 25.9:** GBT Pulsar CALIBRATION PARAMETERS.

| *Capability Parameter Specification* | Boolean Default |
|---|---|
| Flux Density | False |
| Test Source | False |
| Polarization | False |

**Table 25.10:** GBT Radar FIELD SOURCE.

| *Capability Parameter Specification* | Default Input | Table 25.1 Reference |
|---|---|---|
| Name | N/A | N/A |
| Coordinate System | ICRS | Coordinate-1 |
| Position Longitude | 0 degree | Angle-1 |
| Position Latitude | 0 degree | Angle-2 |
| Field of View Shape | point | Shape-1 |
| FOV Longitude | 0 degree | Angle-3 |
| FOV Latitude | 0 degree | Angle-3 |
| Radial Velocity | 0 km $s^{-1}$ | Velocity-1 |
| Velocity Reference Frame | LSRK | Velocity-2 |
| Doppler Type | Radio | Doppler-1 |
| Parallax | 0 mas | Parallax-1 |
| Proper Motion Longitude | 0 mas $yr^{-1}$ | PM-1 |
| Proper Motion Latitude | 0 mas $yr^{-1}$ | PM-1 |
| Peak Continuum Flux Density | 0 Jy | FluxDensity-1 |

**Table 25.11:** GBT Radar SPECTRAL SPECIFICATION.

| *Capability Parameter Specification* | Default Input | Table 25.1 Reference |
|---|---|---|
| Name | N/A | N/A |
| Center Frequency | 0 GHz | Frequency-1 |
| Bandwidth | 0 km $s^{-1}$ | Frequency-2 |
| Spectral Resolution | 0 km $s^{-1}$ | Frequency-2 |

**Table 25.12:** GBT Radar PERFORMANCE PARAMETERS.

| *Capability Parameter Specification* | Default Input | Table 25.1 Reference |
|---|---|---|
| RMS Sensitivity | 0 K | Intensity-1 |

**Table 25.13:** GBT Radar CALIBRATION PARAMETERS.

| *Capability Parameter Specification* | Boolean Default |
|---|---|
| Flux Density | False |
| Test Source | False |
| Polarization | False |

**Table 25.14:** GBT Spectral Line FIELD SOURCE.

| *Capability Parameter Specification* | Default Input | Table 25.1 Reference |
|---|---|---|
| Name | N/A | N/A |
| Coordinate System | ICRS | Coordinate-1 |
| Position Longitude | 0 degree | Angle-1 |
| Position Latitude | 0 degree | Angle-2 |
| Field of View Shape | point | Shape-1 |
| FOV Longitude | 0 degree | Angle-3 |
| FOV Latitude | 0 degree | Angle-3 |
| Radial Velocity | 0 km $s^{-1}$ | Velocity-1 |
| Velocity Reference Frame | LSRK | Velocity-2 |
| Doppler Type | Radio | Doppler-1 |
| Parallax | 0 mas | Parallax-1 |
| Proper Motion Longitude | 0 mas $yr^{-1}$ | PM-1 |
| Proper Motion Latitude | 0 mas $yr^{-1}$ | PM-1 |
| Peak Continuum Flux Density | 0 Jy | FluxDensity-1 |
| Peak Line Flux Density | 0 K | FluxDensity-1 |
| Line Width | 0 km $s^{-1}$ | Velocity-1 |

**Table 25.15:** GBT Spectral Line SPECTRAL SPECIFICATION.

| *Capability Parameter Specification* | Default Input | Table 25.1 Reference |
|---|---|---|
| Name | N/A | N/A |
| Center Frequency | 0 GHz | Frequency-1 |
| Bandwidth | 0 km $s^{-1}$ | Frequency-2 |
| Spectral Resolution | 0 km $s^{-1}$ | Frequency-2 |

**Table 25.16:** GBT Spectral Line PERFORMANCE PARAMETERS.

| *Capability Parameter Specification* | Default Input | Table 25.1 Reference |
|---|---|---|
| RMS Sensitivity | 0 K | Intensity-1 |

**Table 25.17:** GBT Spectral Line CALIBRATION PARAMETERS.

| *Capability Parameter Specification* | Boolean Default |
|---|---|
| Flux Density | False |
| Test Source | False |
| Polarization | False |

**Table 25.18:** VLA Continuum FIELD SOURCE.

| *Capability Parameter Specification* | Default Input | Table 25.1 Reference |
|---|---|---|
| Name | N/A | N/A |
| Coordinate System | ICRS | Coordinate-1 |
| Position Longitude | 0 degree | Angle-1 |
| Position Latitude | 0 degree | Angle-2 |
| Field of View Shape | point | Shape-1 |
| FOV Longitude | 0 degree | Angle-3 |
| FOV Latitude | 0 degree | Angle-3 |
| Radial Velocity | 0 km $s^{-1}$ | Velocity-1 |
| Velocity Reference Frame | LSRK | Velocity-2 |
| Doppler Type | Radio | Doppler-1 |
| Parallax | 0 mas | Parallax-1 |
| Proper Motion Longitude | 0 mas $yr^{-1}$ | PM-1 |
| Proper Motion Latitude | 0 mas $yr^{-1}$ | PM-1 |
| Peak Continuum Flux Density | 0 Jy | FluxDensity-1 |

**Table 25.19:** VLA Continuum SPECTRAL SPECIFICATION.

| *Capability Parameter Specification* | Default Input | Table 25.1 Reference |
|---|---|---|
| Name | N/A | N/A |
| Center Frequency | 0 GHz | Frequency-1 |
| Bandwidth | 0 GHz | Frequency-2 |
| Spectral Resolution | 0 km $s^{-1}$ | Frequency-2 |

**Table 25.20:** VLA Continuum PERFORMANCE PARAMETERS.

| *Capability Parameter Specification* | Default Input | Table 25.1 Reference |
|---|---|---|
| Largest Angular Scale | 0 degree | Angle-3 |
| Angular Resolution | 0 degree | Angle-3 |
| RMS Sensitivity | 0 mJy $beam^{-1}$ | Intensity-1 |
| Dynamic Range | 10:1 | N/A |

**Table 25.21:** VLA Continuum CALIBRATION PARAMETERS.

| *Capability Parameter Specification* | Boolean Default |
|---|---|
| Polarization | False |

**Table 25.22:** VLA Pulsar FIELD SOURCE.

| *Capability Parameter Specification* | Default Input | Table 25.1 Reference |
|---|---|---|
| Name | N/A | N/A |
| Coordinate System | ICRS | Coordinate-1 |
| Position Longitude | 0 degree | Angle-1 |
| Position Latitude | 0 degree | Angle-2 |
| Field of View Shape | point | Shape-1 |
| FOV Longitude | 0 degree | Angle-3 |
| FOV Latitude | 0 degree | Angle-3 |
| Radial Velocity | 0 km $s^{-1}$ | Velocity-1 |
| Velocity Reference Frame | LSRK | Velocity-2 |
| Doppler Type | Radio | Doppler-1 |
| Parallax | 0 mas | Parallax-1 |
| Proper Motion Longitude | 0 mas $yr^{-1}$ | PM-1 |
| Proper Motion Latitude | 0 mas $yr^{-1}$ | PM-1 |
| Peak Continuum Flux Density | 0 Jy | FluxDensity-1 |

