
EXPRES

Release 1.0

Louis, CK; Cecconi, B; Hess SLG

Jan 23, 2024

CONTENTS:

1	Installation Guide	3
1.1	Local installation	3
1.2	Required software	3
1.3	Precomputed data	4
1.4	Testing	4
2	Quickstart	5
2.1	Configuration File	5
2.2	Updating a Configuration File	5
2.3	Running ExPRES Online	7
3	Run-on-Demand	9
3.1	Guest Access	9
3.2	Authenticated Access	9
3.3	Command Line Interface	9
3.4	Guest Access	9
3.5	Authenticated Access	11
4	Advanced Users' Guide	13
4.1	Main Concepts	13
4.2	Simulation Setup	15
4.3	Plasma Density Models	24
4.4	Magnetic Field Models	27
5	References	29
6	Indices and tables	31
	Bibliography	33

ExPRES (Exoplanetary and Planetary Radio Emission Simulator) is a versatile tool that computes the observation opportunities of planetary radio emissions, based on the radio source beaming patterns and the observer's location.

ExPRES is an open source software (available under MIT license). It is maintained on [GitHub](#) and it is citable using its DOI:10.5281/zenodo.4292002. The code is described in details in a paper published in Astronomy & Astrophysics [LHC+19].

ExPRES is part of the [MASER](#) (Measuring Analyzing & Simulating Emissions in Radio frequencies) project. The code, its references and the access to data is presented [here](#).

INSTALLATION GUIDE

1.1 Local installation

EXPRES can be downloaded from: <https://github.com/maserlib/EXPRES>

The single step installation is to clone the repository:

```
git clone https://github.com/maserlib/EXPRES.git
```

Running the code requires to install other software, as described below.

1.2 Required software

1.2.1 IDL

The EXPRES code is running under **IDL (Interactive Data Language)**, has been developed and tested on IDL version 8.5, 8.6 and 8.7. This software must be installed for running the EXPRES code.

Note

The code has not been tested with **GDL (GNU Data Language)**, the open source version of IDL. This is part of future developments of the code.

The EXPRES uses several external IDL libraries, which are described in the following sections.

1.2.2 CDF and CDAWLib

The EXPRES code writes output files in **CDF (Common Data Format)**. The CDF library of IDL requires the installation of several packages:

- The CDF C-Library corresponding to your operating platform, see: <https://spdf.sci.gsfc.nasa.gov/pub/software/cdf/dist/latest-release/> Note that the code has been tested with the CDF C-Library version 3.6.4.
- The CDF IDL Library, available at: <https://spdf.sci.gsfc.nasa.gov/pub/software/cdf/dist/latest-release/idl/>
- The CDAWLib IDL routines, available at: <https://cdaweb.gsfc.nasa.gov/pub/software/cdawlib/>

1.2.3 IDL Astro

The ExPRES code also uses some routines from the `IDL Astro` library.

1.3 Precomputed data

Precomputed sets of data providing planetary magnetic field line models used by the ExPRES code must be downloaded before any operation. The data is available from the `ExPRES` section of the `MASER` data server.

More details on the additional files: https://gitlab.obspm.fr/maser/lesia-mag/lesia-mag_idl

1.4 Testing

The test suite are still under development. It uses `unittest` under Python 3.6 and the IDL-Python bridge (`idlpy`) provided with IDL version 8.7.

QUICKSTART

This section presents a concise user's guide to ExPRES.

2.1 Configuration File

An ExPRES run is configured through a configuration file (in *JSON* format). Examples of configuration files (and the associated results files) are available from the [ExPRES section of the MASER data repository](#) for Io-, Europa- and Ganymede-controlled emissions, and for various observers.

These *routine simulation files* are configured for using the JRM09 [[CKO+18](#)] magnetic field model (or ISaAC [[HBZG11](#)] for some of them), the CAN81 [[CAN81](#)] current sheet model, and an electron energy of 3 keV. The position of the *Active Flux Tube* (AFT) for Io is based on the (corrected) lead angle model of [[HCZ08](#)], whereas for Europa and Ganymede the AFT is the same as the flux tube connected to the moon. The file names are built automatically by ExPRES. The easiest option to build your configuration file is to update an existing one.

2.2 Updating a Configuration File

You can download a configuration file to get a template (please be sure to take one using the JRM09 magnetic field model [[CKO+18](#)], for up-to-date description). There are many items and options in this file. Here are the main ones.

2.2.1 Setting the temporal axis

The temporal axis is configured in the `TIME` section. The *start time* (`MIN` keyword) and *end time* (`MAX` keyword) should be provided in minutes relative to the simulation run time origin. The *time sampling step* is computed from the number of time steps (`NBR` keyword). The absolute time reference of the simulation run given in the `OBSERVER` section, with the `SCTIME` keyword. Hence, in most cases, `MIN` should be set to 0.

[More detailed temporal axis setting description](#)

2.2.2 Setting the spectral axis

The spectral axis is configured in the **FREQUENCY** section. The lower and upper bounds (**MIN** and **MAX** keywords) are given in MHz. The sampling interval is computed from the number of spectral step (**NBR** keyword). The spectral axis can use either a **Linear** or **Log** scale (**TYPE** keyword values). It is also possible to use a customised spectral axis.

[More detailed spectral axis setting description](#)

2.2.3 Setting the observer

The observer defines the place, from which the observation will be done. Basic users only need to use a limited set of parameters. In this short guide, we present the **Pre-Defined** type of observer (set in the **TYPE** keyword). If you need to change the *central body* (**PARENT** keyword), it is recommended to use a configuration file using the desired *central body*. The name of the observer (**SC** keyword) should then be a name known by ExPRES. The current list of known observers is: **Cassini**, **Juno**, **Earth**, **Voyager1**, **Voyager2**. The *time origin* of the simulation run is set with the **SCTIME** keyword, with the format: **YYYYMMDDhhmm**, with **YYYY** is the year, **MM** the month, **DD** the day, **hh** the hour, **mm** the minute, all 0-padded. The other parameters are not used in this case.

