TTAT Algorithms v0.2

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

April 8, 2022

0.1 Updates

- 3.27.22
  - Updating Partitioning to Phase 1, Calibration Strategy to Phase 2, Scheduling Strategy to Phase 3. The ScienceTargetList carries with it a Calibration Strategy and a Scheduling Strategy, which are selected by the Capability.
  - New Chapters to describe the Calibration and Scheduling Strategies, which are NOT Phase 2 and Phase 3. At the conclusion of these Chapters, a "choose your observing adventure" prescription should be established to then navigate through the *Observation Planner*.
  - Folding in usecases.
    1. Updated Section 10.1.1 to mirror 688-TTAT-xxx-MGMT TTA Use Cases v0.1 Section 3.1
  - Updated VLA Observing Strategy 3 and the beginning of Chapter 2.
  - Updated Figure 1.1
  - Need to update other Figures in § 2
- 1.18.22 – 2.17.22
  - Added 3 –
  - Updated Words (e.g., Target list to Science Target List)
- 1.13.22
  - Added Section 11.1
  - Added Section 11.4.
- 12.14.2021
  - Added Section 10.1
  - Added items to Definitions and Concept List at top of Chapter 10
- 12.10.2021
  - Changed DocumentClass to Report; updated syntax as needed
  - Updated Figure 1.1
- 12.1.2021
  - Moved to Overleaf
  - Folded in "Doodad" document
  - Added concept of Source (defined in Observing Strategy)
  - Changed Target to Science Target; *Target List* to *Science Target List*
  - Added concept of Observing Target, Requested Time, Setup Time, Acquisition Time, Observing Instructions (defined in Calibration Strategy)

Contents

List of Figures

List of Tables

Part I

Chapter 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 algorithms. The intended audience is the implementation team and scientists.

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 How to Navigate this Document

1.4 Overview of Concepts Upstream of the Algorithms

The *Capability Parameter Specifications* are provided by a TTA Group member that specifies the parameters that make up a *Capability* for a *Solicitation.* 688-TTAT-004-MGMT System Description v3.0 Section 3.1 and Table 2 define the grouping of the *Capability Parameter Specifications* as

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

Tables 1.1, 1.2, 1.3, and 1.4 overview the groups of *Capability Parameter Specifications* per *Capability.* The *Capability Request Parameters* are then the user's response to the *Capability Parameter Specifications*, and they provide the user supplied information about the requested observations.

The *Capability Request* is composed of the *Capability Request Parameters*, and within a single *Capability Request*, there can be multiple FIELD SOURCES and SPECTRAL SPECIFICATIONS. Generally, a single set of PERFORMANCE PARAMETERS and CALIBRATION PARAMETERS can define any one *Capability Request.* However, the RMS Sensitivity can be specified for each FIELD SOURCE + SPECTRAL SPECIFICATION pair. Table 2 of 688-TTAT-004-MGMT System Description v3.0 provides addition information on the multiplicity of the *Capability Parameter Specifications.*

The *Capability Request* is passed downstream to the algorithm to construct one or more *Observation Specifications.* The remainder of this document provides the details of the algorithms needed to create an *Observation Specification*. Figure 1.1 shows a simple diagram of the algorithms utilized to transform the *Capability Request* into an *Observation Specification*.

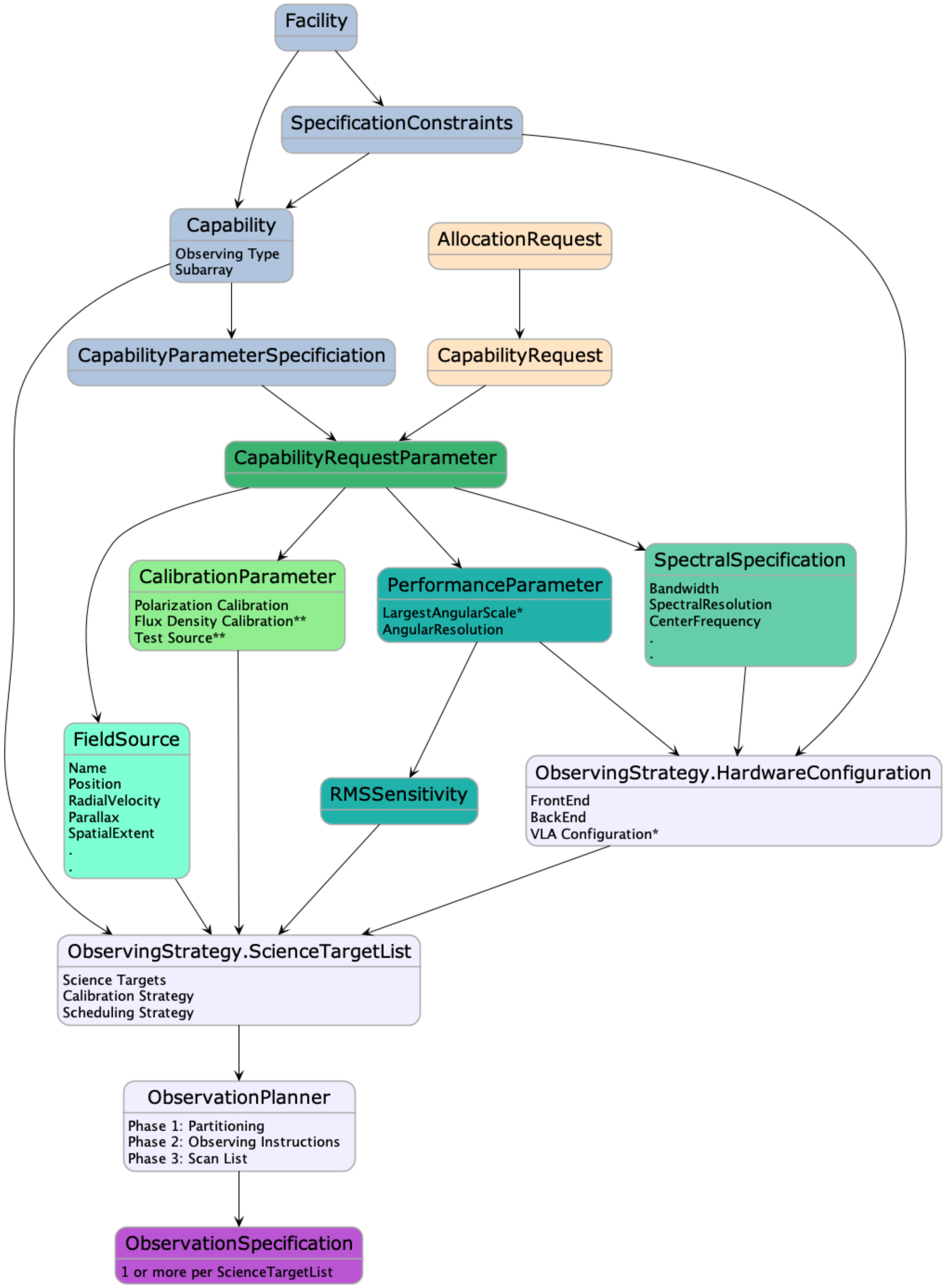


Figure 1.1: Simple diagram to illustrate the relationships between the *Capability Request Parameters*, which are shown in shades of green, and the algorithm (Services; shown in lavender). The algorithm constructs the *Observation Specification* (dark purple), by folding in the *Facility* specific information, which includes the *Capability* (light blue) and the *Specification Constraints*.The parameters marked with an asterisks may not be applicable to all *Facilities* and *Capabilities*.

Table 1.1: FIELD SOURCE per *Capability*

Capability Parameter Specification	Example	VLA Continuum	VLA Spectral Line	GBT Continuum	…
Name[a] (str)	3C286	✓	✓	✓	…
Coordinate System (str)	FK5	✓	✓	✓	…
Equinox (str)	J2000	✓	✓	✓	…
Position (float)	(α, δ) = (02:28:30.87, +19:20:45.53) (ℓ,b) = (152.3754°, -37.9133°)	✓	✓	✓	…
Position Uncertainty (float)		✓	✓	✓	…
Field of View (float)	1 sq deg	✓	✓	✓	…
Radial Velocity (float)	837 km s^{-1}	✓	✓	✓	…
Velocity Reference Frame	LSRK	✓	✓	✓	…
Doppler Type	Radio	✓	✓	✓	…
Parallax (float)	1mas	✓	✓	✓	…
Proper Motion	1mas/yr	✓	✓	✓	…
Peak Continuum Flux Density per Synthesized Beam	5 mJy	✓	✓	✓	…
Peak Line Flux Density per Synthesized Beam	1 mJy		✓		…
⋮	⋮	⋮	⋮	⋮	…

[a] This parameter can be any user supplied string. The naming convention is for the ease of identification for the user; it will be propagated to the *Observation Specification* for identification purposes only.

Table 1.2: SPECTRAL SPECIFICATION per *Capability*

Capability Parameter Specification	Example	VLA Continuum	VLA Spectral Line	GBT Continuum	…
Name[a] (str)	C-band	✓	✓	✓	…
Center Frequency (float)	5 GHz	✓	✓	✓	…
Bandwidth (float)	2.048 GHz	✓	✓	✓	…
Spectral Resolution (float)	1.5 kHz	✓	✓	✓	…

[a] This parameter can be any user supplied string.

Definitions and Concepts –

- See Section 1.4 of 688-TTAT-004-MGMT System Description v3.0
- A CALIBRATOR is Coordinate information for an observation that includes position, field of view, velocity, and time (when ephemerides are required).

Table 1.3: PERFORMANCE PARAMETERS per *Capability*

Capability	Largest Angular Scale (float; $10''$)	Angular Resolution (float; $0.5''$)	RMS Sensitivity (float; 5.0 μJy bm^{-1})
GBT Continuum		✓	✓
GBT Spectral Line		✓	✓
GBT Pulsar		✓	✓
GBT Radar		✓	✓
VLA Continuum	✓	✓	✓
VLA Spectral Line	✓	✓	✓
VLA Pulsar	✓	✓	✓

Table 1.4: CALIBRATION PARAMETERS per *Capability*

Capability	Flux Density (boolean)	Test Source (boolean)	Polarization (boolean)
GBT Continuum	✓	✓	✓
GBT Spectral Line	✓	✓	✓
GBT Pulsar	✓	✓	✓
GBT Radar	✓	✓	✓
VLA Continuum			✓
VLA Spectral Line			✓
VLA Pulsar	✓	✓	✓

Part II

The Observing Strategy

Chapter 2

Overview of the Observing Strategy Algorithm

Overview – The *Observing Strategy* algorithm translates the user specified, science driven *Capability Request* into an intermediate, normalized data structure called a *Science Target List*. If multiple *Capability Requests* exist in an *Allocation Request*, they are realized in a single *Science Target List*. The *Observing Strategy* specifies the Science Targets, the Calibration Strategy, and the Scheduling Strategy to create the *Science Target List*, which is passed downstream to the Observation Planner.

Definitions and Concepts –

- 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.
- A Pointing Pattern describes the trajectory of an antenna over the course of an observation of a FIELD SOURCE. Pointing Patterns are *Facility* dependent.
- 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.
- 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 (Section 15.1.3).
- A Calibration Strategy is an input and a selector for *Observation Planner Phase 1, Phase 2*, and *Phase 3*. It contains the rules for calibrating the Science Target. See Chapter 5.
- A Scheduling Strategy is an input and a selector for *Observation Planner Phase 1, Phase 2*, and *Phase 3*. It contains the rules for scheduling the Science Target. See Chapter 5.

The *Observing Strategy* algorithm generates a normalized data structure called the *Science Target List*, which contains the fundamental user request. The *Science Target List* consists of rows of Science Targets, which have

- a Source;
- a Hardware Configuration;

- a Requested Time;
- a Repeat Count, if applicable;
- a Calibration Strategy;
- a Scheduling Strategy.

The generation of the *Science Target List* is *Capability* specific, but the steps in the *Observing Strategy* are shown in Figure 2.1. The Chapters 3 – 4 detail the specifics of the *Observing Strategy* for the GBT, VLA, VLBA, GMVA, and HSA.