**Table 25.23:** VLA Pulsar SPECTRAL SPECIFICATION.

| *Capability Parameter Specification* | Default Input | Table 25.1 Reference |
|---|---|---|
| Name | N/A | N/A |
| Center Frequency | 0 GHz | Frequency-1 |
| Bandwidth | 0 GHz | Frequency-2 |
| Spectral Resolution | 0 km $s^{-1}$ | Frequency-2 |

**Table 25.24:** VLA Pulsar PERFORMANCE PARAMETERS.

| *Capability Parameter Specification* | Default Input | Table 25.1 Reference |
|---|---|---|
| Largest Angular Scale | 0 degree | Angle-3 |
| Angular Resolution | 0 degree | Angle-3 |
| RMS Sensitivity | 0 mJy $beam^{-1}$ | Intensity-1 |
| Dynamic Range | 10:1 | N/A |

**Table 25.25:** VLA Pulsar CALIBRATION PARAMETERS.

| *Capability Parameter Specification* | Boolean Default |
|---|---|
| Polarization | False |

**Table 25.26:** VLA Spectral Line FIELD SOURCE.

| *Capability Parameter Specification* | Default Input | Table 25.1 Reference |
|---|---|---|
| Name | N/A | N/A |
| Coordinate System | ICRS | Coordinate-1 |
| Position Longitude | 0 degree | Angle-1 |
| Position Latitude | 0 degree | Angle-2 |
| Field of View Shape | point | Shape-1 |
| FOV Longitude | 0 degree | Angle-3 |
| FOV Latitude | 0 degree | Angle-3 |
| Radial Velocity | 0 km $s^{-1}$ | Velocity-1 |
| Velocity Reference Frame | LSRK | Velocity-2 |
| Doppler Type | Radio | Doppler-1 |
| Parallax | 0 mas | Parallax-1 |
| Proper Motion Longitude | 0 mas $yr^{-1}$ | PM-1 |
| Proper Motion Latitude | 0 mas $yr^{-1}$ | PM-1 |
| Peak Continuum Flux Density | 0 Jy | FluxDensity-1 |
| Peak Line Flux Density | 0 K | FluxDensity-1 |
| Line Width | 0 km $s^{-1}$ | Velocity-1 |

**Table 25.27:** VLA Spectral Line SPECTRAL SPECIFICATION.

| *Capability Parameter Specification* | Default Input | Table 25.1 Reference |
|---|---|---|
| Name | N/A | N/A |
| Center Frequency | 0 GHz | Frequency-1 |
| Bandwidth | 0 GHz | Frequency-2 |
| Spectral Resolution | 0 km $s^{-1}$ | Frequency-2 |

**Table 25.28:** VLA Spectral Line PERFORMANCE PARAMETERS.

| *Capability Parameter Specification* | Default Input | Table 25.1 Reference |
|---|---|---|
| Largest Angular Scale | 0 degree | Angle-3 |
| Angular Resolution | 0 degree | Angle-3 |
| RMS Sensitivity | 0 mJy $beam^{-1}$ | Intensity-1 |
| Dynamic Range | 10:1 | N/A |

**Table 25.29:** VLA Spectral Line CALIBRATION PARAMETERS.

| *Capability Parameter Specification* | Boolean Default |
|---|---|
| Polarization | False |

**Table 25.30:** VLBA Continuum FIELD SOURCE.

| *Capability Parameter Specification* | Default Input | Table 25.1 Reference |
|---|---|---|
| Name | N/A | N/A |
| Coordinate System | ICRS | Coordinate-1 |
| Position Longitude | 0 degree | Angle-1 |
| Position Latitude | 0 degree | Angle-2 |
| Field of View Shape | point | Shape-1 |
| FOV Longitude | 0 degree | Angle-3 |
| FOV Latitude | 0 degree | Angle-3 |
| Radial Velocity | 0 km $s^{-1}$ | Velocity-1 |
| Velocity Reference Frame | LSRK | Velocity-2 |
| Doppler Type | Radio | Doppler-1 |
| Parallax | 0 mas | Parallax-1 |
| Proper Motion Longitude | 0 mas $yr^{-1}$ | PM-1 |
| Proper Motion Latitude | 0 mas $yr^{-1}$ | PM-1 |
| Peak Continuum Flux Density | 0 Jy | FluxDensity-1 |

**Table 25.31:** VLBA Continuum SPECTRAL SPECIFICATION.

| *Capability Parameter Specification* | Default Input | Table 25.1 Reference |
|---|---|---|
| Name | N/A | N/A |
| Center Frequency | 0 GHz | Frequency-1 |
| Bandwidth | 0 GHz | Frequency-2 |
| Spectral Resolution | 0 km $s^{-1}$ | Frequency-2 |

**Table 25.32:** VLBA Continuum PERFORMANCE PARAMETERS.

| *Capability Parameter Specification* | Default Input | Table 25.1 Reference |
|---|---|---|
| Largest Angular Scale | 0 degree | Angle-3 |
| Angular Resolution | 0 degree | Angle-3 |
| RMS Sensitivity | 0 mJy $beam^{-1}$ | Intensity-1 |
| Dynamic Range | 10:1 | N/A |

**Table 25.33:** VLBA Continuum CALIBRATION PARAMETERS.

| *Capability Parameter Specification* | Boolean Default |
|---|---|
| Polarization | False |

**Table 25.34:** VLBA Pulsar FIELD SOURCE.

| *Capability Parameter Specification* | Default Input | Table 25.1 Reference |
|---|---|---|
| Name | N/A | N/A |
| Coordinate System | ICRS | Coordinate-1 |
| Position Longitude | 0 degree | Angle-1 |
| Position Latitude | 0 degree | Angle-2 |
| Field of View Shape | point | Shape-1 |
| FOV Longitude | 0 degree | Angle-3 |
| FOV Latitude | 0 degree | Angle-3 |
| Radial Velocity | 0 km $s^{-1}$ | Velocity-1 |
| Velocity Reference Frame | LSRK | Velocity-2 |
| Doppler Type | Radio | Doppler-1 |
| Parallax | 0 mas | Parallax-1 |
| Proper Motion Longitude | 0 mas $yr^{-1}$ | PM-1 |
| Proper Motion Latitude | 0 mas $yr^{-1}$ | PM-1 |
| Peak Continuum Flux Density | 0 Jy | FluxDensity-1 |

**Table 25.35:** VLBA Pulsar SPECTRAL SPECIFICATION.

| *Capability Parameter Specification* | Default Input | Table 25.1 Reference |
|---|---|---|
| Name | N/A | N/A |
| Center Frequency | 0 GHz | Frequency-1 |
| Bandwidth | 0 GHz | Frequency-2 |
| Spectral Resolution | 0 km $s^{-1}$ | Frequency-2 |

**Table 25.36:** VLBA Pulsar PERFORMANCE PARAMETERS.

| *Capability Parameter Specification* | Default Input | Table 25.1 Reference |
|---|---|---|
| Largest Angular Scale | 0 degree | Angle-3 |
| Angular Resolution | 0 degree | Angle-3 |
| RMS Sensitivity | 0 mJy $beam^{-1}$ | Intensity-1 |
| Dynamic Range | 10:1 | N/A |

**Table 25.37:** VLBA Pulsar CALIBRATION PARAMETERS.

| *Capability Parameter Specification* | Boolean Default |
|---|---|
| Polarization | False |