[More detailed observer's setting description](#)

2.2.4 Setting the output parameters

The **CDF** sub-section of **SPDYN** defines the parameters that will be provided in the resulting **CDF** file. Each parameters can be selected/deselected setting its value to **true/false**. In most cases, setting **Theta** keyword (opening angle of the emission cone in the direction of the observer) to **true** is the minimal recommended setup. Note that the more options are set, the bigger is the output file.

[More detailed output parameter's description](#)

2.2.5 Setting the plasma model parameters

The main set of parameters that can be adjusted is the plasma density model at the source. This is done through the **DENS** sub-section of **BODY**. The default model parameters, in case of the **Io**-controlled emissions, are an **Ionospheric** model (based on [HTK98]) and an **Io torus** model (based on [Bag94]).

[More detailed the plasma density model's description](#)

2.2.6 Setting the radio source parameters

The **SOURCE** section defines the radio source parameters. There may be several sources in the configuration file. The parameters are:

- **TYPE**: here, attached to a satellite, which means that the magnetic field lines used will be those connected to a moon.
- **SAT**: if **TYPE**="attached to a satellite", then provide the name of the moon (which also needs to be defined as a **BODY**)
- **aurora_alt**: sets the altitude (in Planetary radius) of the UV aurora (altitude below which electrons are lost by collision with the atmosphere)
- **NORTH**: emission will be produced in the northern hemisphere
- **SOUTH**: emission will be produced in the southern hemisphere

- **Width:** width of the beaming hollow cone (in degrees)
- **current:**
 - In most cases it should be set to `Transient (Alfvenic)`, which calculates self-consistently the beaming angle using the Cyclotron maser Instability (CMI) and a loss cone distribution function
 - It can also be set to `Constant`, so that the beaming angle will not be calculated using the CMI, but will be set at a chosen values (see next parameters)
- **Constant:** if `Current="Constant"` then provide here the value in degree (80.0 for example)
- **Accel:** the energy of the resonant electrons (in keV)
- **Refraction:** to take into account refraction in the source's vicinity

2.3 Running ExPRES Online

The code is available for Run-on-Demand at Observatoire de Paris: <https://voparis-uws-maser.obspm.fr/client/>

Short workflow to use this interface:

- Click on *Job List* (top left)
- In *Job List* for, select *ExPRES*
- Click on + *Create New Job* (top right)
- In *config* choose the configuration file (*.json*) you want to run. The other parameters (*runId*, *slurp_mem* and *Add control parameters* have to be left as there are)
- Click on *Submit*, and wait for a response. It will first marked as *Queued* and then as *Executing*. It will last a few tens of second to a few minutes (depends on how many time/frequency steps and how many cdf-output parameters you asked for).
- Then it will be marked as *Completed*,
- In > *Job Results* you will be able to download resulting files.
- If the Job is marked as *Error*, something went wrong during the simulation. Then, look at the > *Job Details*, and check the *stdout* and *stderr* sections.

For more details see the [Run-on-Demand](#) page.

RUN-ON-DEMAND

The ExPRES code is available on <https://voparis-uws-maser.obspm.fr/client> for run-on-demand requests. This server is implementing the UWS (Universal Worker Service Pattern), using the OPUS (Observatoire de Paris UWS System) framework. The service can thus be used from the web interface, or through UWS command line clients, such as the `uws-client` (python 2) or one of its forks available at `uws-client` implementing Python 3 support.

3.1 Guest Access

This type of access doesn't require to log in the server (no account). This allows guest users to run the code openly. There are some limitation to the usage:

- Run data and results are visible to all
- Run duration is limited to 10 minutes
- Job can only use the *master* branch

3.2 Authenticated Access

The authenticated access has to be requested to the [MASER team](#). This type of access has the following features:

- Run data and results are only accessible to the user
- The maximum run duration is 3 hours.
- Any of the ExPRES git repository branches can be selected

3.3 Command Line Interface

3.4 Guest Access

The code can also be launch from a Command Line Interface, using the `uws client` (more info and examples [here](#)).

You first need to download the `uws client`:

```
git clone https://github.com/aicardi-obspm/uws-client
cd uws-client
git checkout python3-support
python setup.py install
```

The following script gives a python example of how to run a simulation via the uws client, with a guest access:

Example

```
# coding : utf8
from uws import UWS
import time

def uws_call_expres(FILE="example.json", FILE_EPHEM=None, Job_List=None, runID=None,
                    LOOP=False, branch=None, LOGIN=None, executionDuration=None):
    FILE = "@"+FILE
    SERVER = "voparis-uws-maser.obspm.fr"
    if runID == None:
        runID = "test_job"
    if LOGIN == None:
        LOGIN = ""
        TOKEN = ""
        Job_List = "ExPRES"
        branch = "master"
    if branch == None:
        branch = "master"
        Job_List = "ExPRES"
    if branch == "develop"
        Job_List = "ExPRES-dev"

    parameters = {'config':FILE, 'runId':runID, 'branch':branch}
    if FILE_EPHEM != None:
        FILE_EPHEM = "@"+FILE_EPHEM
        parameters['ephemeride'] = FILE_EPHEM
    if executionDuration != None:
        parameters['executionDuration'] = executionDuration

    print(parameters)

    uws_client = UWS.client.Client(url=f"https://{SERVER}/rest/{Job_List}", user=LOGIN,
                                   password=TOKEN)

    job = uws_client.new_job(parameters)
    job = uws_client.run_job(job.job_id)

    print(uws_client.get_phase(job.job_id))
    print(f"Job : {job.job_id}")

    if LOOP:
        while True:
            time.sleep(2)
            phase = uws_client.get_phase(job.job_id)
            if phase == UWS.models.JobPhases.COMPLETED:
                print("Job completed")
                break
            elif phase == UWS.models.JobPhases.ERROR or phase == UWS.models.JobPhases.
↳ABORTED:
                print("Job failed")
                break
```