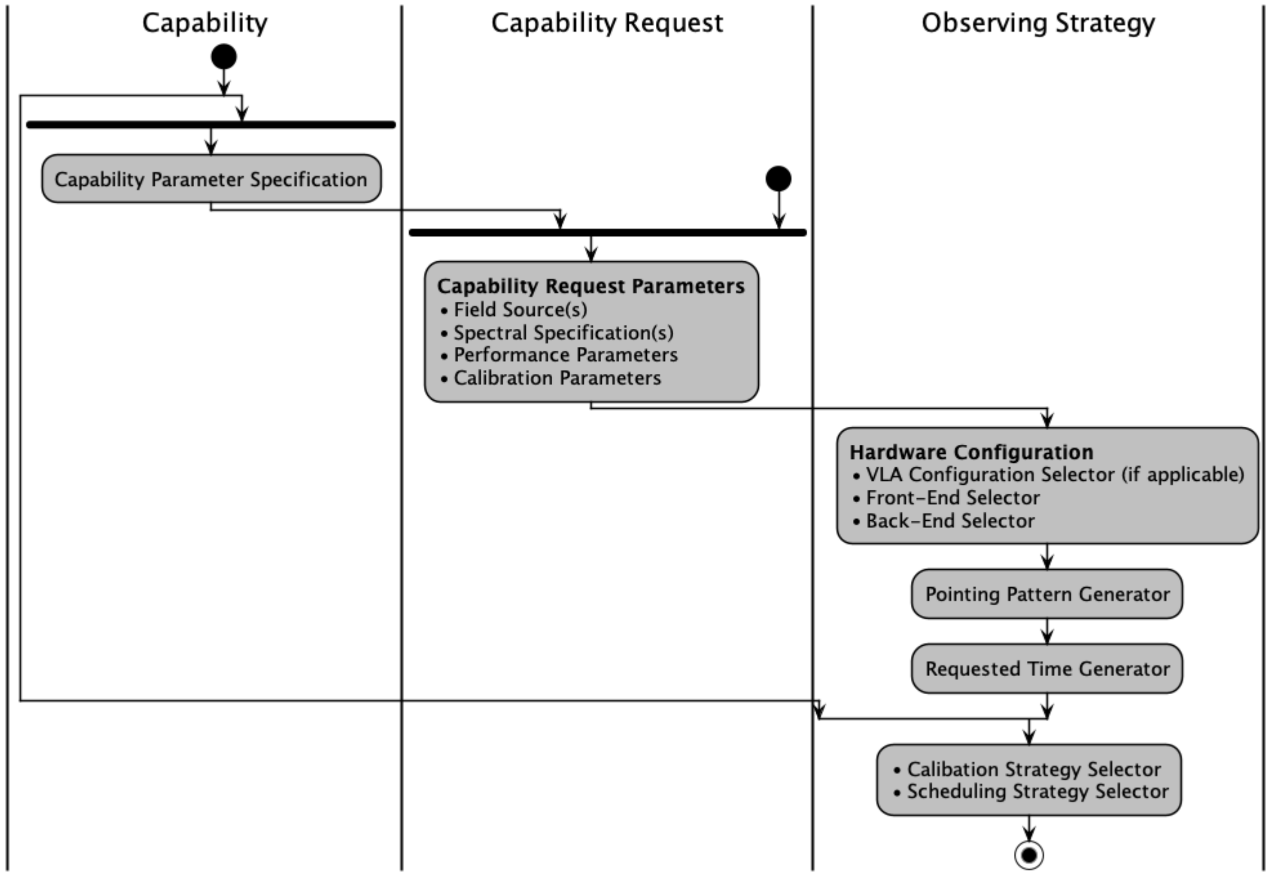


Figure 2.1: Overview of the algorithms implemented by the *Observing Strategy*.

Chapter 3

VLA Observing Strategy

Table 3.1: VLA Receivers and BACK-ENDS: capabilityConfig.json

here's a test table

The VLA *Observing Strategy* takes the inputs of

1. *Capability Request Parameters*
    (a) Tables 1.1, 1.2, 1.3, and 1.4
2. *Capability*
    (a) Table 3.1
3. *Specification Constraints*
    (a) Diameter of Antenna Dish (D_{dish}; m)
    (b) Illumination Taper Edge (T_e; dB)
    (c) Settle Time (s)

The Observing Types include VLA Continuum, VLA Spectral Line, and VLA Pulsar. For all Observing Types, the *Observing Strategy* will implement the algorithms in the order given in Figure 2.1. Once an Observing Type is specified at the *Capability Request* level, the *Capability* selects the appropriate family of *Observing Strategy* algorithms. Table 3.2 lists relevant sections for the *Observing Strategy* algorithms, and Table 3.3 presents the same information as Table 3.2 but organized by *Observing Type*.

To reiterate, once an *Observing Type* is specified, the family of algorithms for the *Observing Strategy* is specified. For example, if "VLA Continuum" is selected in the *Capability Request*, the *Observing Strategy* uses the algorithms referenced in column 2 of Table 3.3 to construct the *Science Target List*. A "VLA Pulsar" capability follows column 4.

3.1 VLA Hardware Configuration

The VLA Hardware Configuration 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 3.1.

Table 3.2: Overview of VLA Observing Strategy.

Algorithm			§
VLA Hardware Configuration			
	VLA Configuration Selector		3.1.1
	VLA FRONT-END Selector		
		VLA Continuum	3.1.2
		VLA Spectral Line	3.1.3
		VLA Pulsar	3.1.4
	VLA BACK-END Selector		
		VLA Continuum	3.1.5
		VLA Spectral Line	3.1.6
		VLA Pulsar	3.1.7
VLA Pointing Pattern Generator			3.2
VLA Requested Time Generator			3.3
VLA Strategy Selector			3.4

Table 3.3: VLA *Observing Strategy* Algorithms by *Observing Types*

Algorithm	VLA Continuum §	VLA Spectral Line §	VLA Pulsar §
VLA Configuration Selector	3.1.1	3.1.1	3.1.1
VLA FRONT-END Selector	3.1.2	3.1.2	3.1.4
VLA BACK-END Selector	3.1.5	3.1.6	3.1.7
VLA Pointing Pattern Generator	3.2	3.2	3.2
VLA Requested Time Generator	3.3	3.3	3.3
VLA Strategy Selector	3.4	3.4	3.4

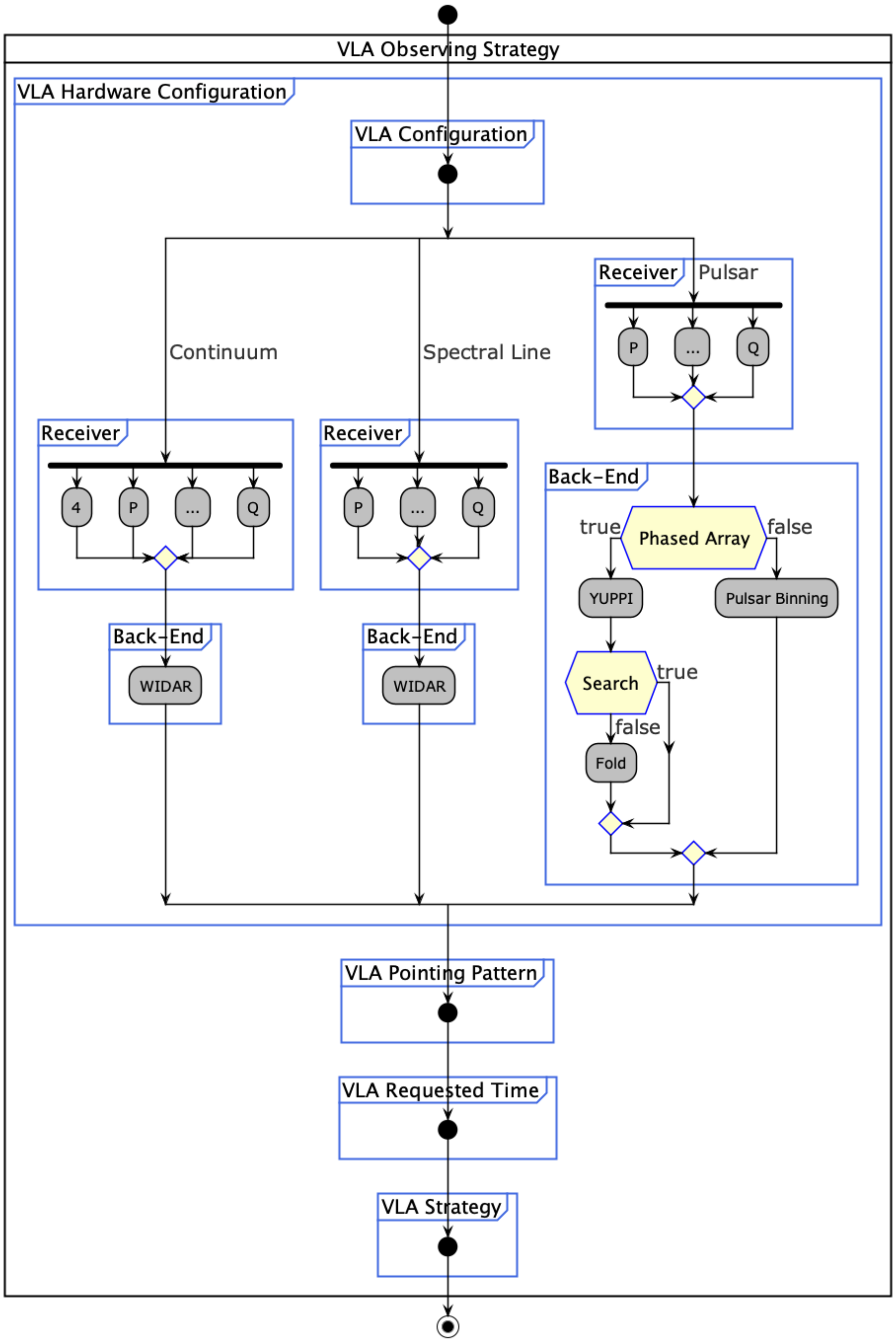


Figure 3.1: Diagram of algorithm to select the VLA Configuration, the FRONT-END, and the BACK-END.

3.1.1 VLA Configuration Selector

- PLACEHOLDER: Algorithm

3.1.2 VLA Continuum front-end Selector

- PLACEHOLDER: Algorithm

3.1.3 VLA Spectral Line front-end Selector

- PLACEHOLDER: Algorithm

3.1.4 VLA Pulsar front-end Selector

- PLACEHOLDER: Algorithm

3.1.5 VLA Continuum back-end Selector

- PLACEHOLDER: Algorithm

3.1.6 VLA Spectral Line back-end Selector

- PLACEHOLDER: Algorithm

3.1.7 VLA Pulsar back-end Selector

- PLACEHOLDER: Algorithm

3.2 VLA Pointing Pattern Generator

Each *Capability Request* will call the VLA Pointing Pattern to determine which Pointing Pattern is needed, regardless of *Observing Type*. The VLA patterns include the following:

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

The following outlines the steps needed to determine which Pointing Pattern to employ, which is diagrammed in Figure 3.2.

1. The qualitative condition for a Single Pointing pattern is when the Primary Beam, θ_{PB}, is much greater than Ω_{FOV}:

   $$\theta_{\mathrm{PB}} > \text{scalar} \times \ max(\Omega_{\mathrm{FOV}}), \tag{3.2.1}$$

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

   $$\theta_{\mathrm{PB}} = (1.02 + 0.0135\, T_e) \times \frac{c}{\nu} \times \frac{1}{\mathrm{D_{dish}}} \ \text{rad}, \tag{3.2.2}$$

   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 upper frequency (Hz) of the requested bandwidth.

2. If Eq (3.2.1) is True, the Single Pointing pattern is used (go to Section 3.2.1).
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 Integration Time, t_{eff}, the overhead becomes large or

$$t_{eff} < XX\text{s}. \tag{3.2.3}$$

        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 3.2.2).
    (c) If both conditions are True, OTF mapping is used (go to Section 3.2.3).