**Table 25.38:** VLBA Spectral Line FIELD SOURCE.

| *Capability Parameter Specification* | Default Input | Table 25.1 Reference |
|---|---|---|
| Name | N/A | N/A |
| Coordinate System | ICRS | Coordinate-1 |
| Position Longitude | 0 degree | Angle-1 |
| Position Latitude | 0 degree | Angle-2 |
| Field of View Shape | point | Shape-1 |
| FOV Longitude | 0 degree | Angle-3 |
| FOV Latitude | 0 degree | Angle-3 |
| Radial Velocity | 0 km $s^{-1}$ | Velocity-1 |
| Velocity Reference Frame | LSRK | Velocity-2 |
| Doppler Type | Radio | Doppler-1 |
| Parallax | 0 mas | Parallax-1 |
| Proper Motion Longitude | 0 mas $yr^{-1}$ | PM-1 |
| Proper Motion Latitude | 0 mas $yr^{-1}$ | PM-1 |
| Peak Continuum Flux Density | 0 Jy | FluxDensity-1 |
| Peak Line Flux Density | 0 K | FluxDensity-1 |
| Line Width | 0 km $s^{-1}$ | Velocity-1 |

**Table 25.39:** VLBA Spectral Line SPECTRAL SPECIFICATION.

| *Capability Parameter Specification* | Default Input | Table 25.1 Reference |
|---|---|---|
| Name | N/A | N/A |
| Center Frequency | 0 GHz | Frequency-1 |
| Bandwidth | 0 GHz | Frequency-2 |
| Spectral Resolution | 0 km $s^{-1}$ | Frequency-2 |

**Table 25.40:** VLBA Spectral Line PERFORMANCE PARAMETERS.

| *Capability Parameter Specification* | Default Input | Table 25.1 Reference |
|---|---|---|
| Largest Angular Scale | 0 degree | Angle-3 |
| Angular Resolution | 0 degree | Angle-3 |
| RMS Sensitivity | 0 mJy $beam^{-1}$ | Intensity-1 |
| Dynamic Range | 10:1 | N/A |

**Table 25.41:** VLBA Spectral Line CALIBRATION PARAMETERS.

| *Capability Parameter Specification* | Boolean Default |
|---|---|
| Polarization | False |

# 26 Examples and Use Cases

## 26.1 Use Cases of Concepts and Definitions in the Observation Planner

### 26.1.1 (VLA) A Faraday Rotation Study of the Stellar Bubble and HII Region Associated with the W4 Complex

Consider use case 3.1 from 688-TTAT-xxx-MGMT TTA Use Cases v0.1, which describes VLA observation of two FIELD SOURCES, I1 and I5, at two SPECTRAL SPECIFICATIONS. The *Observing Strategy* determined the HARDWARE CONFIGURATIONS are C-band (5 GHz) and X-band (9 GHz) in C-Configuration and that both FIELD SOURCES are described by the `Pointing Pattern` of Single Pointing. The *Observing Strategy* creates the *Science Target List* which contains four Science Targets[1,2]: $I1_{5GHz}$, $I1_{9GHz}$, $I5_{5GHz}$, and $I5_{9GHz}$.

The *Capability* selects the appropriate *Calibration Strategy* and the *Scheduling Strategy* for the *Science Target List*, which dictate to the *Observation Planner Phase 1*, 2, and 3 how to partition, calibrate, and schedule the observations. For this example *Science Target List*, the *Observation Planner* determines that all of the Science Targets will be in one *Observation Specification* and that there will be Complex Gain (Phase Referencing), Flux Density, and Bandpass calibrations. The following paragraphs describe the contents of the *Scan List* the *Observation Planner* creates for this use case. Tables 26.1 and 26.2 present the final *Scan List* and summary of the Observation Specification.

**Contents of the Scan List prior to Phase 3**

- A configuration scan, typically 10 minutes long, is required. It has `Scan Intent` of SYSTEM_CONFIGURATION.
- *Phase 2* selects appropriate Observing Targets for flux density calibration and bandpass calibration. In this example, the same one is suitable for as both the flux density and bandpass calibrator, and furthermore, it is suitable for both HARDWARE CONFIGURATIONS. The selected Observing Target is called OT-$B_{5GHz}$ and OT-$B_{9GHz}$ for 5 and 9 GHz HARDWARE CONFIGURATIONS, respectively.
  - *Phase 2* determines that the time spent collecting data for Observing Targets with Source OT-B is 280 seconds and a only single *Subscan* is needed. Thus, the `Acquisition Time` is 280 seconds. There `Subscan Intent` is ON_SOURCE, and the `Scan Intent` is CALIBRATE_FLUX and CALIBRATE_BANDPASS.
  - The duration of a *Subscan* can be greater than the `Acquisition Time`, as it accounts for the `Acquisition Time` plus any additional time that is not the collecting data. The latter can include the time it takes for the antenna to slew to the position on the sky of the Observing Target (`antenna slew time`), time for the antenna to settle after its movement (settle time), and the time required for any changes to the HARDWARE CONFIGURATION(e.g., changing from X-band to C-band). The sum of these is called the `Setup Time`.
  - The § 21.1 algorithm determines the `Setup Time` for this *Subscan* is 20 seconds.
  - The duration of the Scan is the sum of all the *Subscan* times.

[1]Note, Science Targets are Observing Targets but not all Observing Targets are Science Targets (e.g., calibrators are not Science Targets). The use of these two terms is interchangeable without any loss of meaning for Science Targets. It is a useful distinction, however, for calibrators, particularly when answering questions such as "How much time did the antenna spend collecting data on science sources versus calibrators?"

[2]There are four Science Targets from the pairing of the FIELD SOURCES with the SPECTRAL SPECIFICATIONS.

- In this example, the `RMS Sensitivity` determines how much time is needed to address the motivating science of the proposal[3]. The `RMS Sensitivity` is 100 $\mu$Jy bm$^{-1}$. The VLA Exposure Time Calculator returns $\sim$ 5 minutes as being sufficient to reach this sensitivity (Figure 26.1). Therefore, the `Requested Time` is 5 minutes per Science Target.

- The § 21.1 algorithm determines that 22 seconds is the `Setup Time` associated with *Scans* of the Science Targets.

- *Phase 2* selects an appropriate Observing Target for the Phase Referencing calibration. Fortunately, the same Phase Referencing Observing Target is a good choice for both HARDWARE CONFIGURATIONS. The selected Observing Target is called OT-$A_{5GHz}$ and OT-$A_{9GHz}$ for the 5 and 9 GHz HARDWARE CONFIGURATIONS, respectively.
  - *Phase 2* determines that 80 seconds is the `Acquisition Time` needed in any *Subscan* of the Phase Referencing Observing Target.
  - The `Setup Time` associated with an observation of the Phase Referencing Observing Target is 20 seconds.