(continues on next page)

(continued from previous page)

```
job = uws_client.get_job(job.job_id)
for result in job.results:
    filename = "." + result.id
    print(f"Downloading {result.id}")
    uws_client.connection.download_file(str(result.reference), LOGIN, TOKEN, ↵
↵filename)
```

Then to run the simulation and retrieve the results:

```
from uws_call_expres import uws_call_expres
uws_call_expres(FILE="example.json", LOOP=True)
```

3.5 Authenticated Access

The above script is also valid for people with authenticated access. At this point, you must replace LOGIN=None and TOKEN=None with your login credentials (in text format). You will be able to access any of the ExPRES git repository branches by replacing branch=None by the desired branch (e.g. branch="master" or branch="develop").

ADVANCED USERS' GUIDE

ExpRES (Exoplanetary and Planetary Radio Emission Simulator) is a versatile tool that is fully configurable through the simulation run input file. We present here the details of each configuration parameter.

4.1 Main Concepts

The ExpRES tool is modeling planetary radio emission observability. It is described in details in [LHC+19]. It is implementing the Cyclotron Maser Instability (CMI) radio emission mechanism [Wu85], which predicts a strongly anisotropic radio source beaming pattern. The beaming pattern is a hollow cone, whose axis is aligned with the local magnetic field direction, and the cone opening angle is related to the unstable particle distribution function properties. ExpRES computes the *radio source* to *observer* spatial vector and compares its direction to the modelled radio source beaming pattern.

The ExpRES code configuration requires the definition of:

- *The celestial bodies involved in the simulation.* At least one *central body* must be defined, which serves as the *spatial origin* for the simulation. When several bodies are defined, their relative location with the *central body* must be available either as precomputed data, or through orbital parameters provided in the configuration file.
- *The location of the observer with respect to the central body.* The location data must be available (either as precomputed data, or through parameters provided in the configuration file).
- *The magnetic field and plasma density models associated to the celestial bodies.* Several type of models can be configured. ExpRES is using a set of pre-computed magnetic field lines from a series of magnetic field models. The plasma density models are set through configuration parameters.
- *The spatial distribution of the radio sources.* This location is related to the magnetic field line carrying the unstable particles. The range of radio source frequencies must also be set.
- *The radio source properties.* The radio emission mechanism is defined by a set of parameters characterising the radio source beaming pattern.

The *central body* is the simulation run spatial origin and its radius sets the units of spatial parameters. The times are given in a UTC scale in the observer's frame. The time origin of the simulation run is provided in the observer's definition.

All spatial parameters of the simulation configuration (distances, radii, lengths...) must be defined in the same units as that of provided *central body* radius. Hence, setting the central body radius to 1 implies that all other spatial parameters are provided in units of the central body planetary radii. On the contrary, providing the radius of the central body in km implies that all other spatial parameters must be also provided in km. The recommended convention is to provide all spatial parameters in units of the *central body* radius. This convention is followed in the examples provided below.

The file output file names are built by ExpRES, using a set up configuration parameters. The general scheme is: `expres_{OBS}_{BODY}_{SRC}_{MAG}_{SRC_PROP}_{DATE}_v{VERS}.json`. The parts of the tem-

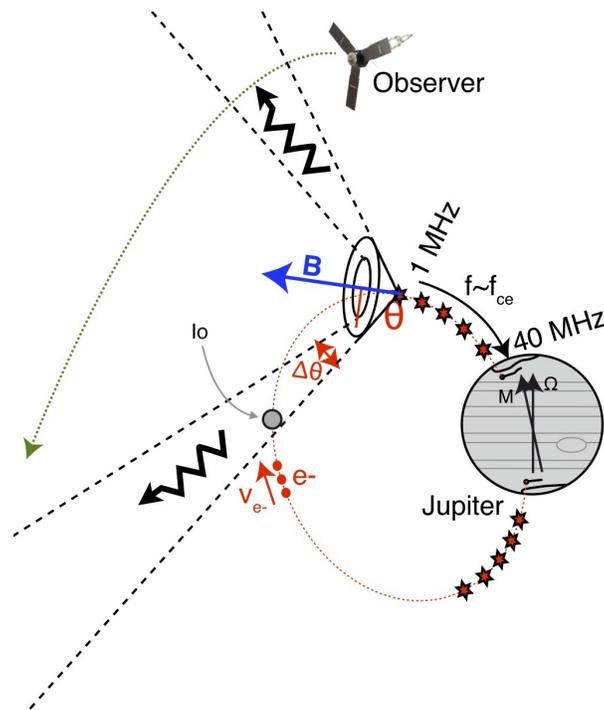


Fig. 1: **Fig. 1:** Schematic outline of the ExPRES code. The central body here is set up Jupiter and Juno is the observer. The radio sources are placed along the magnetic field line connected to Io. The radio emission beaming pattern model (hollow cone opening angle and thickness) is computed from the magnetic and plasma models. The radio emission observable when the observer is located within the radio source beaming pattern. (Figure adapted from C. Louis, PhD Dissertation, 2018)

plate are explained in the table below, with an example using the file: `expres_juno_jupiter_io_jrm09_lossc-wid1deg_3kev_20180913_v01.json`

Template part	Description	Example
{OBS}	Observer name	juno
{BODY}	Central body name	jupiter
{SRC}	Radio source driver	io
{MAG}	Magnetic field model name	jrm09
{SRC_PROP}	Radio source properties	lossc-wid1deg_3kev
{DATE}	Simulation date	20180913
{VERS}	Version of ExPRES	01

4.2 Simulation Setup

The simulation setup is configured via an ExPRES configuration file (in *JSON* format), following the [ExPRES JSON-Schema v1.3](#).