Text in Figure 3.2	Condition	Reference in Text
(1) Condition for Single Pointing	$\theta_{\rm PB} >$ scalar $\times$ $max(\Omega_{\rm FOV})$	§3.2
(2) Condition for OTF	$t_{eff} <$ XX s AND Data Rate $<$ XX	§3.2
(3) Validate OTF RA Spatial Extent	$\Omega_{\rm RA} >$ XX	§3.2.3
(4) Validate *scan rate*	$scan\ rate < 3$ arcmin s^{-1}	§3.2.3
(5) Validate Dump Time	$t_{\rm dump} < 0.6$ s	§3.2.3

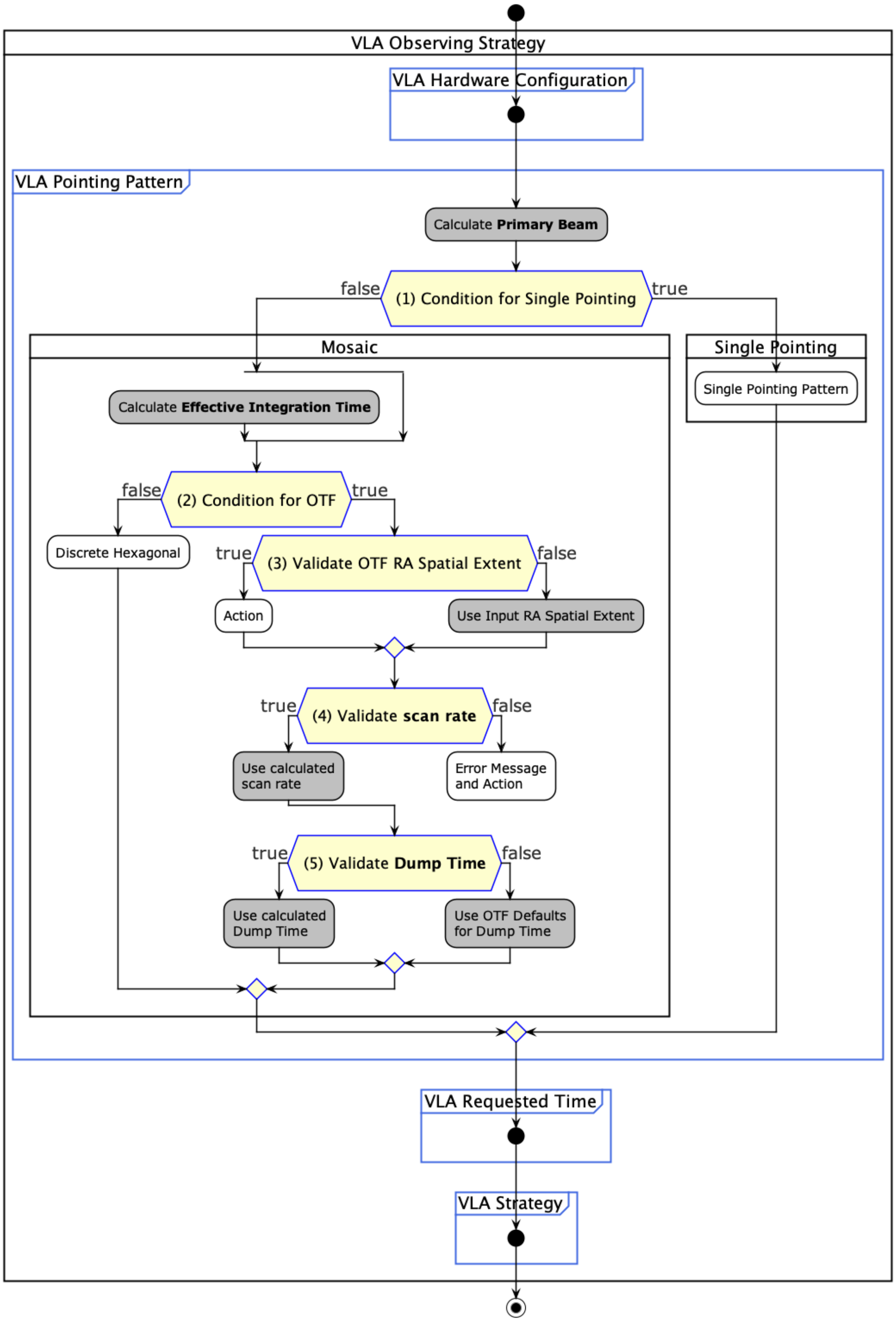


Figure 3.2: VLA *Observing Strategy*

3.2.1 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

3.2.2 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 theFIELD 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 (Ω_{RA}). The angular extent in Dec (Ω_{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.

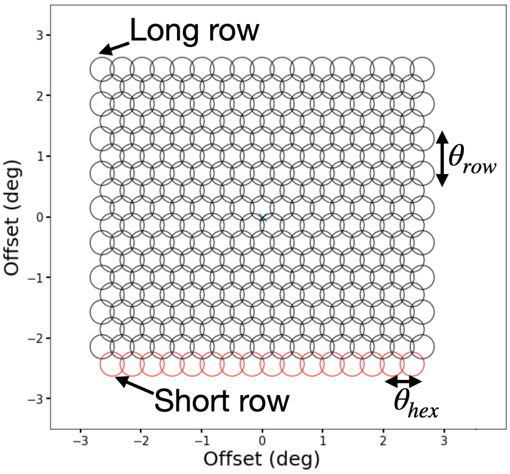


Figure 3.3: Example of a VLA Hexagonal Pointing Pattern. The variables, terms, and circles in red are described in the text.

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 (Ω_{RA}) and that in Dec (Ω_{Dec}) are

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

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

   respectively, rounded up to the nearest integer. θ_{hex} and θ_{row} are defined as

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

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

      where

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

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

      ii. The angular spacing between the rows of the mosaic

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

   (b) The pattern is constructed by alternating rows of different lengths, called short rows and long rows. The rows are offset in Declination by θ_{row}. An example of a short row is highlighted in red in Figure 3.3.

   (c) TODO – Someone probably has a better way on implementing the pattern than my code does. Have at. Here are the rules though:

      i. Long rows consist of $\mathrm{n}_{\mathrm{ra}}+1$ pointings, each pointing offset in RA by θ_{hex}.
      ii. Short rows consist of n_{ra} pointings, each pointing offset in RA by θ_{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 (§15.1.2.1) to determine the Effective Integration Time, t_{eff}.
   (b) The Requested Time Per Pointing is

$$\text{Requested Time Per Pointing} = \frac{t_{eff}}{n_{tot}},$$

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

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

   rounded up to the nearest integer.

3. The additional parameters that are included in the *Observation Specification* for reporting purposes are
   (a) Mosaic Beam Area:

$$\Omega_{beam} = 0.5665\ \theta_{\text{PB}}^2 \tag{3.2.4}$$

   (b) Requested Time (no overhead):

$$\text{Requested Time} = n_{tot} \times \text{Requested Time Per Pointing} \tag{3.2.5}$$

   (c) Survey Speed:

$$\text{Survey Speed} = \frac{\Omega_{beam}}{t_{eff}} \tag{3.2.6}$$

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

3.2.3 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.

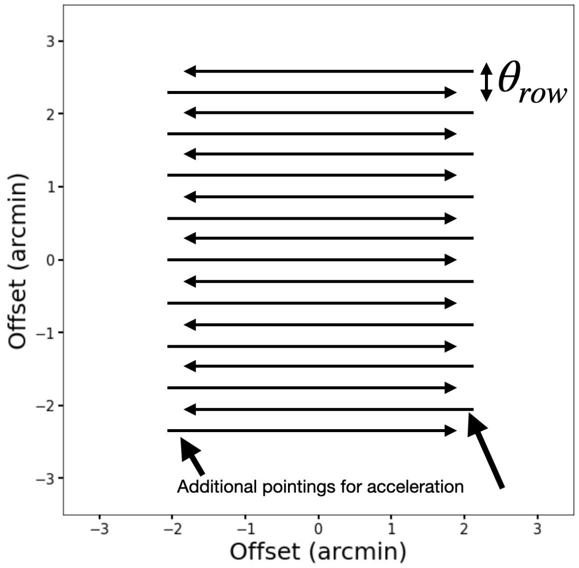


Figure 3.4: Example of VLA OTF mapping for a small Ω_{ra}. The offset in Dec is equal to θ_{row}.

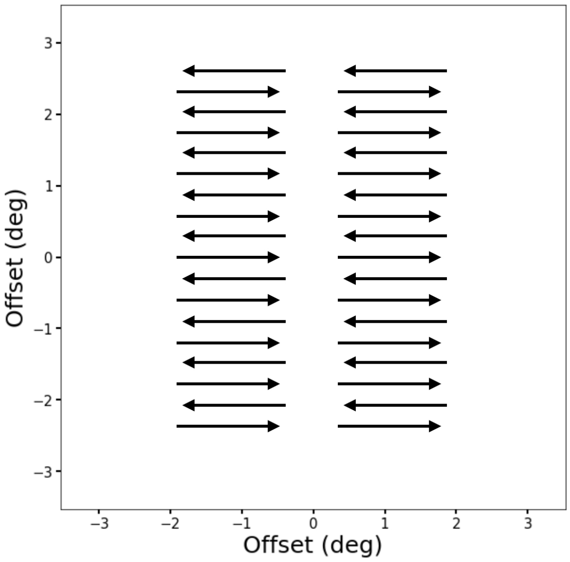


Figure 3.5: Example of VLA OTF mapping for a large Ω_{ra}. The offset in RA between the tracks is artificially inflated. The offset in Dec is between rows is equal to θ_{row}.

1. If the angular extent in RA, $\Omega_{\rm ra}$, 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 $\Omega_{\rm ra}$, multiple OTF patterns are constructed instead of a single OTF pattern. In a sub-pattern, all the rows needed to span the requested angular extent in Declination are observed beFigures 3.4 and 3.5 show examples of OTF pointings for a small and large angular extent, respectively. The condition for splitting up the OTF pattern is when

   $$\Omega_{\rm RA} \; > \; XX, \tag{3.2.7}$$

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

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

   (a) If this condition is met, $\Delta_{\rm RA}$ should be used in place of $\Omega_{\rm RA}$ 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

   $$scan\ rate = \frac{[0.5665\ \theta_{\rm PB}^2]}{t_{eff} \times \theta_{\rm row}}. \tag{3.2.8}$$

   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, $\theta_{\rm PB}$. The Effective Integration Time is t_{eff} (see Section 15.1.2.1), and $\theta_{\rm row}$ is the angular spacing between rows (stripes):

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

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

   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 3.6 illustrates the relationship between t_{eff}, θ_{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_{dump}, for OTF mapping is

$$t_{\text{dump}} \sim 0.1 \times \frac{\theta_{\text{PB}}}{scan\ rate}. \tag{3.2.9}$$

(d) The algorithm performs a check of value of t_{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

$$n_{integ} = \frac{t_{dump,default}}{t_{\text{dump}}}, \tag{3.2.10}$$

otherwise, $n_{integ} = 1$.

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

$$n_{\text{stripe}} = \frac{\Omega_{ra}}{[scan\ rate \ \times \ n_{integ} \times t_{dump}]} \tag{3.2.11}$$

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

(g) The Time Per Row is then

$$\text{Row Duration} = (n_{stripe} + 1) \times n_{integ} \ \times \ t_{dump}. \tag{3.2.12}$$

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 in 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

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

and

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

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

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

and

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

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

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

and

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

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

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

   and is the number of rows required to span Ω_{Dec}.

4. There are additional parameters that the User will want displayed, that are useful for validation purposes, or may need to be accessible to other portions of the algorithm.
   (a) Beam Area: Equation 3.2.4
   (b) Survey Speed: Equation 3.2.6
   (c) If Condition 3.2.7 is True, the number of OTF sub-patterns, N, and the extent in RA of each sub-pattern, Δ_{RA}.
   (d) Scan Rate: Equation 3.2.8
   (e) Dump Time + Validation Check: Equation 3.2.9
   (f) Number of Integrations Per Step: Equation 3.2.10
   (g) Requested Time, which does not include overhead:

$$\text{Requested Time} = n_{rows} \times \text{Time Per Row} \tag{3.2.13}$$

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 time needs to be accounted for.

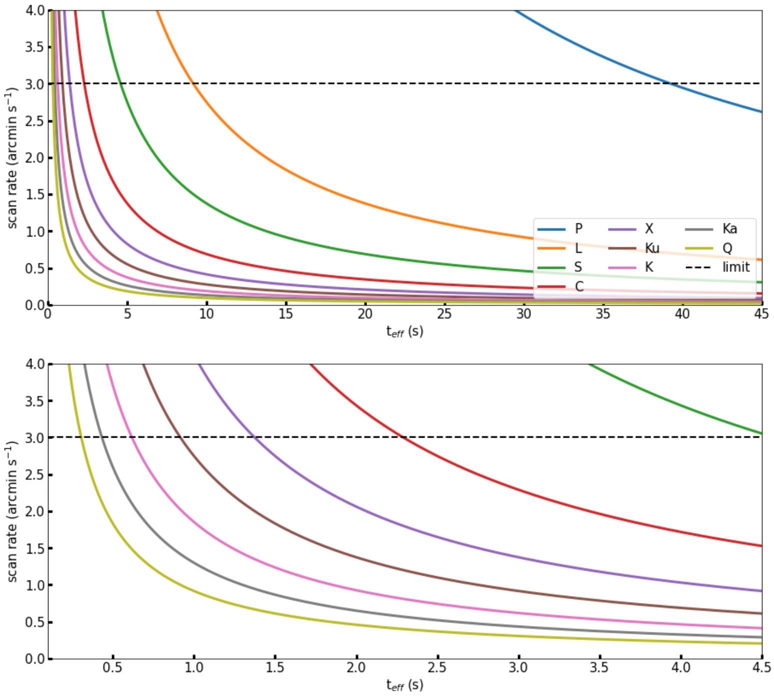


Figure 3.6: *Top.* The *scan rate* as a function of different Effective Integration Times and VLA frequency bands, as represented by the different colored curves. *Bottom.* Same as the top plot except the range of the abscissa is truncated to provide better detail.