- Observations of the Science Targets will be interleaved with those of the Phase Referencing Observing Target. The Cycle Time for C-configuration at 5 and 9 GHz is 10 minutes (Figure 20.1). Given the `Scan Duration` of the Phase Referencing calibration, there are 400 seconds (600 seconds - (2 $\times$ 100 seconds)) in a Phase Referencing cycle available for *Scans* of the Science Targets because the calibrator must be returned to before the end of the Cycle Time. The `Requested Times` must be distributed across multiple *Subscans*/*Scans* because the total `Requested Time` is already greater than or equal to the Cycle Time. The suitable approach for two HARDWARE CONFIGURATIONS is to interleave full cycles instead of switching to a different HARDWARE CONFIGURATION during a Phase Referencing cycle.
  - Additional constraints, such as a need for uv-coverage, may set a Maximum `Acquisition Time` for *Subscans*.
    - For this example, the Maximum `Acquisition Time` for Observing Target I1 is 120 seconds.
    - Observing Target I5 does not need to share, or have, a Maximum `Acquisition Time`[4]. For this example, Observing Target I5 does not have a Maximum `Acquisition Time`.
  - A `maximum duration`, which limits the total time of a Scan (`Acquisition Time` + `Setup Time`), can be set. For this example, a `maximum duration` of 300 seconds is set for the Science Targets.
  - As a rule, the `Acquisition Times` for the same Observing Targets (with equivalent `Scan Intent` and equivalent `Subscan Intent`) should be the nearly equivalent.
  - Accounting for the Maximum `Acquisition Time`, `maximum duration`, and Cycle Time, 3 *Scans* of $I1_{5GHz}$ and $I1_{9GHz}$, and 2 *Scans* of $I5_{5GHz}$ and $I5_{9GHz}$ are needed to achieve the `Requested Time` .
  - For *Subscans* that share `Scan Intent` OBSERVE_TARGET, `Subscan Intent` ON_SOURCE, and are the same Science Target, the sum of the `Acquisition Time` is the `Science Target Integration Time` for that Science Target. Note, the `Science Target Integration Time` should only be greater than or equal to the `Requested Time` .

**F.A.Q.–**

Q. What is the total time of Observation Specification (which becomes a VLA Scheduling Block?)

[3] We have intentionally shortened the `Requested Time` to simplify this illustration. There are other factors (e.g., `Pointing Pattern`, uv-coverage) that can contribute to the determination of the `Requested Time`. For Faraday Rotation studies, parallactic angle is a driving consideration for example

[4] The details of how the `Requested Time` is distributed amongst the *Subscans* for Observing Targets are in § 9.2.1

A. Duration = $\sum$ `Setup Time` + `Time on Observing Targets`

Q. How much time is spent observing the Science Targets?

A. `Science Target Integration Times`

Q. How much time is spent observing the calibrators?

A. `Time on Observing Targets` - `Science Target Integration Times`

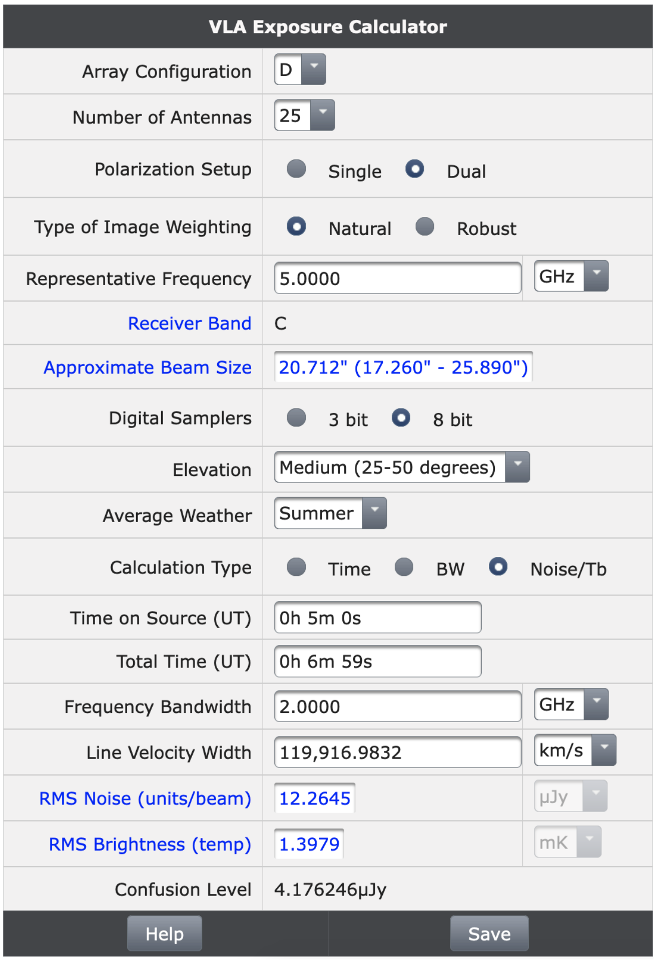



**Figure 26.1:** Example of Use Case with Time Concepts

**Table 26.1:** VLA Example Observation Specification Part 1: Scan List

| Scan Intent | Observing Target | Setup Time (s) | Acquisition Time (s) | Subscan Intent |
|---|---|---|---|---|
| SYSTEM_CONFIG | OT-$A_{5GHz}$ | 600 | 0 | UNSPECIFIED |
| CALIBRATE_FLUX | OT-$B_{5GHz}$ | 20 | 280 | ON_SOURCE |
| CALIBRATE_FLUX | OT-$B_{9GHz}$ | 20 | 280 | ON_SOURCE |
| | Phase Referencing Cycles | | | |
| CALIBRATE_PHASE; CALIBRATE_AMPLI | OT-$A_{5GHz}$ | 20 | 80 | ON_SOURCE |
| OBSERVE_TARGET | $I1_{5GHz}$ | 22 | 110 | ON_SOURCE |
| OBSERVE_TARGET | $I5_{5GHz}$ | 22 | 150 | ON_SOURCE |
| OBSERVE_TARGET | $I1_{5GHz}$ | 22 | 110 | ON_SOURCE |
| CALIBRATE_PHASE; CALIBRATE_AMPLI | OT-$A_{5GHz}$ | 20 | 80 | ON_SOURCE |
| CALIBRATE_PHASE; CALIBRATE_AMPLI | OT-$A_{9GHz}$ | 20 | 80 | ON_SOURCE |
| OBSERVE_TARGET | $I1_{9GHz}$ | 22 | 110 | ON_SOURCE |
| OBSERVE_TARGET | $I5_{9GHz}$ | 22 | 150 | ON_SOURCE |
| OBSERVE_TARGET | $I1_{9GHz}$ | 22 | 110 | ON_SOURCE |
| CALIBRATE_PHASE; CALIBRATE_AMPLI | OT-$A_{9GHz}$ | 20 | 80 | ON_SOURCE |
| CALIBRATE_PHASE; CALIBRATE_AMPLI | OT-$A_{5GHz}$ | 20 | 80 | ON_SOURCE |
| OBSERVE_TARGET | $I5_{5GHz}$ | 22 | 150 | ON_SOURCE |
| OBSERVE_TARGET | $I1_{5GHz}$ | 22 | 110 | ON_SOURCE |
| CALIBRATE_PHASE; CALIBRATE_AMPLI | OT-$A_{5GHz}$ | 20 | 80 | ON_SOURCE |
| CALIBRATE_PHASE; CALIBRATE_AMPLI | OT-$A_{9GHz}$ | 20 | 80 | ON_SOURCE |
| OBSERVE_TARGET | $I5_{9GHz}$ | 22 | 150 | ON_SOURCE |
| OBSERVE_TARGET | $I1_{9GHz}$ | 22 | 110 | ON_SOURCE |
| CALIBRATE_PHASE; CALIBRATE_AMPLI | OT-$A_{9GHz}$ | 20 | 80 | ON_SOURCE |