4.2.1 Configuration File Description

The ExPRES configuration file should start with the reference to the validation schema to be used. The configuration sections and structure are summarised below:

```
{
  "$schema": "https://voparis-ns.obspm.fr/maser/expres/v1.2/schema#",
  "SIMU": {...},
  "NUMBER": {...},
  "TIME": {...},
  "FREQUENCY": {...},
  "OBSERVER": {...},
  "SPDYN": {...},
  "MOVIE2D": {...},
  "MOVIE3D": {...},
  "BODY": [{...}, {...}]
  "SOURCE": [{...}, {...}]
}
```

Each JSON entry shown here is described in the next sections. The *BODY* section is specific: it is a list of *BODY* elements, each of which containing a list of *DENS* elements.

Section	Mandatory in v1.2	Description
<i>SIMU</i>	no	Simulation run description
<i>NUMBER</i>	yes	Number of elements for lists
<i>TIME</i>	yes	Time axis configuration
<i>FREQUENCY</i>	yes	Spectral axis configuration
<i>OBSERVER</i>	yes	Observer's configuration
<i>SPDYN</i>	yes	Dynamic Spectra output configuration
<i>MOVIE2D</i>	yes	2D movie output configuration
<i>MOVIE3D</i>	yes	3D movie output configuration
<i>BODY</i>	yes	Celestial bodies configuration
<i>SOURCE</i>	yes	Radio Sources configuration

4.2.2 General Parameters

The general parameters cover the time and frequency domain covered by the simulation, allow to give it a name to set the number of objects that will be included in the model. It is composed of 4 sections: SIMU, NUMBER, TIME, FREQUENCY.

Simulation Run Description

The SIMU section contains the simulation run description. It is composed of 2 keywords:

- NAME: The name of the simulation
- OUT: Output directory location (full path). If this path is empty, the current execution location is used. If this path points a file, the parent directory is selected.

Example: The simulation name is set to *Io2015-04-30*, and the output directory is defined from the path of the ExPRES configuration file.

```
"SIMU": {
  "NAME": "Io2015-04-30",
  "OUT": "/Groups/SERPE/SERPE_6.1/Corentin/save/Earth/VIPAL/2015/3kev/IO/IO2015-04-30.
↪json"
},
```

Simulation List Sizes

The NUMBER section defines maximum numbers of BODY, DENSITY and SOURCE objects, which can be configured in the simulation run. It is composed of 3 keywords:

- BODY: The number of planetary bodies in the simulation (e.g., 2 for Jupiter and Io)
- DENSITY: The number of plasma density model in the simulation (usually 1 per body)
- SOURCE: The number of radio source types in the simulation (usually 1 per interaction and per hemisphere)

Example: We want to define two bodies (Jupiter and Io), two density models (one for Jupiter's ionosphere, and the other for the Io Torus) and two sets of radio sources (one for each hemisphere).

```
"NUMBER": {
  "BODY": 2,
  "DENSITY": 2,
  "SOURCE": 2
},
```

Temporal Axis

The TIME section contains the simulation time configuration. Times are given in minute from the simulation time origin. The time origin is either set by the input ephemeris data or by the input orbital parameters. It is composed of 3 keywords:

- MIN: The start time of the simulation (in minutes), usually set to 0.
- MAX: The end time of the simulation (in minutes).
- NBR: The number of time steps of the simulation.

Example: The simulation starts at the simulation time origin, with 1440 minutes duration (one day), with one step per minute.

```
"TIME": {
  "MIN": 0,
  "MAX": 1439,
  "NBR": 1440
}
```

Spectral Axis

The FREQUENCY section contains the simulation spectral configuration. Frequency values are always in MHz units.

The spectral axis can be defined in several ways. The more generic way is to set the spectral axis bounds, the number of steps and the linear and logarithmic scale (see example below). It is also possible to use a predefined set of frequencies, corresponding to an existing instrument. Finally an external file containing a list of frequencies can be provided.

This section is composed of 5 keywords:

- TYPE: The spectral axis type. The allowed values are Linear, Log and Pre-Defined.
- MIN: The spectral axis lower bound in MHz. Not used when TYPE="Pre-Defined"
- MAX: The spectral axis upper bound in MHz. Not used when TYPE="Pre-Defined"
- NBR: The number of steps of the spectral axis. Not used when TYPE="Pre-Defined"
- SC: In case TYPE="Pre-Defined", the name of the specific spacecraft (not implemented, allowed values TBD), or a list of frequency values.