3.3 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 Concept Document for information about this
    (a) uv-coverage for an interferometer
    (b) event occurrence rate
    (c) sensitivity
    (d) “etc”
2. uv-coverage is concern: how to tell the algorithm to observing for e.g., 30 minutes across an 8 hour LST range.
3. Pulsars don’t care about RMS (really) but they do want to repeat count at specific times
4. If overhead dominated (rule? for VLA $<$ 50% for example), then increase integration time. How to get the algorithm to do this.
5. Does there need to be a performance parameters for Dynamic range or Parallactic Angle?

3.4 VLA Strategy Selector

Beginning Scaffolding for this section...

- How many unique Sources
- How many unique Hardware Configurations
- Total Requested Time
- Logic for input for a Frequency Partition Instruction

Chapter 4

GBT Observing Strategy

The GBT *Observing Strategy* takes the inputs of

1. *Capability Request Parameters*
   (a) Column 5 of Tables 1.1 and 1.2; row 1 of 1.3 and 1.4
2. *Capability*
   (a) Table 4.1
3. *Specification Constraints*
   (a) Diameter of Antenna Dish (D_{dish}; m)
   (b) Illumination Taper Edge (T_e; dB)
   (c) Settle Time (s)

Table 4.1: GBT Receivers and BACK-ENDS: capabilityConfig.json

here's a test table

Table 4.2: Overview of GBT Observing Strategy.

Algorithm	§
GBT Hardware Configuration	
GBT FRONT-END Selector	
GBT Continuum	4.1.1
GBT Spectral Line	4.1.2
GBT Pulsar	4.1.3
GBT Radar	4.1.4
GBT BACK-END Selector	
GBT Continuum	4.1.5
GBT Spectral Line	4.1.6
GBT Pulsar	4.1.7
GBT Radar	4.1.8
GBT Pointing Pattern Generator	4.2
GBT Requested Time Generator	4.3
GBT Strategy Selector	4.4

Table 4.3: GBT *Observing Strategy* Algorithms by *Observing Types*

Algorithm	GBT Continuum §	GBT Spectral Line §	GBT Pulsar §	GBT Radar §
GBT FRONT-END Selector	4.1.1	4.1.1	4.1.3	4.1.4
GBT BACK-END Selector	4.1.5	4.1.6	4.1.7	4.1.8
GBT Pointing Pattern Generator	4.2	4.2	4.2	4.2
GBT Strategy Selector	4.4	4.4	4.4	4.4

4.1 GBT Hardware Configuration

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 4.1.

4.1.1 GBT Continuum front-end Selector

- PLACEHOLDER: Algorithm

4.1.2 GBT Spectral Line front-end Selector

- PLACEHOLDER: Algorithm

4.1.3 GBT Pulsar front-end Selector

- PLACEHOLDER: Algorithm

4.1.4 GBT Radar front-end Selector

- PLACEHOLDER: Algorithm

4.1.5 GBT Continuum back-end Selector

- PLACEHOLDER: Algorithm

4.1.6 GBT Spectral Line back-end Selector

- PLACEHOLDER: Algorithm

4.1.7 GBT Pulsar back-end Selector

- PLACEHOLDER: Algorithm

4.1.8 GBT Radar back-end Selector

- PLACEHOLDER: Algorithm

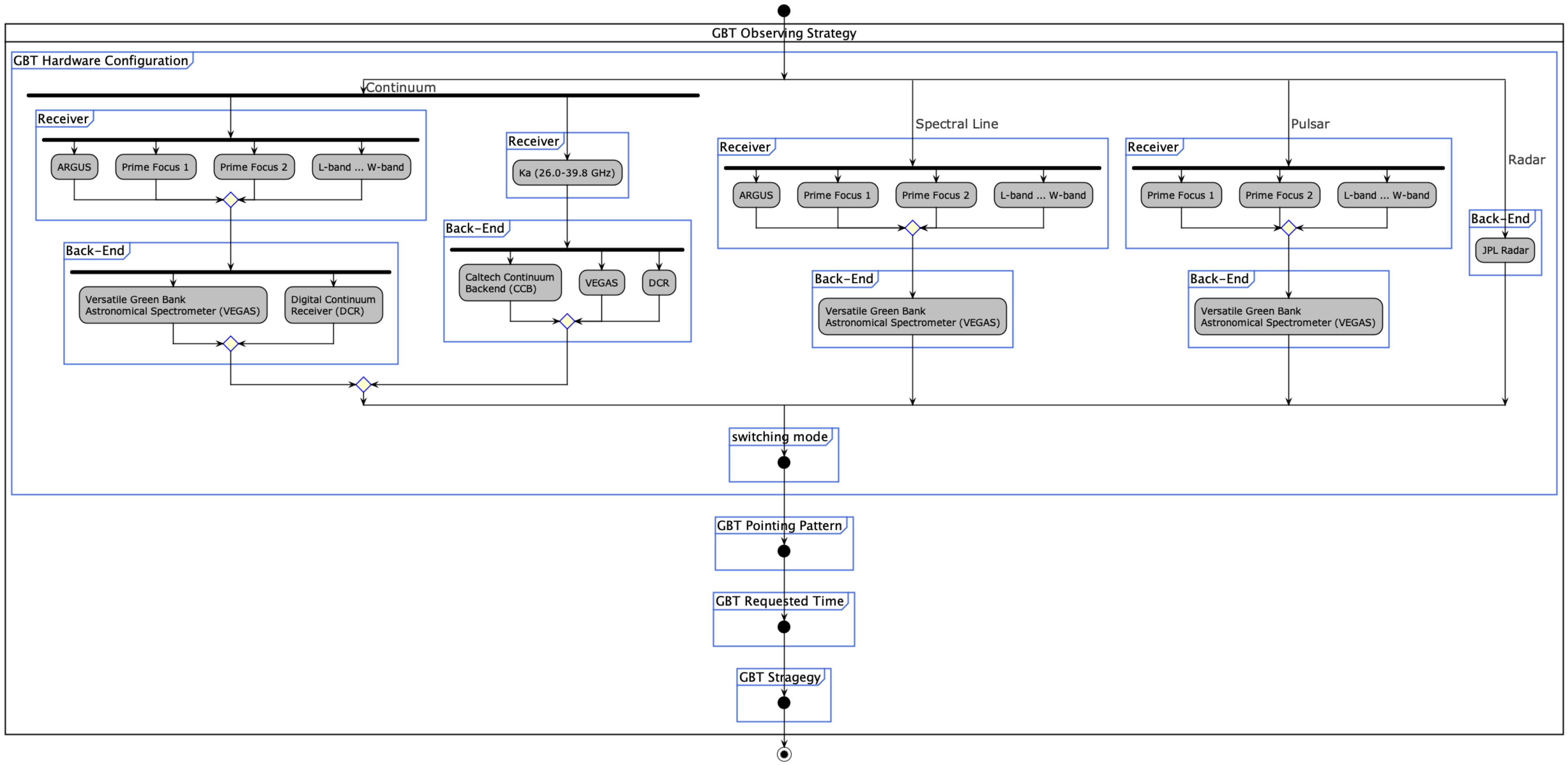


Figure 4.1: Diagram of algorithm to select the FRONT-END for the GBT and the BACK-END.

4.2 Pointing Patterns for the GBT

Dana: It may not be the best way to say single pointing vs mosaic for the GBT.

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

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

The condition for Single Pointing is when the Primary Beam, θ_{PB}, is much greater than the Field of View. The Primary Beam is compared to the largest dimension of Ω_{FOV}.

$$\theta_{\text{PB}} > \text{scalar} \times \ max(\Omega_{\text{FOV}}), \tag{4.2.1}$$

where scalar = XX. The Primary Beam of the telescope is

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

or

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

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 4.

If condition (4.2.1) is not met, i.e., the primary beam is smaller than the Field of View of the source, then a mosaic pattern is considered: a discrete mosaic or OTF mapping. The algorithm decides between OTF mapping and a discrete mosaic based on two conditions.

---

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

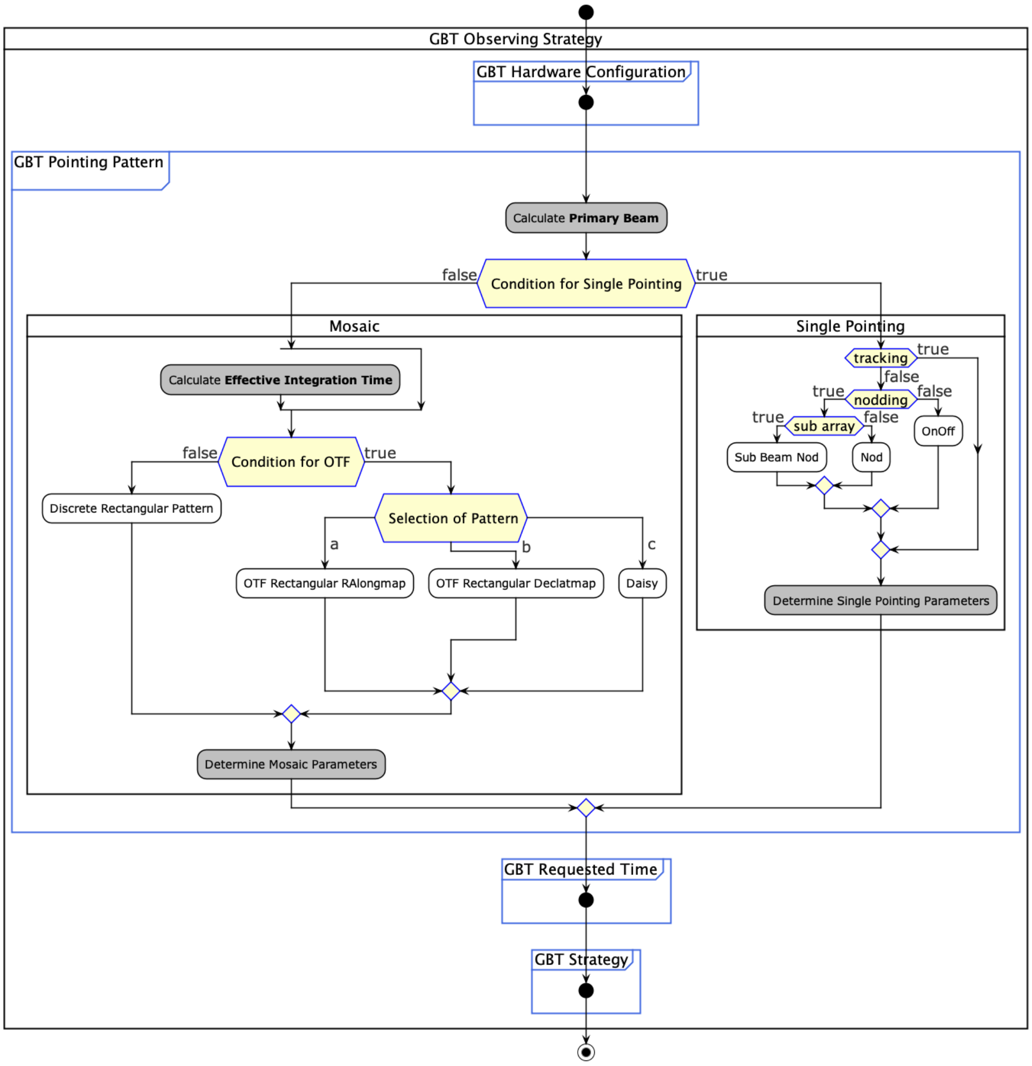


Figure 4.2: GBT *Observing Strategy*

Condition for Figure 4.2		Section
Condition for Single Pointing	$\theta_{\mathrm{PB}} >$ scalar $\times$ $max(\Omega_{\mathrm{FOV}})$	4.2
Condition for OTF	???	4.2
Selection of Discrete Pattern	???	4.2.2
Tracking	???	4.2.1
Nodding	???	4.2.1
Sub array	???	4.2.1