**Table 26.2:** VLA Example Observation Specification Part 2: Summary

| Source Name | Center Frequency (GHz) | Time on Observing Target (s) | Science Target Integration Time (s) |
|---|---|---|---|
| OT-B | 5 | 280 | 0 |
| OT-B | 9 | 280 | 0 |
| OT-A | 5 | 240 | 0 |
| OT-A | 9 | 240 | 0 |
| I1 | 5 | 330 | 330 |
| I1 | 9 | 330 | 330 |
| I5 | 5 | 300 | 300 |
| I5 | 9 | 300 | 300 |

| | |
|---|---|
| Science Target Integration Times = $\sum$ Science Target Integration Time | 1260 s |
| Time on Observing Targets = $\sum$ Time on Observing Target | 2300 s |
| Duration = $\sum$ Setup Time + Time on Observing Targets | 3520 s |
| Overhead = Duration - Science Target Integration Times | 2260 s |

### 26.1.2 GBT Observations of Two Field Sources at One Frequency

Consider a GBT observations of two SOURCES, called 'A' and 'B' at one HARDWARE CONFIGURATION. The *Observing Strategy* determined the HARDWARE CONFIGURATION and that both SOURCES have a `Pointing Pattern` of Single Pointing. There are two Science Targets in this *Science Target List*.

The *Capability* selects the appropriate *Calibration Strategy* and *Scheduling Strategy*, which dictate to the *Observation Planner Phase 1*, 2, and 3 how to partition, calibrate, and schedule the observations. For this example, the *Observation Planner* determines that the Science Targets will be partitioned into one *Observation Specification* and that there will be Position Switching and Focus calibrations. *Phase 3* determines the order of the *Scan List*. The following paragraphs describe the contents of the *Scan List* that the *Observation Planner* creates. Table 26.3 shows the *Scan List* and summary of the Observation Specification for this example.

**Contents of Scan List prior to Phase 3**

**Table 26.3:** GBT Example Observation Specification Part 1: Scan List

| Scan Intent | Observing Target | Setup Time (s) | Acquisition Time (s) | Subscan Intent |
|---|---|---|---|---|
| CALIBRATE_FOCUS | Cal-A | 20 | 180 | ON_SOURCE |
| OBSERVE_TARGET | A | 22 | 110 | ON_SOURCE |
| OBSERVE_TARGET | A | 22 | 110 | OFF_SOURCE |
| OBSERVE_TARGET | B | 22 | 150 | ON_SOURCE |
| OBSERVE_TARGET | B | 22 | 150 | OFF_SOURCE |

**Table 26.4:** GBT Example Observation Specification Part 2: Summary

| Source Name | Center Frequency (GHz) | Time on Observing Target (s) | Science Target Integration Time (s) |
|---|---|---|---|
| Cal-A | | 180 | 0 |
| A | | 220 | 110 |
| B | | 300 | 150 |

| | |
|---|---|
| Science Target Integration Times = $\sum$ Science Target Integration Time | 260 s |
| Time on Observing Targets = $\sum$ Time on Observing Target | 700 s |
| Duration = $\sum$ Setup Time + Time on Observing Targets | 808 s |
| Overhead = Duration - Science Target Integration Times | 548 s |

# Part VI

# Proposal Process Subsystems

# 27 | Proposal Process Overview

The *Proposal Processes* currently implemented are the Panel Proposal Review (PPR) process and the Observatory Site Review (OSR) process. PPR and OSR processes consist of different phases:

- Review Configuration
- Review Process
- Allocation Disposition
- Allocation Approval
- Closeout

The details of the phases are specific to the *Proposal Process*, which are discussed in the following chapters.

# 28 Panel Proposal Review Process

## 28.1 PPR Process Subsystem Overview

There are a number of different user roles during the PPR process, which restrict access to components of the system, process, and UI as required. They include

- Science Review Panel (SRP) member
  - A SRP member can only belong to one SRP panel, though there may be multiple panels. They author a Individual Science Review (ISR) per proposal to which they are assigned. They also participate in the *Consensus* meeting with other SRP members on their panel, which reviews all of the ISRs per proposal to reach a consensus opinion.
- SRP Chair
  - A SRP Chair has all the responsibilities and abilities of a SRP member and then additional ones. They manage the panel during the *Review Process* phase and are a TAC member.
- Feasibility Reviewer
  - A Feasibility Reviewer authors feasibility reviews per proposal to which they are assigned.
- Telescope Time Allocation (TAC) member
- TAC Chair
- TTA member

Table 28.1 lists the associated sections with phases of the PPR *Proposal Process*.

**Table 28.1:** Section by PPR Phase

| Phase | § |
|---|---|
| Review Configuration | 28.2 |
| Review Process | 28.3 |
| Allocation Disposition | 28.3.5 |
| Allocation Approval | 28.3.6 |
| Closeout | |