Example: The simulation spectral axis is a linear scale, ranging from 10 kHz to 44 MHz, with 781 steps.

```
"FREQUENCY": {
  "TYPE": "Linear",
  "MIN": 0.01,
  "MAX": 44.0,
  "NBR": 781,
  "SC": ""
},
```

Example: The simulation spectral axis is set of predefined frequencies.

```
"FREQUENCY": {
  "TYPE": "Pre-Defined",
  "MIN": 0,
  "MAX": 0,
  "NBR": 0,
  "SC": [0.1, 0.2, 0.3, 0.4, 0.5, 0.6, 0.7, 0.8, 0.9, 1, 1.1, 1.2, 1.3, 1.4, 1.5, 1.6, 1.
↵7, 1.8, 1.9, 2,
  2.1, 2.2, 2.3, 2.4, 2.5, 2.6, 2.7, 2.8, 2.9, 3, 3.1, 3.2, 3.3, 3.4, 3.5, 3.6, 3.7, 3.
↵8, 3.9, 4, 4.1,
  4.2, 4.3, 4.4, 4.5, 4.6, 4.7, 4.8, 4.9, 5, 6, 7, 8, 9, 10, 11, 12]
},
```

4.2.3 Observer Definition

The OBSERVER section contains the observer's configuration. There are three types of observers, configured by the TYPE keyword:

- Fixed observers, whose position does not vary in the reference frame of the simulation;
- Orbiter, which moves in the reference frame of the simulation, orbiting around a celestial body;
- Pre-Defined observers, which concerns known space mission around celestial bodies.

The observer's location is provided with respect to the simulation *central body*, defined in the BODY section.

This section is composed of a series of keywords. The table below provides which keyword shall be used, or left empty, or with a specific value. The following subsections give details for each observer's type.

Keyword	Observer's type		
TYPE	Fixed	Orbiter	Pre-Defined
EPHEM	<i>empty</i>	<i>empty</i>	<i>file name or empty</i>
FIXE_DIST	<i>distance</i>	auto	auto
FIXE_SUBL	<i>longitude</i>	auto	auto
FIXE_DECL	<i>latitude</i>	auto	auto
PARENT	<i>Reference body name</i>		
SC	<i>Observer's name</i>		
SCTIME	<i>Start time</i>		
SEMI_MAJ	0	<i>Semi major axis</i>	0
SEMI_MIN	0	<i>Semi minor axis</i>	0
SUBL	0	<i>Apoapsis longitude</i>	0
DECL	0	<i>Apoapsis latitude</i>	0
PHASE	0	<i>Phase from apoapsis</i>	0
INCL	0	<i>Inclination</i>	0

The observer's name (SC keyword) must be set, and can't be empty. When TYPE="Pre-Defined" and EPHEM="", the current allowed list of values is: Juno, Earth, Galileo, JUICE, Cassini, Voyager1, Voyager2.

The PARENT keyword must be set to one of the celestial body names defined in the BODY section. Except for specific cases, it is usually the *central body* name.

The simulation start time (SCTIME keyword) is provided in SCET (spacecraft event time), with a YYYYMMDDHHMMSS format.

Fixed Observer

A fixed observer is configured by a location at the start of the simulation: its distance (FIXE_DIST keyword) to the *central body*, its sub-longitude in degrees (FIXE_SUBL keyword) and its declination in degrees (FIXE_DECL keyword) in the reference body frame. The location of such an the observer is fixed in an absolute frame centered on the *central body*. Hence it is not fixed in the *central body* frame, which is rotating with its sidereal period.

Orbiter

The observer's orbital parameters are its semi-major (SEMI_MAJ keyword) and semi-minor (SEMI_MIN keyword) axis lengths, its apoapsis sub-longitude (SUBL keyword) and declination (DECL keyword), as well as the inclination of the orbit plane around the semi-major axis (INCL keyword). All angles are provided in the *central body* reference frame, and at the simulation time origin. Finally, the orbiter position requires the definition of its initial phase (PHASE keyword) on the orbit, i.e., 0 degree is at the apoapsis position.

Pre-Defined

In the case of predefined observers, the code is expecting to have access to ephemeris information. For a set of space missions (Cassini, Voyager1, Voyager2, Juno) or planetary bodies (Earth), the code will call the *Miriade* ephemeris webservice at IMCCE. For all other cases, an ephemeris file extracted from WebGeoCalc shall be provided using the EPHEM keyword.

Example: We configure a simulation with an observer at Earth, with a simulation starting on 2015-04-30T00:00:00.

```
"OBSERVER": {
  "TYPE": "Pre-defined",
  "EPHEM": "",
  "FIXE_DIST": "auto",
  "FIXE_SUBL": "auto",
  "FIXE_DECL": "auto",
  "PARENT": "Jupiter",
  "SC": "Earth",
  "SCTIME": "201504300000",
  "SEMI_MAJ": 0,
  "SEMI_MIN": 0,
  "SUBL": 0,
  "DECL": 0,
  "PHASE": 0,
  "INCL": 0
},
```

Example: We configure a simulation from the JUICE spacecraft, providing a WebGeocalc output CSV file.

```
"OBSERVER": {
  "TYPE": "Pre-Defined",
  "EPHEM": "WGC_StateVector_JUICE_SC_20320111T175800_20320111T185900.csv",
  "FIXE_DIST": "auto",
  "FIXE_SUBL": "auto",
  "FIXE_DECL": "auto",
  "PARENT": "Jupiter",
  "SC": "JUICE",
  "SCTIME": "",
  "SEMI_MAJ": 0,
  "SEMI_MIN": 0,
  "SUBL": 0,
  "DECL": 0,
  "PHASE": 0,
  "INCL": 0
},
```

4.2.4 Celestial Bodies Definition

The BODY section contains the celestial bodies configuration.

Two types of celestial bodies can be included in the simulations:

- Fixed bodies (at least is one needed): the simulation run reference body (MOTION=false);
- Orbiting bodies, which can orbit both fixed and orbiting bodies (MOTION=true).

Each body must be given a unique name within the configuration file, since the name is used internally by ExPRES to refer to them. Each body radius must be specified. All distances and scales units must be consistent throughout a configuration file.