4.2.1 GBT Single Pointing

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

4.2.2 GBT Discrete Mosaic

tbd

4.2.3 GBT OTF

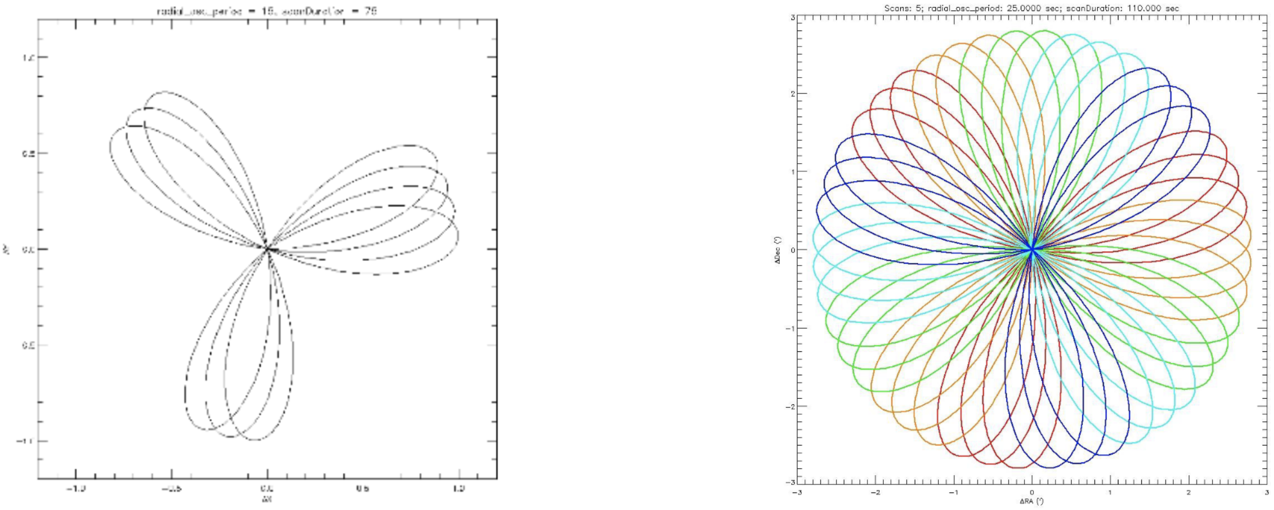
tbd

4.2.3.1 GBT RAlongmap and Declatmap

Details tbd.

4.2.3.2 GBT Daisy Map

A Daisy map scans continuously around a central point as shown in Figure 4.3. This pattern is discussed in detail in Section 6.4.3.7 of *Observing with the Green Bank Telescope*[2].



(a) (b)

Figure 4.3: GBT Daisy maps at 5 radial oscillations (a) and 22 radial oscillations (b), which is a closed pattern.

---

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

4.3 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

4.4 GBT Strategy Selector

Part III

The Calibration and Scheduling Strategies

Chapter 5

Overview of Calibration and Scheduling Strategies

A *Observing Strategy* Strategy Selector algorithm (e.g., § 3.4, 4.4) selects the appropriate Calibration Strategy and Scheduling Strategy for the Science Targets in a single *Science Target List.* The Calibration Strategy and Scheduling Strategy, fundamentally, are 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. Note, the *Observing Strategy* is an actor: it selects a Calibration Strategy and Scheduling Strategy, whereas a Calibration Strategy and Scheduling Strategy are selected and take no action.

A Calibration Strategy and a Scheduling Strategy are inputs and selectors for the *Observation Planner Phase 1, Phase 2*, and *Phase 3.* A Calibration Strategy contains a Partition Plan and a Calibration Plan. 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. The functional descriptions of the Partition Instructions and Observing Instructions are in Chapters 11 and 12, respectively. Similarly, a Scheduling Strategy contains the instructions for creating the Scan List.

Chapters 6, 7, 8, and 9 are a review of the strategies available to the algorithm, as Calibration and Scheduling Strategies are dependent on the *Capability.* As the *Observing Strategy* Selector algorithm selects the appropriate Calibration Strategy and Scheduling Strategy on a per *Capability* basis (see Tables 3.2, 3.3, 4.2, and 4.3), Table 5.1 provides a similar overview of the strategies available to a *Observing Strategy* Strategy Selector algorithm.

Table 5.1: Overview of Calibration and Scheduling Strategies.

Strategy	§
VLA Continuum Calibration Strategy	6.1
VLA Spectral Line Calibration Strategy	6.2
VLA Pulsar Calibration Strategy	6.3
VLA Continuum Scheduling Strategy	8.1
VLA Spectral Line Scheduling Strategy	8.1
VLA Pulsar Scheduling Strategy	8.2
GBT Continuum Calibration Strategy	7.1
GBT Spectral Line Calibration Strategy	7.2
GBT Pulsar Calibration Strategy	7.3
GBT Radar Calibration Strategy	7.4
GBT Continuum Scheduling Strategy	9.1
GBT Spectral Line Scheduling Strategy	9.2
GBT Pulsar Scheduling Strategy	9.3
GBT Radar Scheduling Strategy	9.4

Chapter 6

VLA Calibration Strategies

The following sections discuss the VLA specific strategies available to a VLA Strategy Selector algorithm (§ 3.4). The algorithm will select a suitable Partition Plan and Calibration Plan.

6.1 VLA Continuum Calibration Strategy

Section 3.1 of 688-TTAT-xxx-MGMT TTA Use Cases v0.1 details a VLA Continuum use case. ... extracted best practices are...

- Partition Plan
  - Create a Priority Partition Instruction(§ 11.2.1).
  - Create a VLA Configuration Partition Instruction (§ 11.2.2).
  - Create a Frequency Partition Instruction (§ 11.2.3).
  - Create a Distance Partition Instruction (§ 11.2.4).
  - Create a Time Partition Instruction (§ 11.3).
- Calibration Plan
  - For each Science Target in the *Science Target List* create a Science Target Observing Instruction with the Source, Hardware Configuration, and Requested Time from the partitioned set (§ 12.1.1).
  - Create a Phase Referencing Observing Instruction for each cluster of Science Observing Instructions (§ 12.1.1.2). The Cycle Time should be set according to Table 14.1.
  - For each Phase Referencing Observing Instruction, select an appropriate calibrator (§ 15.1.3) and create a Calibrator Observing Instruction (§ 12.1.2) 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.

    For now should it should be set to 180 seconds.
  - Select Observing Targets appropriate for the intents of CALIBRATE_FLUX and CALIBRATE_BANDPASS for each unique Hardware Configuration (§ 15.1.3). Create Calibrator Observing Instructions with the execute "once flag" set for each of the selected calibrators. The Acquisition Time is 180 seconds.
  - If the CALIBRATION PARAMETER of "Polarization Study" is set, select an appropriate Observing Target for the intents of CALIBRATE_POL_LEAKAGE and create a Calibrator Observing Instruction with an Acquisition Time of XX seconds (§ 15.1.3).
  - If any of the Hardware Configurations have frequencies of 15 GHz or above, create a VLA Pointing Observing Instruction 12.1.2.1.1. Set this as a prerequisite for all Observing Instruction that have Hardware Configurations above 15 GHz.

6.2 VLA Spectral Line Calibration Strategy

6.3 VLA Pulsar Calibration Strategy

Chapter 7

GBT Calibration Strategies

7.1 GBT Continuum Calibration Strategy

7.2 GBT Spectral Line Calibration Strategy

- Partition Plan
  - Create a Priority Partition Instruction (§ 11.2.1).
  - Create a Frequency Partition Instruction (§ 11.2.3).
  - Create a Distance Partition Instruction (§ 11.2.4).
- Observing Instruction Creation
  - Each of the partitioned sets of Science Targets should be treated independently going forward.
  - If each of the Science Targets in the set has a Pointing Pattern equal to Single Pointing, create a Position Switching Observing Instruction (§ 12.1.1.1) 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 14.1.
  - For each unique Hardware Configuration, create a Calibrator Observing Instruction (§ 12.1.2) with the single Scan Intent of CALIBRATE_FLUX and a single Sub-scan Intent of ON_SOURCE. The Hardware Configuration should be the same as for the Science Target.
  - Create a GBT Pointing Calibrator OI (§ 12.1.2.1.2) and a GBT Focus Calibrator OI (§ 12.1.2.1.3) for each unique Hardware Configuration and set them as prerequisites for the Calibrator and Position Switched Observing Instructions that share the Hardware Configuration.

7.3 GBT Pulsar Calibration Strategy

7.4 GBT Radar Calibration Strategy

Chapter 8

VLA Scheduling Strategies

8.1 VLA Spectral Line Scheduling Strategy

8.2 VLA Pulsar Scheduling Strategy

Chapter 9

GBT Scheduling Strategies

9.1 GBT Continuum Scheduling Strategy

9.2 GBT Spectral Line Scheduling Strategy

9.3 GBT Pulsar Scheduling Strategy

9.4 GBT Radar Scheduling Strategy

Part IV

The Observation Planner

Chapter 10

Overview of Observation Planner

Overview – 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.*

Hierarchical Definitions and Concepts –

- Acquisition Time is the time an antenna spends taking data in a Sub-scan;
- 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 (§ 15.1.1) is the sum of the antenna slew time + settle time + hardware configuration overhead;
- Sub-scan
  - Specification of the shortest, contiguous block of time over which an antenna is taking data. Each Sub-scan 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 Sub-scan Intent (Table 10.1).
- Sub-scan Prototype
  - A sub-scan that has a Source, a Hardware Configuration, and Sub-scan Intents but does not include an Acquisition Time or a Setup Time.
- Scan Intent
  - A tag that describes the scientific purpose of a set of Sub-scans, e.g., a flux, phase, or bandpass calibration, a pointing, an observation of a Science Target. A single scan can have multiple Scan Intents. See Table 10.1.
- Scan
  - A group of Sub-scans that share Scan Intent. All scans have at least one sub-scan.

- Maximum Duration
    - Maximum length for any single Scan including all associated Setup Times on an Observing Target. The maximum length of time allowed for a Sub-Scan, Scan, all Sub-scans, or all Scans.
- Maximum Acquisition Time
    - The maximum time of any single Scan on an Observing Target
- Requested Time
    - The time specified for a Science Target in the *Science Target List.*
- Science Target Integration Time(s)
    - The sum of the Acquisition Times for all Sub-scans on a Science Target with Sub-scan 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 Sub-scans are complete.
    - The sum of all Acquisition Times for all Sub-scans for all Science Targets with Sub-scan Intent ON_SOURCE and associated with a Scan Intent of OBSERVE_TARGET (Scan List level).
- Observing Target
    - The generalization of a Science Target to include Calibrators; it consists of
        - a Hardware Configuration,
        - a Source.
    - All Science Targets are Observing Targets, but not all Observing Targets are Science Targets.
- Time on Observing Target(s)
    - The sum of the Acquisition Times for all Sub-scans on 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 Sub-scans for all Observing Targets (Scan List level).
- Duration
    - Generally, a duration is the total time of a Sub-Scan, Scan, all Sub-scans, or all Scans. This includes overheads. Specifically, it is the total time of all Scans, and Duration = Setup Time + Time on Observing Targets.
- Overhead
    - Generally, the overhead is any time an antenna is not collecting data on a Science Target. Specifically, Overhead = Duration - Science Target Integration Times

Table 10.1: Table of Scan and Sub-Scan Intents

Intent	Comment	GBT	VLA	VLBA
UNSPECIFIED	Sub-Scan; Scan	✓	✓	
ON_SOURCE	Sub-scan	✓	✓	
OFF_SOURCE	Sub-scan	✓	✓	
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		✓	

Table 10.2: Interrelated Terms in *Capability Request Parameters* and *Observation Specification*

	Science		Observation
Spatial	Field Source	$\xrightarrow[\text{Strategy}]{\text{Observing}}$	Source
Spectral	Spectral Spec		Hardware Configuration
Time	Requested Time[a]	$\xrightarrow[\text{Planner}]{\text{Observation}}$	Science Target Integration Time

[a] The Requested Time is determined by the *Observing Strategy.*

The *Capability Request* contains the user request. Through the *Observing Strategy* and the *Observation Planner*, that request is translated into the *Observation Specification.* Table 10.2 presents the mapping of key terms between these two parts of the *Allocation Request.* Figure 10.1 presents the hierarchy of the time related concepts in the *Observation Planner.* To (hopefully) clarify the definitions and concepts above, consider the two simple, but contrived, examples in the Section 10.1 prior to reviewing the in depth details their usage in Chapter 12.