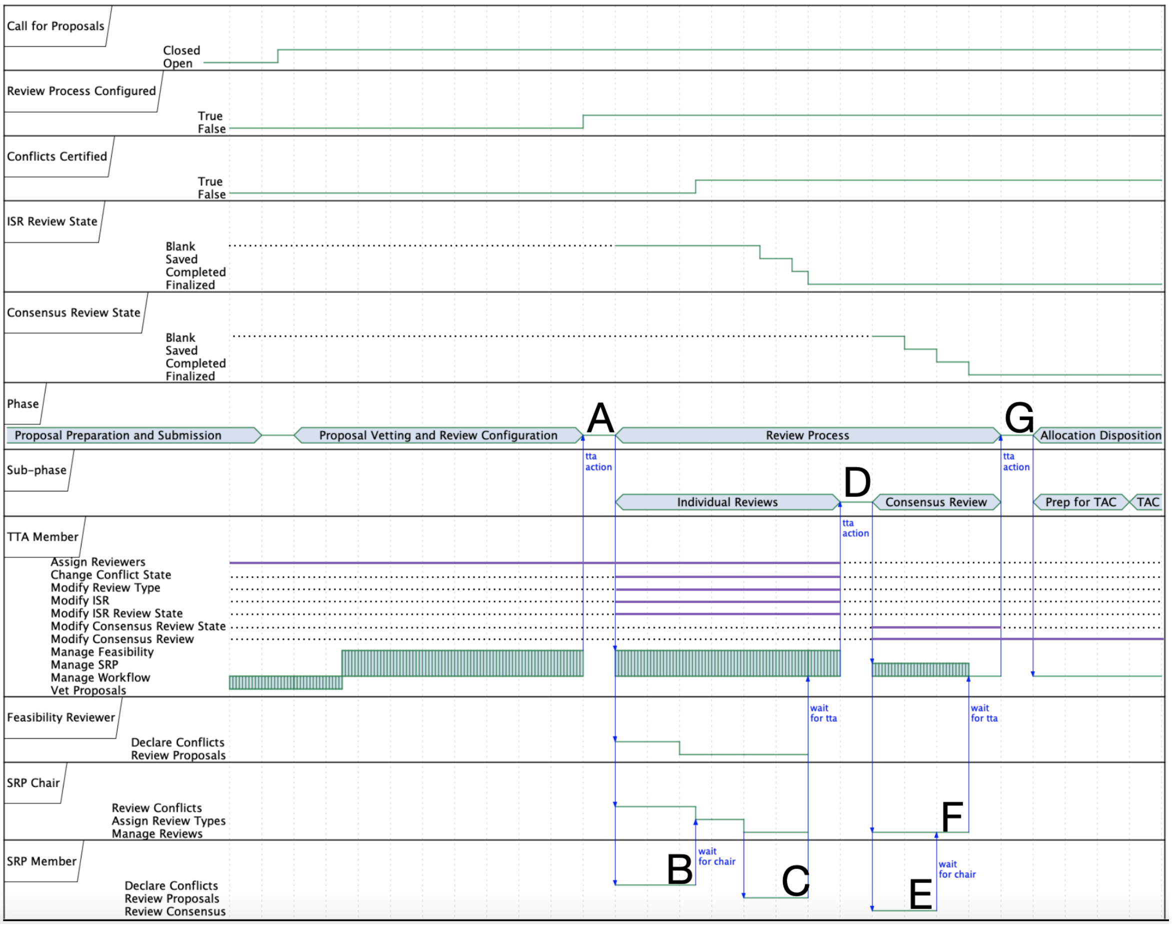



**Figure 28.1:** Big picture summary of the PPR Proposal Process. The top two rows show key properties that account for progress or transitions in the process. The third row illustrates the relationship between a SRP member's workflow and Conflict Declaration. The fourth row is an example of an ISR's Review State. The fifth row is an example of a Proposal's Proposal REVIEW STATE. The sixth row is the phase of the Proposal Process; the seventh is the sub-phase, if applicable. The eighth row shows the TTA member's workflow through the Proposal Process. A select set of actions is shown, where a solid purple line indicates the TTA member can perform the action. A dotted black line indicates the action is forbidden. A single Feasibility Reviewer's, SRP Chair's, SRP Chair's general tasks are shown in rows nine through ten, respectively. The bold letters indicate important actions a TTA member, SRP member, or SRP Chair must take to advance the system: (A) a TTA member finalizes the Review Configuration; (B) a SRP member certifies their conflicts; (C) a SRP member finalizes their ISRs; (D) a TTA member initiates Consensus sub-phase on a per panel basis; (E) a SRP member completes the Proposal Reviews; (F) a SRP Chair finalizes Proposal Reviews; (G) a TTA member closes the Review Process phase by generating the NLR Scores.

### 28.1.1 Scores in the PPR Process

- INDIVIDUAL SCORE a user input value per proposal per reviewer; not modifiable once ISRs are completed
- NORMALIZED INDIVIDUAL SCORE - a calculated value per proposal per reviewer; it is the normalization of the INDIVIDUAL SCORE for all proposals per reviewer; not modifiable
- FINALIZED NORMALIZED INDIVIDUAL SCORE - initially a copy of the NORMALIZED INDIVIDUAL SCORE but can be modified during the *Consensus* meeting (re-voting) by the SRP Chair or TTA member
- MEAN NORMALIZED SCORE average of all reviewers' FINALIZED NORMALIZED INDIVIDUAL SCOREs per proposal; not modifiable by user but always updated if the FINALIZED NORMALIZED INDIVIDUAL SCORE is updated.
- STANDARD DEVIATION OF THE MEAN NORMALIZED SCORE not modifiable by the user but always updated if the MEAN NORMALIZED SCOREs of a proposal changes
- SRP SCORE a copy of the MEAN NORMALIZED SCORE but can be modified by the SRP Chair or TTA member or via an updated to the MEAN NORMALIZED SCORE. If it is modified by the SRP Chair or TTA member, it cannot be updated by the MEAN NORMALIZED SCORE

### 28.1.2 Transitions between Phases in the PPR Process

At *Solicitation* configuration and if a PPR process specified, the *Review Configuration* is accessible to a TTA member to begin configuring the review panels (§ 28.2). The *Review Configuration* continues after the *Solicitation* is closed, and subsequent transitions between phases and sub-phases are generally marked by an action a TTA member initiates in the system based on a set of conditions.

### 28.1.3 Out of Process Actions by a TTA Member

The majority of use cases are supported by the prescribed workflow in these sections. There are anticipated edge cases though for which the system is designed with a level of flexibility to accommodate. Primarily, the TTA member has extended administrative powers to overcome the anticipated edge cases. These administrative powers are described here and Figure 28.1 shows when these actions can be taken with respect to the PPR process workflow.

A TTA member can modify an ISR's comments, score, REVIEW TYPE, and REVIEW STATE at any time that is not explicitly restricted in the following list:

- The Comments and Scores of an ISR can be modified until the Consensus sub-phase is initiated for the panel.
- The REVIEW STATE and REVIEW TYPE may be modified until the Consensus sub-phase is initiated for the panel. Some notes what constitutes good TTA member workflow:
  - In almost all cases, the TTA member should avoid changing the REVIEW STATE of a single ISR to Finalized, as the NIS is only calculated or recalculated when the bulk Finalize action is used.
  - The TTA member should carefully consider their intent when changing the REVIEW TYPE from None if the REVIEW STATE is Closed. Changing the REVIEW TYPE would allow, potentially, the SRP member to edit the Proposal Review COMMENTS TO PI and INTERNAL COMMENTS.

– After the SRP member has finalized their ISRs, the TTA member should almost always use the bulk Finalize action again after modifying a REVIEW TYPE or a REVIEW STATE as there is not an automatic recalculation of the NIS so the state of the ISRs may not be consistent with the previously calculated NIS. The exception to this statement is if the TTA member is adding an external ISR to a SRP member's workload and only changing the newly added ISR's REVIEW TYPE. In this case, the proper workflow is for the TTA member to add the external ISR, set the REVIEW TYPE to Primary, Secondary, or Tertiary, and use the "Save" button. The SRP member will then be able to do their review and use the bulk Finalize action. If the TTA member uses the bulk finalize button in this case, there will be unintended consequences; see Section 29.1.2.

A TTA member can modify a Proposal Review COMMENTS TO PI and INTERNAL COMMENTS at any point in the system until Disposition Letters are sent and the PROPOSAL STATE is Completed.

## 28.2 PPR Review Configuration

Under Construction.

## 28.3 PPR Review Process

### 28.3.1 Conflict Declaration and Conflict States

A TTA member can change any conflict declaration but persisting the change requires the TTA member to certify the conflicts for the reviewer. If a reviewer has already certified their conflicts, then the TTA member may re-certify. Generally, the following behavior and workflow is supported in the system.

- If a SRP member has not certified their conflicts, then any one change the TTA member makes will cause the conflicts to certify and the SRP member will not have done so directly.
- If the SRP member is unable to certify, then the TTA member may use this bulk certification for them, provided they have communicated to the reviewer.
- If the SRP member has indicated that an automatic conflict, which is done prior to their certification, is incorrect, then the TTA member should wait until the SRP member has certified their conflicts. Then, the TTA member can make the appropriate changes and re-certify.

### 28.3.2 Individual Science Review Sub-Phase

An Individual Science Review (ISR) consists of

- an INDIVIDUAL SCORE
- COMMENTS TO THE SRP
- a REVIEW TYPE
- a REVIEW STATE

#### 28.3.2.1 Review Types and ISR Review States

Parts of the system rely on certain conditions to trigger an action or to allow a user to take an action. Some of these conditions are defined by rule sets that rely on the combination of the REVIEW TYPE and the REVIEW STATE. This section describes the rules and triggers the system relies on for the PPR process.