Celestial body definitions include the following keywords:

- ON: Flag to activate the current body (true or false)
- NAME: The name of the current body (must be unique in the configuration file)
- RADIUS: The radius of the current body (in consistent units throughout the configuration file)
- PERIOD: The sidereal rotation period of the current body (in minutes)
- FLAT: The polar flatening ratio of the current body.
- ORB_PER: The orbital period according to 3rd Kepler's law at 1 radius (in minutes)

Example: For Io, we have $M_{Io} = 8.931 \times 10^{22}$ kg, $a = 1821 \times 10^3$ m and $G = 6.674 \times 10^{-11}$ N.m².kg⁻², therefore

$$T = \sqrt{\frac{a^3 * 4 * \pi^2}{G * M_{Io}}} * \frac{1}{60} = 105.4 \text{ min}$$

- INIT_AX: The reference longitude (in degrees)
- MAG: The internal body magnetic field model (see the *Magnetic Field Model* section below)
- MOTION: Flag to indicate if the current body is moving in the simulation frame (must be false for the central body)
- PARENT: Named body, around which the current body is orbiting (must be one of the defined bodies, and must be empty for the central body)
- SEMI_MAJ: The semi-major axis orbital parameter of the current body (must be 0 for the central body)
- SEMI_MIN: The semi-minor axis orbital parameter of the current body (must be 0 for the central body)
- DECLINATION: The declination orbital parameter of the current body (must be 0 for the central body)
- APO_LONG: The apoapsis Longitude parameter of the current body (must be 0 for the central body)
- INCLINATION: The inclination orbital parameter of the current body (must be 0 for the central body)
- PHASE: The initial orbital phase (at simulation start time) of the current body (must be 0 for the central body)
- DENS: A list of configuration of the plasma density model(s) related to the current body (see the *DENS* section)

Example: Defining Jupiter with the latest JRM09 magnetic field model and the CAN81 current sheet model. The body radius is set to 1, so that all distance and scale parameters must be given in Jovian radii in the configuration file.

```
{
  "ON": true,
  "NAME": "Jupiter",
  "RADIUS": 1,
  "PERIOD": 595.5,
  "FLAT": 0.064935,
  "ORB_PER": 177.83,
```

(continues on next page)

(continued from previous page)

```

"INIT_AX": 0,
"MAG": "JRM09+Connerney CS",
"MOTION": false,
"PARENT": "",
"SEMI_MAJ": 0,
"SEMI_MIN": 0,
"DECLINATION": 0,
"APO_LONG": 0,
"INCLINATION": 0,
"PHASE": 0,
"DENS": [...]
}

```

Orbital Parameters

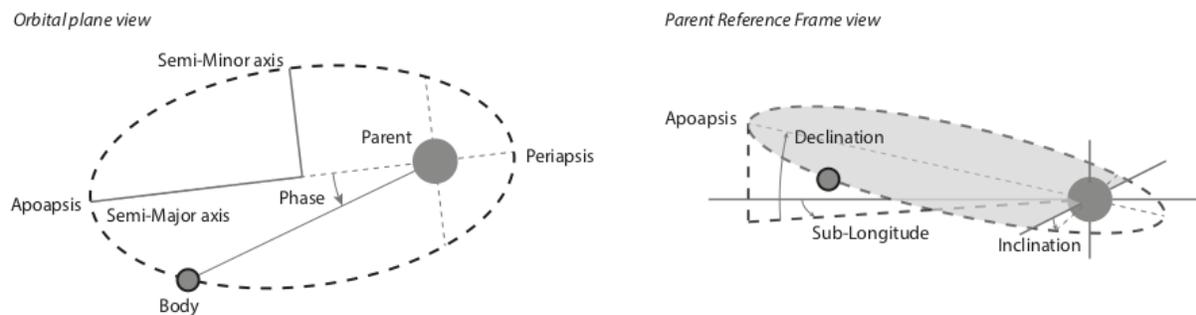


Fig. 2: **Fig. 2:** Sketch illustrating the orbital parameters of celestial bodies.

4.2.5 Radio Source Configuration

- ON: Flag to activate the current radio source (true or false)
- NAME: The name of the current radio source
- PARENT: The name of the parent body for this source (must correspond to a defined BODY name)
- TYPE: The type of radio source location. Four allowed values fixed in latitude, attached to a satellite, L-shell, M-shell.
- LG_MIN: The lower bound value of the source longitude (in degrees)
- LG_MAX: The upper bound value of the source longitude (in degrees)
- LG_NBR: The number of steps for the source longitude.
- LAG_MODEL: Model of the lead angle for the Io active flux tube; choices are: hess2011 [HBZG11], bonfond2009 [BGG+09], bonfond2017 [BGB+17], hinton2019 [HBB19], Hue2023 [HGL+23].
- LAT: If Fixed in latitude: Latitude in degree; else: apex distance in planetary radii.
- SUB: The subcorotation rate of the source (0 = no corotation)
- AURORA_ALT: The altitude of the aurora (in planetary radii)
- SAT: The name of the satellite when attached to a satellite is selected

- NORTH: Flag to activate the Northern hemisphere source (exclusive with SOUTH item)
- SOUTH: Flag to activate the Southern hemisphere source (exclusive with NORTH item)
- WIDTH: The thickness of the radio emission sheet (in degrees)
- CURRENT: The type of electron distribution in the source (see documentation). Allowed values: Transient (Alfvenic), Constant, Steady-State, Shell
- CONSTANT: The value of beaming pattern half-cone opening angle (if Constant is selected), in degrees
- MODE: The type of the wave mode. Allowed values: RX, LO (default is RX)
- ACCEL: The value of resonant electron beam energy in keV (not used when Constant is selected)
- TEMP: The value of the cold electron distribution temperature (in keV)
- TEMPH: The value of the halo electron distribution temperature (in keV)
- REFRACTION: Flag to activate refraction effects