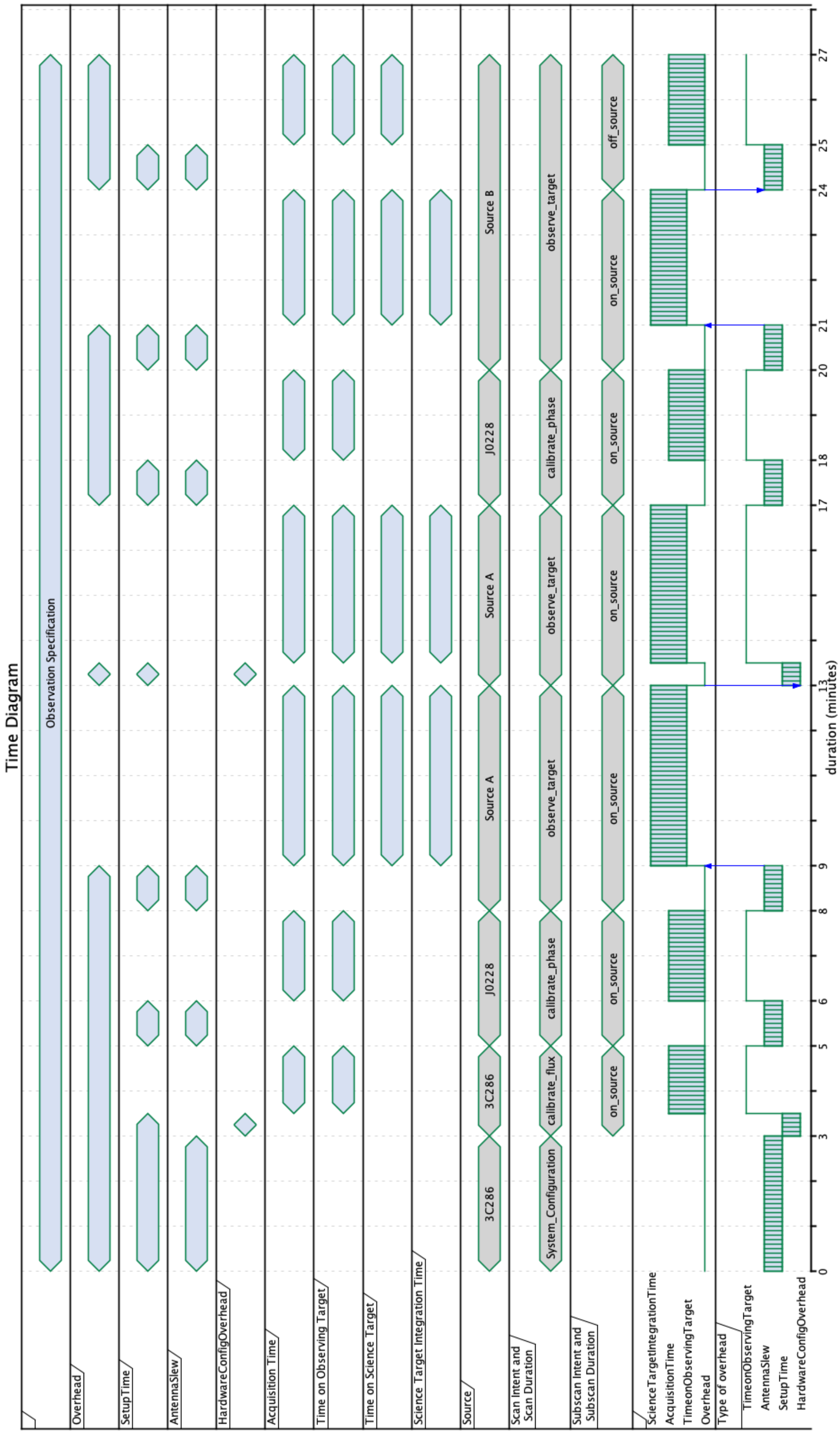


Figure 10.1: Hierarchy and concepts of time in the *Observation Planner*.

10.1 Use Cases of Concepts and Definitions in the Observation Planner

10.1.1 (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 SPECS. 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 10.3 and 10.4 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 sub-scan is needed. Thus, the Acquisition Time is 280 seconds. There Sub-scan Intent is ON_SOURCE, and the Scan Intent is CALIBRATE_FLUX and CALIBRATE_BANDPASS.
  - The duration of a sub-scan 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 § 15.1.1 algorithm determines the Setup Time for this sub-scan is 20 seconds.
  - The duration of the Scan is the sum of all the sub-scan times.
- 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 μJy bm^{-1}. The VLA Exposure Time Calculator returns $\sim$ 5 minutes as being sufficient to reach this sensitivity (Figure 10.2). Therefore, the Requested Time is 5 minutes per Science Target.

---

[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 Specs.

[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

- The § 15.1.1 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 sub-scan 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 14.1). Given the Scan Duration of the Phase Referencing calibration, there are 400 seconds (600 seconds - (2 × 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 Sub-scans/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 sub-scans.
    - 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 Sub-scan 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 Sub-scans that share Scan Intent OBSERVE_TARGET, Sub-scan 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?)

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

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

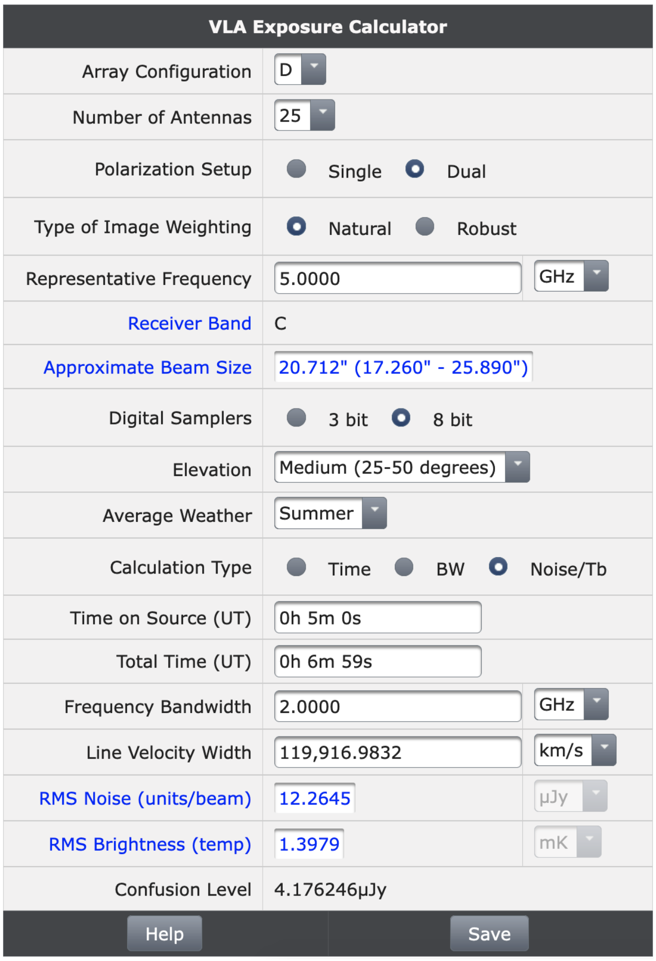
A. Science Target Integration Times

Q. How much time is spent observing the calibrators?

A. Time on Observing Targets - Science Target Integration Times

---

[4] The details of how the Requested Time is distributed amongst the sub-scans for Observing Targets are in § 12.1



VLA Exposure Calculator

Array Configuration	D	
Number of Antennas	25	
Polarization Setup	○ Single ● Dual	
Type of Image Weighting	● Natural ○ Robust	
Representative Frequency	5.0000	GHz
Receiver Band	C	
Approximate Beam Size	20.712" (17.260" - 25.890")	
Digital Samplers	○ 3 bit ● 8 bit	
Elevation	Medium (25-50 degrees)	
Average Weather	Summer	
Calculation Type	○ Time ○ BW ● Noise/Tb	
Time on Source (UT)	0h 5m 0s	
Total Time (UT)	0h 6m 59s	
Frequency Bandwidth	2.0000	GHz
Line Velocity Width	119,916.9832	km/s
RMS Noise (units/beam)	12.2645	µJy
RMS Brightness (temp)	1.3979	mK
Confusion Level	4.176246µJy	

Figure 10.2: Example of Use Case with Time Concepts

Table 10.3: VLA Example Observation Specification Part 1: Scan List

Scan Intent	Observing Target	Setup Time (s)	Acquisition Time (s)	Sub-scan 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 10.4: 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

10.1.2 A Story of 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 10.5 shows the Scan List and summary of the Observation Specification for this example.

Contents of Scan List prior to Phase 3

Table 10.5: GBT Example Observation Specification Part 1: Scan List

Scan Intent	Observing Target	Setup Time (s)	Acquisition Time (s)	Sub-scan 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 10.6: 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

Chapter 11

Observation Planner Phase 1

A work in progress - i.e., not ready for review.

Phase 1 applies two partitioning algorithms, Initial Partitioning and Final Partitioning, to the *Science Target List* to form clusters of Science Targets that can then be scheduled. 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.

An *Observing Strategy* Strategy Selector (e.g., § 3.4, 4.4) selects the appropriate Partition Plan, which is in Calibration Strategy (Chapter 5). The Partition Plan specifies how the partitioning is performed. The implementation details of the Partition Plan and its Partition Instructions are described in this chapter.

11.1 Initial Partitioning

Partition Instructions (PIs) contain the instructions for how to group Science Targets. There are two types of PIs: Hierarchical Partition Instructions and Time Partition Instructions. There are multiple types of Hierarchical PIs, which capture the various criteria needed to partition the *Science Target List.* The Hierarchical PIs are completed in the Partition Plan before the Time Partition Instruction is implemented. Table 11.1 summarizes the types of PIs and the sections below provide the details.

Table 11.1: Summary of Partition Instructions

Name	Criteria	Section
Hierarchical PI		11.2
Priority PI	*Scheduling priority*	11.2.1
VLA Configuration PI	VLA Configuration	11.2.2
Frequency PI	Frequency	11.2.3
Distance PI	Angular Separation on the Sky	11.2.4
Setup Time PI	Setup Time	11.2.5
Time PI	Local Sidereal Time and Requested Time	11.3

11.2 Hierarchical Partition Instructions

Hierarchical Partition Instructions utilize Scipy's Hierarchy[1], Linkage[2], and fcluster[3] to perform hierarchical clustering. A Hierarchical Partition Instruction will

❖ Construct a linkage matrix using scipy.cluster.hierarchy.linkage, which takes as inputs

---

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

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

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

  - a set of Science Targets,
  - a custom metric that defines a distance function,
  - a method that specifies the linkage algorithm to use in calculating the distance.
- ❖ 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 of distance here is not necessarily the angular distance between astronomical objects; it is the dissimilarity between two objects for a given metric. The linkage algorithm methods defined in the documentation for the linkage are single, complete, average, weighted, centroid, and ward.

From the criterion defined in the documentation for fcluster, the Hierarchical Partition Instructions 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]"[3].

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 11.1, the tree shows the hierarchical relationship between Nodes 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.

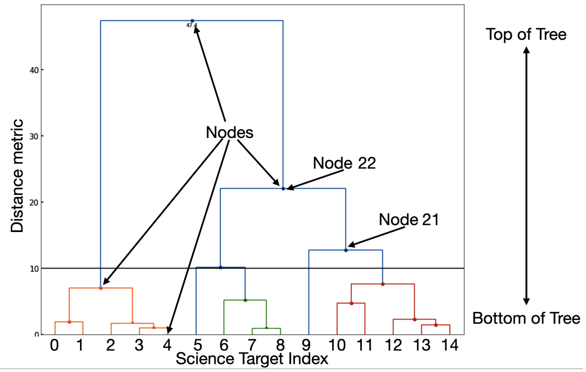


Figure 11.1: Example Dendrogram tree for 15 Science Targets using angular distance on the sky as the custom metric. The solid black line at y = 10 cuts the tree, yielding 5 clusters.

11.2.1 Priority Partition Instruction

One *scheduling priority* is permitted per *Observation Specification.* If a *scheduling priority* has been set for the Science Targets, they should be partitioned into different clusters based on their *scheduling priority.* Generally, this Partition Instruction would not be implemented until after the *Allocation Disposition* and a Proposal has a non unique *scheduling priority*, which could occur if multiple *Allocation Requests* exist in the Proposal.