During the ISR sub-phase, the REVIEW TYPE and REVIEW STATES of ISRs may be modified.

- A TTA member may modify the REVIEW TYPE regardless of the REVIEW STATE until the Consensus sub-phase has started for the panel that contains that ISR of interest.
- The SRP Chair may modify a REVIEW TYPE of an ISR if the SRP member has certified their conflicts and the CONFLICT STATE is not equal to Conflicted (automatically or user declared) and if the REVIEW STATE is not equal to Finalized or Closed.
- SRP members may modify the REVIEW STATE of their ISR through specific actions, such as using the save, complete, or finalize functionality in the UI.

Table 28.2 lists the allowed combinations of REVIEW TYPES and ISR REVIEW STATES. An ISR can only have one REVIEW TYPE and one REVIEW STATE, so when multiple are listed in a cell, an "or" should be assumed between each comma and an "and" should be assumed between columns. For example in the first row, an ISR with a REVIEW TYPE of Primary and a REVIEW STATE of Blank; a REVIEW TYPE of Secondary and a REVIEW STATE of Blank; a REVIEW TYPE of Tertiary and a REVIEW STATE of Blank; or a REVIEW TYPE of None and a REVIEW STATE of Blank are all valid combinations. The last column of the table gives the reference for a description in the text.

Table 28.3 lists the allowed transitions between ISR REVIEW STATE qualified by the REVIEW TYPE. As before, a comma within a cell should be interpreted as an "or" and between columns an "and". The use of directional arrows gives the allow state model transition. For example in the third row, an ISR with a REVIEW TYPE of Primary and a REVIEW STATE of Saved or Completed may change to that of Finalized. Similarly, a REVIEW TYPE of Secondary or Tertiary is valid for such a state model transition. The final column in the table gives the appropriate reference, which contains additional explanation such as how to trigger the state transition.

Table 28.4 lists the correlation of ISR REVIEW STATE and REVIEW TYPE functionality for the ISR sub-phase. The final column in the table gives the appropriate reference, which contains additional explanations.

**Table 28.2:** Summary of allowed combinations of Review Types and ISR Review States

| REVIEW TYPE | ISR REVIEW STATE | Reference[a] |
|---|---|---|
| Primary, Secondary, Tertiary, None | Blank | ISR-1 |
| Primary, Secondary, Tertiary, None | Saved | ISR-2 |
| Primary, Secondary, Tertiary, None | Completed | ISR-3 |
| Primary, Secondary, Tertiary, None[b] | Finalized | ISR-4 |
| None | Closed | ISR-5 |

[a] See text for associated description.

[b] In practice, a REVIEW TYPE of None is not combined with an ISR REVIEW STATE of Finalized. It can only occur if the TTA member manually modifies a REVIEW TYPE and does not modify the REVIEW STATE.

**Table 28.3:** Allowed ISR Review State Transitions

| REVIEW TYPES | ISR REVIEW STATE | Reference[a] |
|---|---|---|
| Primary, Secondary, Tertiary, None | Blank → Saved | ISR-2 |
| Primary, Secondary, Tertiary, None | Blank, Saved → Completed | ISR-3 |
| Primary, Secondary, Tertiary | Saved, Completed → Finalized | ISR-3; IS4-4 |
| Primary, Secondary, Tertiary | Blank, Saved, Completed, Finalized → Closed | ISR-5 |
| None | Blank, Saved, Completed → Closed | ISR-5 |
| None → Primary, Secondary, Tertiary | Closed → Blank, Saved, Completed | ISR-6 |

[a] See text for associated description.

**Table 28.4:** Allowed Review Types Transitions by ISR Review State

| REVIEW TYPE | ISR REVIEW STATE | Reference[a] |
|---|---|---|
| Primary ↔ Secondary ↔ Tertiary | Blank, Saved, Completed, Finalized | ISR-7 |
| Primary, Secondary, Tertiary ↔ None | Blank, Saved, Completed | ISR-8 |

[a] See text for associated description.

Below are additional descriptions relevant to the tables presented in this section. The list is independently useful to review for a holistic understanding of the expected and anticipated behavior of the PPR process.

ISR-1 Blank is the default REVIEW STATE of an ISR when first created in the system.

ISR-2 A valid ISR can be saved via the "Save" button in the UI. A valid ISR consists of

- a INDIVIDUAL SCORE between [0.1 – 9.9],
- COMMENTS TO THE SRP that contain at least one character.

For a valid ISR that has a REVIEW STATE of Blank, the state model will transition to that of Saved. It is a triggered on a per ISR basis. An SRP member can save as many times as needed, though the state change only occurs on the first instance. Once an ISR is Saved, it cannot be reverted to the Blank state by the SRP member.

ISR-3 The REVIEW STATE of Completed is a convenient bookkeeping tool for the SRP member to track progress on their individuals reviews. It is an optional state in the state model. It requires a valid ISR, as defined in ISR-2. The state transition occurs when a SRP member uses the "Completed" button; it is a per ISR action and will also save the ISR if the ISR is valid. If the "Save" button is used after this point, the REVIEW STATE will be marked as Saved. Once an ISR is Completed, it cannot be reverted to the Blank state by the SRP member.

ISR-4 For a SRP member, if an ISR with a REVIEW TYPE of Primary, Secondary, or Tertiary exists with a REVIEW STATE of either Saved, or Completed, the SRP member may finalize their reviews via the "Finalize" button in the UI. The use of the "Finalize" button does not change the REVIEW STATE of

- ISRs with a REVIEW TYPE of None,
- ISRs with a REVIEW STATE of Closed,
- ISRs with a REVIEW STATE of Finalized.

ISRs with a REVIEW STATE of Finalized cannot be further modified by the SRP member, and the act of finalizing will calculate the NORMALIZED INDIVIDUAL SCORE (§ 29.1.1). The SRP member may finalize additional times if an external ISR is assigned to the SRP member; the external ISR has a REVIEW TYPE of Primary, Secondary, or Tertiary; and the external ISR has a REVIEW STATE not equal to Finalized or Closed. The qualifying ISRs will transition to a REVIEW STATE of Finalized and the NORMALIZED INDIVIDUAL SCORES are recalculated[1].

ISR-5 Only a TTA member can change the REVIEW STATE of an ISR to Closed, which will automatically change the REVIEW TYPE to None. An ISR with a REVIEW STATE of Closed enforces a REVIEW TYPE of None. See Section 28.1.3.

ISR-6 If a REVIEW TYPE of None is changed to Primary, Secondary, or Tertiary, the REVIEW STATE can then be changed from Closed to Blank, Saved, Completed, or Finalized. Only a TTA member can facilitate these changes within the constraints in Section 28.1.3.

ISR-7 A SRP Chair may assign or change the REVIEW TYPE of an ISR until the ISR REVIEW STATE is Finalized and the SRP member has certified their conflicts. The transition between Primary, Secondary, Tertiary, and None has no functional consequences during the ISR sub-phase with regards to the level of access a SRP member has to a Proposal, as that is determined by the CONFLICT STATE (§ 28.3.1). See Section 28.3.4 for implications of REVIEW TYPE in the Consensus sub-phase however.