Example: We configure a simulation with emission induced by Io ("TYPE": "attached to a satellite", "SAT": "Io"), in the northern ("NAME": "Source1", "NORTH": true) and the southern ("NAME"="Source2", "SOUTH": true) hemispheres. We use the lead angle model based on [hinton_JGR_20119] ("LAG_MODEL": "hinton2019") to determine the active magnetic field lines that will produce the emission. The electron have an energy of 3 keV ("ACCEL": 3) and the distribution function is of the loss cone type ("CURRENT": "Transient (Alfvenic)").

```
"SOURCE": [
  {
    "ON": true,
    "NAME": "Source1",
    "PARENT": "Jupiter",
    "TYPE": "attached to a satellite",
    "LG_MIN": 0,
    "LG_MAX": 0,
    "LG_NBR": 1,
    "LAT": 0,
    "LAG_MODEL": "hinton2019" ,
    "SUB": 0,
    "AURORA_ALT": 0.009091926738619804,
    "SAT": "Io",
    "NORTH": true,
    "SOUTH": false,
    "WIDTH": 1,
    "CURRENT": "Transient (Alfvenic)",
    "CONSTANT": 0.0,
    "MODE": "",
    "ACCEL": 3,
    "TEMP": 0,
    "TEMPH": 0,
    "REFRACTION": false
  },
  {
    "ON": true,
    "NAME": "Source2",
    "PARENT": "Jupiter",
```

(continues on next page)

(continued from previous page)

```

"TYPE": "attached to a satellite",
"LG_MIN": 0,
"LG_MAX": 0,
"LG_NBR": 1,
"LAG_MODEL": "hinton2019",
"LAT": 0,
"SUB": 0,
"AURORA_ALT": 0.009091926738619804,
"SAT": "Io",
"NORTH": false,
"SOUTH": true,
"WIDTH": 1,
"CURRENT": "Transient (Alfvenic)",
"CONSTANT": 0.0,
"MODE": "",
"ACCEL": 3,
"TEMP": 0,
"TEMPH": 0,
"REFRACTION": false
}
]

```

4.2.6 Output Configuration

Dynamic Spectrum Output

Dynamic Spectra output setup:

- INTENSITY: Flag to output 'Intensity' plots (true or false)
- POLAR: Flag to output 'Polar' plots (true or false)
- FREQ: Flags to setup output plot spectral axes
- LONG: Flags to setup output plot longitude axes
- LAT: Flags to setup output plot latitude axes
- DRANGE: Distance range for plot setup (number, min and max)
- LGRANGE: Longitude range for plot setup (number, min and max)
- LARANGE: Latitude range for plot setup (number, min and max)
- LTRANGE: Local-Time range for plot setup (number, min and max)
- KHZ: Flag for spectral axis output in kHz (true or false, default is MHz)
- LOG: Flag for spectral axis output in log scale (true or false)
- PDF: Flag for PDF file output (true or false)
- **CDF: Configuration of CDF file output**
 - THETA: Flag for THETA parameter output in the CDF file (true or false)
 - FP: Flag for FP parameter output in the CDF file (true or false)
 - FC: Flag for FC parameter output in the CDF file (true or false)

- "AZIMUTH: Flag for AZIMUTH parameter output in the CDF file (true or false)
- OBSLATITUDE: Flag for OBSLATITUDE parameter output in the CDF file (true or false)
- SRCLONGITUDE: Flag for SRCLONGITUDE parameter output in the CDF file (true or false)
- SRCFREQMAX: Flag for SRCFREQMAX parameter output in the CDF file (true or false)
- OBSDISTANCE: Flag for OBSDISTANCE parameter output in the CDF file (true or false)
- OBSLOCALTIME: Flag for OBSLOCALTIME parameter output in the CDF file (true or false)
- CML: Flag for CML parameter output in the CDF file (true or false)
- SRCPOS: Flag for SRCPOS parameter output in the CDF file (true or false)
- SRCVIS: Flag for SRCVIS parameter output in the CDF file (true or false)
- INFOS: IDL Saveset output (for debugging) (true or false)

2D Movie Output

- ON: Flag to activate Movie2D generation (true or false)
- SUBCYCLE: Subsampling rate of movie images (1=all temporal steps)
- RANGE: Size of Field of view (in central body planetary radii)

3D Movie Output

- ON Flag to activate Movie3D generation (true or false)
- SUBCYCLE: Subsampling rate of movie images (1=all temporal steps)
- XRANGE: Plotting Range in X axis (in central planet radius units)
- YRANGE: Plotting Range in Y axis (in central planet radius units)
- ZRANGE: Plotting Range in Z axis (in central planet radius units)
- OBS: Flag to activate plotting the location of the observer
- TRAJ: Flag to activate plotting the trajectories of the objects

4.3 Plasma Density Models

Various types of plasma density models can be used in ExPRES. They are configured by the DENS section in the BODY section (see the *Celestial Body* section above). Four types of density models are available:

- **Ionospheric:** exponential decrease with distance,
- **Stellar:** decreases with the distance squared,
- **Disk:** exponential decrease with altitude relative to equatorial plane and radial distance,
- **Torus:** exponential decrease from the center of a torus of given radius.

Plasma density model definitions include the following keywords:

- ON: Set to true to activate the density model or to false deactivate.
- NAME: The name of the density model (must be present, not empty and unique in the configuration file).

- TYPE: The type of the density model, with the allowed values: Ionospheric, Stellar, Disk, Torus.
- RH00: Definition depends on density model type (see below).
- SCALE: Definition depends on density model type (see below).
- PERP: Definition depends on density model type (see below).