- ❖ linkage
  - The custom metric compares the *scheduling priority* of any two Science Targets.
    - ∗ If equal, the metric returns 0.
    - ∗ If different, the metric returns 1.
  - Linkage algorithm method is "complete".
- ❖ fcluster
  - The threshold is set to 0.5.

11.2.2 VLA Configuration Partition Instruction

One VLA Configuration is permitted per *Observation Specification*; *Capability Requests* comprised of multiple VLA Configurations are partitioned into different clusters.

- linkage
    - The custom metric compares the VLA Configuration of any two Science Targets.
        - If equal, the metric returns 0.
        - If different, the metric returns 1.
    - Linkage algorithm method is “complete”.
- fcluster
    - The threshold is set to 0.5.

11.2.3 Frequency Partition Instruction

The Frequency Partition Instruction is a frequency based partitioning of the *Science Target List*. Commonly, DSS sessions and Scheduling Blocks contain observations spanning multiple receivers. For example, high frequency observations are rarely scheduled with middling frequency ones because the weather constraints are considerable in the former. Similarly, very low frequency observations have their own considerations. Another example is the overhead associated with higher frequency observations. The VLA at Q-band has a much shorter Phase Referencing cycle than C-band, so these two frequencies should not be in the same cluster, as the overhead of the C-band observations would be considerable by comparison. The Frequency Partition Instruction is instantiated with the information to appropriately partition the *Science Target List*.

- linkage
    - The custom metric compares the frequency bands of any two Science Targets. The metric returns the dissimilarity value specified in the template.
    - Linkage algorithm method is “complete”.
- fcluster
    - The threshold is set to 15.

11.2.4 Distance Partition Instructions

The Distance Partition Instructions partitions the *Science Target List* based on the angular separation between Sources.

- linkage
    - The custom metric uses the Vincenty Formula[4] to calculate the angular separation between any two Sources.
    - Linkage algorithm method is “complete”.
- fcluster
    - The threshold is set by the Strategy. The default value is 10 (degrees).

---

[4] `https://en.wikipedia.org/wiki/Great-circle_distance`

11.2.5 Setup Time Partition Instructions

The Setup Time Partition Instructions account for the Antenna slew time and the Hardware Configuration overhead (see Chapter 10 for definitions).

- linkage
  - The custom metric returns a time in seconds that is the sum of the (1) simulated motion of antennas (Section 15.1.4)and (2) the Hardware Configuration overhead for a list of Science Targets.
  - Linkage algorithm method is "complete".
- fcluster
  - The threshold is set by the Strategy. The default value is 540 (seconds).

11.3 Time Partition Instructions

A Time Partition Instruction takes the output of a Hierarchical Partition Instruction to evaluate, and if needed, further assemble clusters, as the Hierarchical Partition Instructions do not consider the time requirements of the Science Targets. A Time Partition Instruction is responsible for delivering temporally suitable clusters that can be scheduled, all the while accounting for frequency, and facility, and science dependent phenomenon. This algorithm needs be as efficient as possible, follow a *Facility*'s best practices, and be deterministic. The constraints on this problem include the following:

- the total Requested Time of a cluster
  - There are great number of sources are not above the horizon all day[5]. The algorithm needs have the intelligence to partition the Requested Time of a Science Target such that clusters can be assembled. Then, the temporal extent of the cluster needs to be considered. It is easy to create too small of clusters that are then overhead dominated. The algorithm must account for the macro nature of a group of Science Targets and intelligently build the clusters around LST considerations.
- the maximum duration of an *Observation Specification*
  - For example, the nature of dynamic scheduling of the VLA favors shorter ($\lesssim$ 3 hr) Scheduling Blocks. Similarly, DSS sessions should be nominally $\lesssim$ 6 hr. The *scheduling priority* (the rank of A, B, etc) of a *Observation Specification* may also be a factor under consideration. Before a proposal is reviewed of course, the algorithm would treat every proposal as an A-rank, meaning it creates the best possible clusters for the science goals. However, the algorithm should be able to accommodate any differences that may exist in scheduling constraints that are dependent on the ranking of the proposal.
- the overhead associated with a cluster
  - Overhead dominated *Observation Specifications* are undesirable, so the algorithm must assess the clusters it has created to avoid building, for example, 2 good clusters and 1 severely overhead dominated cluster. It needs to be able to self evaluate to intelligently redistribute sources if necessary. To do this, it needs to have flexibility in understanding the constraints. For example, if the redistribution of the 1 severely overhead dominated cluster produces 2 clusters that are marginally in violation of the maximum duration of a Scheduling Block of 3 hrs, this could be acceptable as long as the cluster isn't in violation of the LST constraints. Ideally, the algorithm can self assess in a reasonable manner to arrive at the best solution.

---

[5] surprise!

An experienced observer simultaneously considers these constraints *and* their science goals as they arrange their sources for observation. To automate this process, however, is no small feat. This section attempts to explain an algorithm that produces reasonably suitable clusters. An experienced observer can likely produce a more optimized arrangement of clusters by hand. As always, the user is welcome to ignore the suggested clustering in favor of their own implementation[6].

The groups of Science Targets that the Hierarchical Partition Instruction produces need to be evaluated to determine if they are optimal clusters[7]. The Time Partition Instruction algorithm has a knowledge base and a mix of if-then rules and procedural code to facilitate the evaluation or assembly of clusters. The knowledge base and behavior of the code is described below.

1. The algorithm defines a cluster as being composed of one or more Science Targets and as one that has been evaluated as a suitable collection (see item 5). 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 that has yet to be evaluated.
2. The algorithm keeps track of how many and which Science Targets are assigned to clusters. The state of the algorithm is the accounting of this progress. The algorithm assembles Science Targets into groups, evaluates the collection, and assigns the group as a cluster if is passes the criteria or it discards the grouping for later consideration. When all Science Targets have been assigned to clusters, the state is complete and the Time Partition Instruction delivers the clusters to the Final Clustering portion of *Phase 1*.
3. The algorithm has *Capability* specific knowledge that constrains the assembly and evaluation of clusters, such as
   (a) a MAXIMUM DURATION, which is the maximum duration of an *Observation Specification*, excluding all overhead associated with calibration (e.g., Phase Referencing, Flux Density, Pointing, etc);
   (b) effective knowledge of the sky and how an antenna is able to observe.
4. The algorithm assesses the length of time a Science Target is able to be observed. This assessment is dependent on LST considerations (e.g., how long is the Source above or below a certain elevation) and the MAXIMUM DURATION. For each Science Target, this knowledge is maintained by two attributes associated with a Science Target: the PARTITIONED REQUESTED TIME and the REPEAT COUNT. The algorithm can dynamically update these attributes as it assembles a group. The attributes are interdependent and related to the Requested Time:

   $$\text{PARTITIONED REQUESTED TIME} \times \text{REPEAT COUNT} \geq \text{Requested Time}. \tag{11.3.1}$$

   The algorithm cannot specify a PARTITIONED REQUESTED TIME that is longer than a Source is available to be observed. As a consequence, it can never lower the REPEAT COUNT of a Science Target once it is specified.
   (a) Once a cluster is formed, the PARTITIONED REQUESTED TIME is given to *Phase 2* as an initial estimate for the amount of time an antenna will collect data on the specific Science Target in a single *Observation Specification.*
   (b) The REPEAT COUNT specifies how many times an element repeats; it is by definition a positive integer. REPEAT COUNTS are specified for Science Targets and inherited by clusters.
5. The algorithm assesses a group of Science Targets to determine if it is suitable for observation, which is insipidly refer to here as a "Good Cluster". A Good Cluster has the following attributes:
   (a) None of the Science Targets in the group belong to a different cluster.

---

[6] You'll just have to do by hand.

[7] Or, just not bad ones. Let's keep the bar reasonable.

   (b) All of the constituents of a cluster are required to have the same REPEAT COUNT.
   (c) For all Science Targets in a cluster, the PARTITIONED REQUESTED TIME of any one Science Target cannot be longer than the length of time that Source is available to be observed.
   (d) A cluster's total PARTITIONED REQUESTED TIME must be less than the LST span of the cluster. (there's a better way to say this...)
   (e) A cluster's total PARTITIONED REQUESTED TIME × REPEAT COUNT is equal to the total Requested Time of the cluster and less than or equal to 30 minutes + total Requested Time.
   (f) A cluster's total PARTITIONED REQUESTED TIME must be less than or equal to the MAXIMUM DURATION;
   (g) A cluster's total PARTITIONED REQUESTED TIME is greater than or equal to 95% of the MAXIMUM DURATION;
   (h) The total Setup Time associated with a cluster is less than or equal to 5% of the MAXIMUM DURATION.

6. The algorithm adheres to the constraints it is given (e.g., no cluster may have a duration that exceeds the MAXIMUM DURATION); however, it has 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 is allowed to supersede the conditions of 5f and 5g in favor of the following:
   (a) A cluster's total PARTITIONED REQUESTED TIME must be $\leq$ 1.2 × MAXIMUM DURATION;
   (b) A cluster's total PARTITIONED REQUESTED TIME is greater than or equal to 70% of the MAXIMUM DURATION. Note, the algorithm prioritizes 6a when assembling clusters.
7. As a final resort, the algorithm may decide the only course is to assign all or some of the Science Targets as their own cluster. If the algorithm assesses this state of itself, it will supersede condition 5g (and 6b) by setting effectively no minimum to how much PARTITIONED REQUESTED TIME is in a cluster. The algorithm always attempts to maximize the PARTITIONED REQUESTED TIME, so it will (should?) never allow an unreasonably small PARTITIONED REQUESTED TIME (and thus a large REPEAT COUNT). Validates of the cluster adds an additional level of protection against this possibility.

Note, the values in blue and *Capability* specific information should exist in a table in this document. The appropriate *Facility* subsystem scientist should validate the values.

11.3.1 Example of a Time Partition Instruction on a Simple *Science Target List*

A *Science Target List* has a single Science Target with a Requested Time of 12 hours. The Science Target is only above the horizon for 3 hours. The MAXIMUM DURATION is 2.2 hours. The algorithm makes a single cluster composed of the Science Target and compares the MAXIMUM DURATION, the time above the horizon, and a rough approximation of the overhead requirements (e.g., excess time needed for calibration). The MAXIMUM DURATION is stricter time constraint, so the the REPEAT COUNT is equal to 6, even though the Requested Time /MAXIMUM DURATION ~ 5.45, as the REPEAT COUNT must be an integer and Eq (11.3.1) must be true. The PARTITIONED REQUESTED TIME is then Requested Time/REPEAT COUNT = 2 hr.

The algorithm evaluates the cluster and then its state. The algorithm's state is that all Science Targets have been assigned to a cluster and the clusters are "Good Clusters". The Time Partition Instruction terminates.

11.4 Final Partitioning

The Final Partitioning takes the partitioned Science Targets and assesses the LST constraints and spatial considerations to find common *calibrators*.

Chapter 12

Observation Planner Phase 2

The Calibration Strategy contains set of prescriptive templates for how to partition and calibrate the *Science Target List*, which is facilitated by the creation of Observing Instructions. The *Capability* makes the selection of what Calibration Strategy to use; for example, the *Capability* would select the Calibration Strategy of GBT Spectral Line Observations for an Observing Type of GBT Spectral Line. The role of the Calibration Strategy in the system is to convert a *Science Target List* into one or more Calibration Plans where a Calibration Plan is expressed as a set of Observing Instructions which can then be scheduled. The following sections detail the Observing Instructions before describing the Calibration Plans.

12.1 Observing Instructions

Observing Instructions (OIs) 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 sub-scans,
- how often the observations are required to be observed.

Observing Instructions (OIs) 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, which include

- Science Observing Instructions, which are correlated with the *Science Target List*;
- 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 12.1 briefly summarizes the OIs, and Figure 12.1 provides a illustrative guide for the hierarchy.