ISR-8 SRP members are allowed to save and mark complete ISRs that have REVIEW TYPES of None. Though SRP members are not expected to review Proposals for which they are not assigned a REVIEW TYPE of Primary, Secondary, or Tertiary, they may still participate in the *Consensus* discussion for Proposals for which they are not conflicted on. However, the COMMENTS TO THE SRP and INDIVIDUAL SCORE for any ISR with REVIEW TYPE of None will not be available in the aggregated comments for the Proposal in the Consensus UI nor will the INDIVIDUAL SCORE of the ISR(s) affect the calculation of the

- NORMALIZED INDIVIDUAL SCORE
- FINALIZED NORMALIZED INDIVIDUAL SCORE
- MEAN NORMALIZED SCORE
- STANDARD DEVIATION OF THE MEAN NORMALIZED SCORE
- SRP SCORE.

### 28.3.3 Feasibility Review

Under Construction.

### 28.3.4 Proposal Review in Consensus Sub-Phase

During the Consensus sub-phase, the SRP members and SRP Chair will participate in a *Consensus* discussion to come to a consensus opinion for Proposals in their panel. This consensus opinion is recorded in the Proposal's Proposal Review, which consists of

- a Proposal Review REVIEW STATE

[1] Depending on the timing of the start of the SRP member's Consensus sub-phase vs the assignment of the external ISR, the recalculated NISs may not be consistent with the NISs of the SRP member's Consensus panel. This is an acceptable discontinuity. This process has the advantage that the external ISR's INDIVIDUAL SCORE will be normalized in the context of all of the SRP member's ISRs.

- a SRP SCORE
- a MEAN NORMALIZED SCORE
- a STANDARD DEVIATION OF THE MEAN NORMALIZED SCORE
- a COMMENTS TO THE PI
- a INTERNAL COMMENTS
- TECHNICAL COMMENTS per Allocation Request per Proposal, if available
- DATA MANAGEMENT COMMENTS per Allocation Request per Proposal, if available
- COMMENTS FOR THE SRP per ISR
- NORMALIZED INDIVIDUAL SCORE per ISR
- FINALIZED NORMALIZED INDIVIDUAL SCORE per ISR

The Proposal Review REVIEW STATE controls the workflow of a Proposal through the Consensus sub-phase. It is independent of any ISR REVIEW STATE. Table 28.5 gives the Proposal Review REVIEW STATES and additional descriptions of the system's behavior.

The ISR REVIEW STATE and REVIEW TYPE determine the inclusion of the ISR's COMMENTS TO THE SRP to the Proposal Review and the inclusion of the ISR's INDIVIDUAL SCORE in calculating the NORMALIZED INDIVIDUAL SCORE. Once the Consensus sub-phase is initiated for a panel, the COMMENTS TO THE SRP per ISR and the NORMALIZED INDIVIDUAL SCORES are populated into the Proposal Review. Once a panel's Consensus sub-phase is started, no modification can be made to the ISR in any way, even by the TTA member (§ 28.1.3).

A SRP member with a CONFLICT STATE of Available for a Proposal can see the Proposal Review for that proposal. Only an SRP member who has an ISR with a REVIEW TYPE of Primary or Secondary may enter or modify the COMMENTS FOR THE PI and INTERNAL COMMENTS for the associated Proposal. Saving or marking a Proposal Review as Completed can only be done be a Primary or Secondary.

**Table 28.5:** State Model for Proposal Review

| REVIEW TYPE | Consensus REVIEW STATE | Reference[a] |
|---|---|---|
| Primary, Secondary | Blank → Saved → Completed | Consensus-1 |
| SRP Chair | Completed → Finalized | Consensus-2 |

[a] See text for associated description.

Consensus-1 — For a Proposal, the SRP member who has the ISR with a REVIEW TYPE of Primary or Secondary may save and/or mark a Proposal Review as complete. Once completed, only the SRP Chair (or TTA member) may make further modifications to the Proposal Review.

Consensus-2 — Only the SRP Chair (or a TTA member) can finalize a Proposal Review. Once finalized, the SRP Chair can no longer make modifications to the Proposal Review.

Once all Proposal Review have been finalized in a panel, the Normalized Linear Rank (NLR) is calculated per panel (Section 29.1.3).

### 28.3.5 PPR Allocation Disposition

Under Construction.

### 28.3.6 PPR Allocation Approval

Under Construction.

## 28.4 PPR Edge Cases

# Part VII

# Proposal Process Algorithms

# 29 | Proposal Panel Process Algorithms

## 29.1 | PPR Review Process Algorithms

### 29.1.1 | Method: Finalize ISRs

**Method** For a SRP Member, their ISRs with REVIEW TYPES of Primary, Secondary, and Tertiary and REVIEW STATE of Finalized are used to calculate the NORMALIZED INDIVIDUAL SCORE, which is calculated with the following equation:

$$\hat{s} = \frac{\hat{\sigma}}{\sigma} s + \left( \hat{\mu} - \frac{\hat{\sigma}}{\sigma} \mu \right), \tag{29.1.1}$$

where $\hat{s}$ is the NORMALIZED INDIVIDUAL SCORES of a population of proposals, $s$ is the raw scores of the population of proposals, $\hat{\sigma}$ is the desired standard deviation, $\sigma$ is the standard deviation of population of proposals, $\hat{\mu}$ is the desired mean, and $\mu$ is the mean of the population of proposals. Here, $\hat{\mu} = 5$ and $\hat{\sigma} = 2.5$.

**Validation** All of the INDIVIDUAL SCORES included in the set to calculate the NORMALIZED INDIVIDUAL SCORE must not be equal. If they are all equal, a validation error is given to the user and Finalization fails.

**Behavior** If only one ISR's INDIVIDUAL SCORE is included in the set to calculate the NORMALIZED INDIVIDUAL SCORE, then the NORMALIZED INDIVIDUAL SCORE equals the INDIVIDUAL SCORE.

### 29.1.2 | Method: Bulk Finalize ISRs

- If the TTA member uses the bulk "Finalize" action (i.e., the Finalize button in the UI) for an SRP member, it will transition ISRs with
  - a REVIEW TYPE of Primary, Secondary, or Tertiary **and** a REVIEW STATE of Saved or Completed will have a REVIEW STATE transition to Finalized.
  - a REVIEW TYPE of Primary, Secondary, of Tertiary **and** a REVIEW STATE of Blank will have a REVIEW STATE transition to Closed.
  - a REVIEW TYPE of None will not have a state transition, regardless of the REVIEW STATE.
  - an ISR with a REVIEW STATE of Finalized or Closed will not have a state transition.
  - The NIS is calculated; see Section 29.1.1.
- When a TTA member initiates the Consensus sub-phase for a panel, all ISRs are transitioned to an immutable state for the SRP Member.

### 29.1.3 | Method: Normalized Linear Rank

The Normalized Linear Rank (NLR) is calculated per panel once all Proposal Reviews have a REVIEW STATE of Finalized. The proposals are assembled into an ordered list by their SRP Score from 0.1 to 9.99. The index in the ordered list is denoted as $R$ and the NRL for each proposal in the panel is then

$$\text{NLR} = \frac{R * 10}{N}, \tag{29.1.2}$$

where $N$ is the number of proposals in the panel.

For example, the "best" proposal (i.e., the smallest SRP Score in the panel) would be given a rank of 1 in the panel. If the panel had 12 proposals, the "best" proposal then has a NLR of 0.83.

# Index