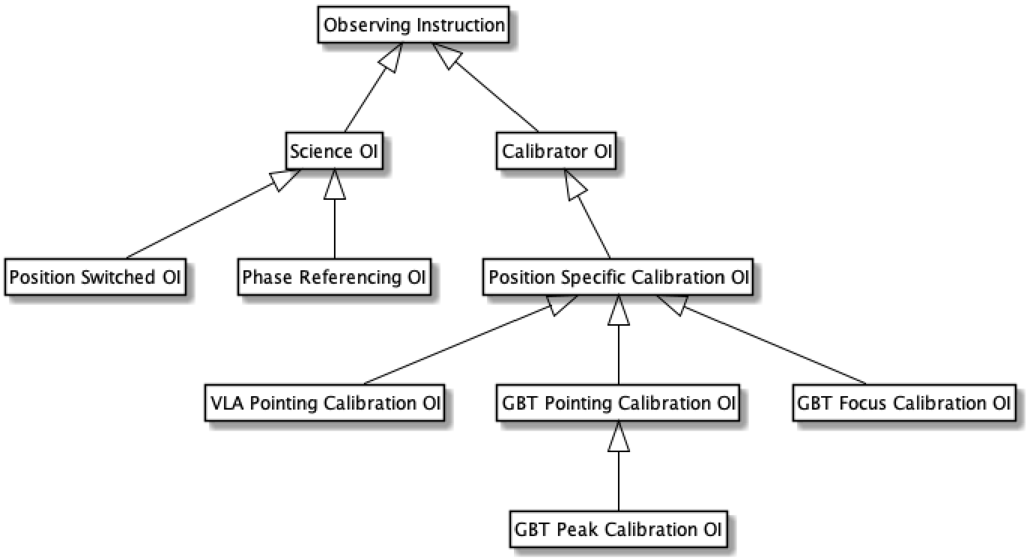


Figure 12.1: Hierarchy of the Observing Instructions.

Table 12.1: Summary of Observing Instructions

Name	Summary	Section
Science OI	Encapsulates the observation of a Science Target	12.1.1
Position Switching OI	A Science OI for moving between on-source and off-source positions the purposes of calibration.	12.1.1.1
Phase Referencing OI	A Science OI for implementing the phase referencing technique for interferometry.	12.1.1.2
Calibrator OI	Encapsulates the observation of a calibrator	12.1.2
Position Specific Cal OI	Instructions for observations pertaining to a distance on the sky over which an OI is valid.	12.1.2.1
VLA Pointing Cal OI	A Position Specific OI for VLA pointing calibrations.	12.1.2.1.1
GBT Pointing Cal OI	A Position Specific OI for GBT pointing calibrations.	12.1.2.1.2
GBT Focus Cal OI	A Position Specific OI for GBT focus calibrations	12.1.2.1.3

12.1.1 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 Sub-scan Prototypes to be realized within each Scan. Each Sub-scan Prototype will have an associated weight, which specifies how the Requested Time is distributed (as Acquisition Time) amongst the Sub-scans 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 sub-scans according to their specified weights.

12.1.1.1 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 Sub-scan Prototypes of equal weight. The first Sub-scan Prototype has the off-source position, a Hardware Configuration, and a Sub-scan Intent of OFF_SOURCE. The second has the on-source position, the Hardware Configuration, and the Sub-scan 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.

12.1.1.2 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.

12.1.2 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 Sub-scan Prototypes;
- a list of Scan Intents; note that each Scan has all of these intents;
- the Acquisition Time for each sub-scan;

- 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 Sub-scan Prototype in the input list should be scheduled with the specified Acquisition Time. Setup Times should be calculated based on the information in the Sub-scan Prototype.

12.1.2.1 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 Sub-scan 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.

12.1.2.1.1 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 Sub-scans 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 Sub-scan 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.

12.1.2.1.2 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

12.1.2.1.2.1 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

12.1.2.1.3 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

Chapter 13

Observation Planner Phase 3

13.1 Scheduling Instructions

13.1.1 Facility Scheduling Instruction

- sets minimum elevation for observations (freq dependent, polarization dependent)

13.1.2 Priority Scheduling Instruction

The schedule priority of a *Proposal* is not known at the same of its creation, as no *Allocation Disposition* as been distributed. The algorithms assume *a priori* that "A" priority will be awarded to generate *Observation Specifications.* Adjusting the schedule priority can have implication on the creation of the *Observation Specification.* This can be accommodated by the creation of a Priority Scheduling Instruction in the Partition Plan.

A Priority Scheduling Instruction will set

1. An absolute maximum duration of an *Observation Specification*
2. An absolute minimum duration of an *Observation Specification*
    - equal to the sum of the Requested Time of a cluster divided by (1) and expressed as a value between 0 and 1
3. An absolute maximum Setup Time associated with a cluster
    - equal to the sum of the Setup Time of a cluster divided by (1) and expressed as a value between 0 and 1
4. An absolute maximum extent on the sky for a cluster

Note, the proto-type has more flexibility than this but that detail is not necessary at this time.

13.1.2.1 Priority SI: VLA A-priority Scheduling Instruction

VLA A-priority SI is a type of Priority SI, and sets the parameters of the Priority SI as

1. 10800 seconds (3 hours)
2. 0.9
3. 0.1
4. 10 degrees

13.1.2.2 Priority SI: VLA C-priority Scheduling Instruction

VLA C-priority SI is a type of Priority SI, and sets the parameters of the Priority SI as

1. 10800 seconds (3 hours)
2. 0.9
3. 0.1
4. 10 degrees

Part V

Misc

Chapter 14

Specification Constraints

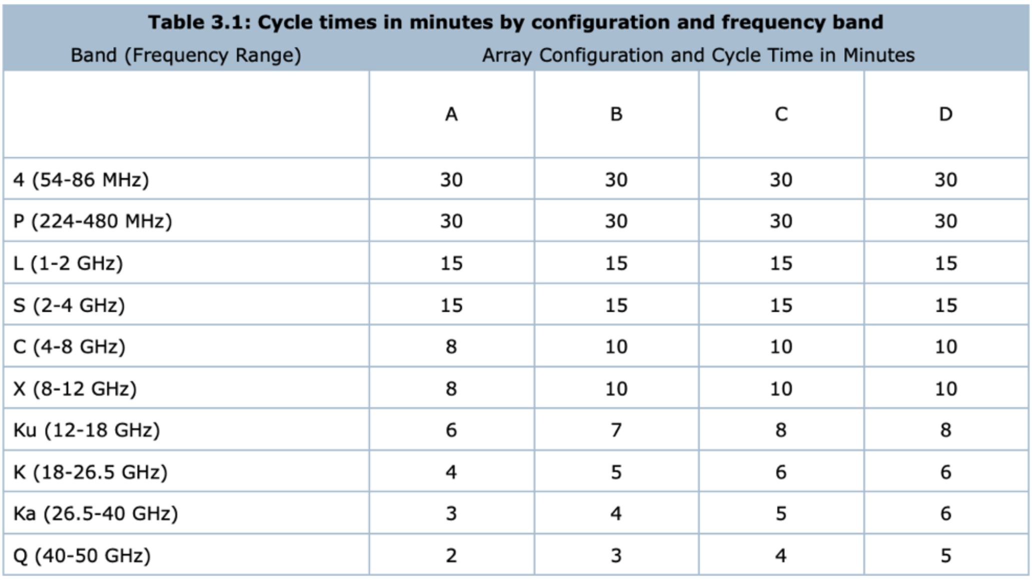


Table 3.1: Cycle times in minutes by configuration and frequency band

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 14.1: Cycle Times in Minutes by Configuration and Band

Table 14.1: Placeholder Table of Cycle Times

Band	Receiver	Maximum Cycle Time

Chapter 15

Misc

15.1 Calculations

15.1.1 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 configuration between 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 overheads

15.1.2 Sensitivity Calculators

15.1.2.1 VLA Sensitivity Calculator

15.1.3 Catchall: Needs to be written

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

15.1.4 Antenna Motion

Toy code exists but needs written up.

Chapter 16

Worked Examples

16.1 Initial Partitioning Example

Depending on the Partition Plan, iterations through Partition Instructions can further fragment clusters; this produces a set of lists of nodes that are inputs for the next iteration of partitioning.

Consider Figure 16.1 and the following examples of two Sources, A and B, and two Hardware Configurations, Hardware 1 and Hardware 2.

- If the *Science Target List* contains two Sources but each has two (or more) Hardware Configurations with less than an hour of Requested Time per Hardware Configuration, the Strategy selected by the *Capability* is a Distance Partition Instruction.

Ex 1. If the two Sources are not partitioned in distance, the result is one cluster with all Sources and all Hardware Configurations (i.e., all Science Targets).

Ex 2. If the Sources are partitioned by the Distance Partition Instruction, there are 2 clusters: one for each Source, where each Source has both Hardware Configurations.

- If the *Science Target List* contains two Sources, has two (or more) Hardware Configurations per Source, and more than an hour of Requested Time per Hardware Configuration, the Strategy includes a Hardware Configuration Partition Instruction and a Distance Partition Instruction.

Ex 3. If the two Sources are not partitioned in distance but the Hardware Configuration is partitioned into two clusters, the result of Initial Partitioning is two clusters: each one containing the Sources but different Hardware Configurations.

Ex 4. If the Sources are partitioned by the Distance and Hardware Configuration Partition Instructions, then there are four clusters: one for each Science Target.

Ex 5. If the Sources are partitioned by the Distance Partition Instruction but not the Hardware Configuration Partition Instruction, there are two clusters: each one containing the Sources but different Hardware Configurations.

Ex 6. If the Sources are not partitioned by the Distance or Hardware Configuration Partition Instructions, there is 1 cluster composed of all the Science Targets.

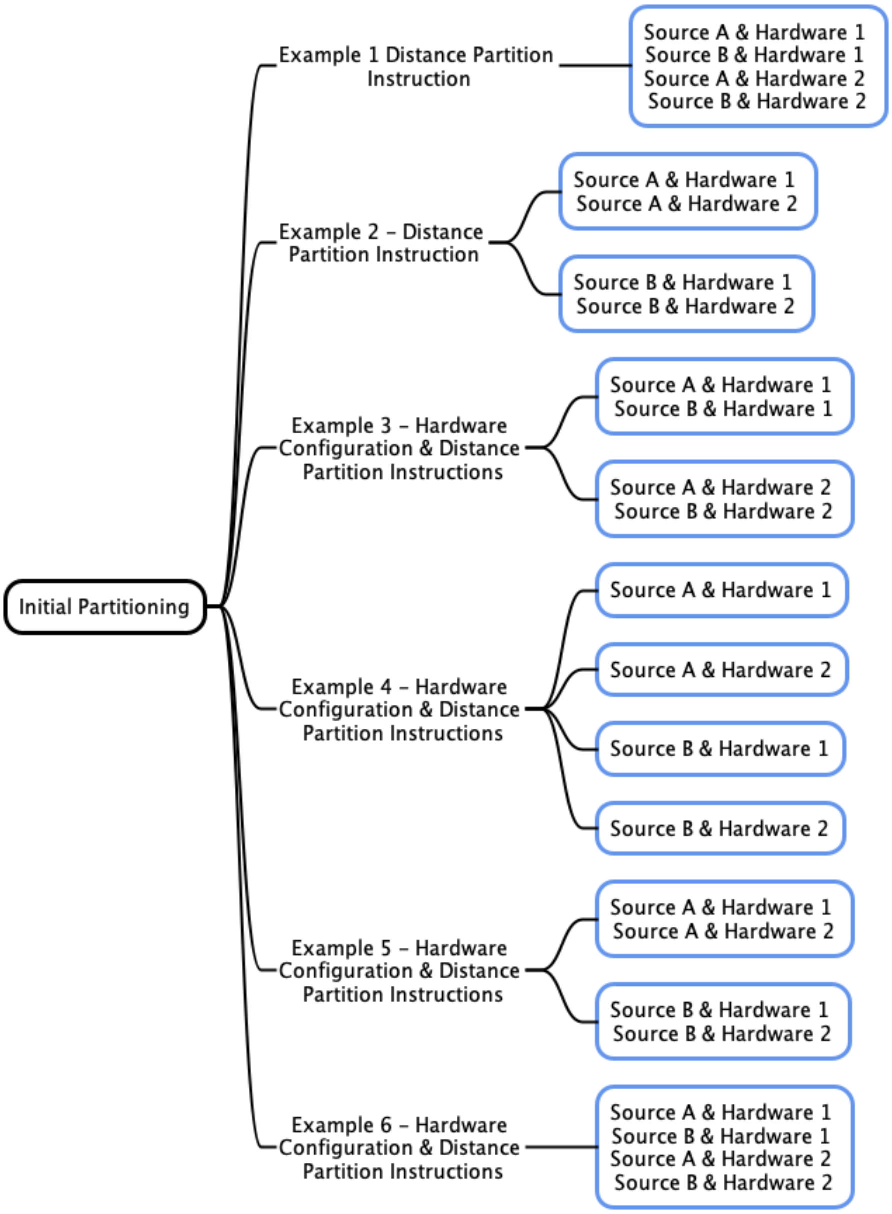


Figure 16.1: Example Initial Partitioning for two Sources and two Hardware Configurations.

Index