4.3.1 Ionospheric Model

The Ionospheric density profile is modeled as:

$$\rho = \rho_0 \exp\left(-\frac{r - (r_{ref} + h_0)}{H}\right)$$

where:

Parameter	Definition	Unit	Keyword
ρ_0	Reference plasma number density	cm^{-3}	RH00
r	Radial distance	R_p	
r_{ref}	Reference radial distance on ellipsoid	R_p	
h_0	Peak density altitude above 1 bar level	R_p	PERP
H	Scale-height	R_p	SCALE

The r_{ref} is computed by ExPRES using the ellipsoid flattening parameter (FLAT keyword in BODY section) and the radio source latitude (computed from the SOURCE section).

Example: We define a Jovian ionospheric model, with a peak reference density of $3.5 \cdot 10^5 \text{ cm}^{-3}$ at an altitude of 650 km above the 1 bar level ($0.009092 R_p$) and a scale height of 1600 km ($0.0223801 R_p$), as defined in [HTK98].

```
{
  "ON": true,
  "NAME": "Body1_density1",
  "TYPE": "Ionospheric",
  "RH00": 350000.0,
  "SCALE": 0.0223801,
  "PERP": 0.009092
}
```

4.3.2 Stellar Model

The Stellar density profile is modeled as:

$$\rho = \rho_0 / r^2$$

where:

Parameter	Definition	Unit	Keyword
ρ_0	Reference plasma number density	cm^{-3}	RH00
r	Radial distance	R_p	

Note: Configuration keywords SCALE and PERP are not used for this model.

4.3.3 Disk Model

The Disk density profile is modeled as:

$$\rho = \rho_0 \exp\left(-\frac{r}{H_r}\right) \exp\left(-\frac{z}{H_z}\right)$$

where:

Parameter	Definition	Unit	Keyword
ρ_0	Reference plasma number density	cm^{-3}	RHO0
r	Equatorial radial distance	R_p	
z	Altitude above equator	R_p	
H_r	Equatorial radial scale-height	R_p	PERP
H_z	Vertical scale-height	R_p	SCALE

4.3.4 Torus Model

The Torus density profile is modeled as:

$$\rho = \rho_0 \exp\left(-\frac{\sqrt{(r - r_0)^2 + z^2}}{H}\right)$$

where:

Parameter	Definition	Unit	Keyword
ρ_0	Reference plasma number density	cm^{-3}	RHO0
r	Equatorial radial distance	R_p	
z	Altitude above equator	R_p	
r_0	Torus center equatorial diameter	R_p	PERP
H	Torus scale-height	R_p	SCALE

Example: We define the Io torus, with a peak reference density of 2000 cm^{-3} , an equatorial diameter of 5.91 Jovian Radii (orbit of Io) and a torus scale-height of 1 Jovian radius, as defined in [Bag94].

```
{
  "ON": true,
  "NAME": "Body1_density2",
  "TYPE": "Torus",
  "RHO0": 2000,
  "SCALE": 1,
  "PERP": 5.91
}
```

4.4 Magnetic Field Models

The detailed magnetic field models available for ExPRES are listed in the [LESIA_mag](#) repository. We recall below the list of models and the related references.

Planet	Magnetic Field Model		Current Sheet Model		BODY.MAG Value
	Short Name	Reference	Model Name	Reference	
Mercury	A12	[AJK+12]			A12
Jupiter	JRM33	[CTO+22]	CON20	[CTHJ20]	JRM33
	ISaAC	[HBZG11]	CAN81	[CAN81]	ISaAC+Connerney CS
	JRM09	[CKO+18]			JRM09+Connerney CS
	O6	[Con92]			O6+Connerney CS
	VIP4	[CANS98]			VIP4+Connerney CS
	VIPAL	[HBBL17]			VIPAL+Connerney CS
	VIT4	[Con07]			VIT4+Connerney CS
Saturn	SPV	[DS90]			
	Z3	[CAN84]			Z3
Uranus	AH5	[Her09]			AH5
	Q3	[CAN87]			Q3

REFERENCES

INDICES AND TABLES

- genindex
- modindex
- search

BIBLIOGRAPHY

- [AJK+12] Brian J. Anderson, Catherine L. Johnson, Haje Korth, Reka M. Winslow, Joseph E. Borovsky, Michael E. Purucker, James A. Slavin, Sean C. Solomon, Maria T. Zuber, and Ralph L. McNutt Jr. Low-degree structure in mercury’s planetary magnetic field. *Journal of Geophysical Research: Planets*, 2012. doi:10.1029/2012JE004159.
- [Bag94] Fran Bagenal. Empirical model of the io plasma torus: voyager measurements. *Journal of Geophysical Research: Space Physics*, 99(A6):11043–11062, 1994. doi:10.1029/93JA02908.
- [BGB+17] B. Bonfond, D. Grodent, S.V. Badman, J. Saur, J.-C. Gérard, and A. Radioti. Similarity of the jovian satellite footprints: spots multiplicity and dynamics. *Icarus*, 292:208–217, 2017. doi:10.1016/j.icarus.2017.01.009.
- [BGG+09] B. Bonfond, D. Grodent, J.-C. Gérard, A. Radioti, V. Dols, P. A. Delamere, and J. T. Clarke. The io uv footprint: location, inter-spot distances and tail vertical extent. *Journal of Geophysical Research: Space Physics*, 114(A7):, 2009. doi:10.1029/2009JA014312.
- [Con92] J. E. P. Connerney. Doing more with Jupiter’s magnetic field. In S. J. Bauer and H. O. Rucker, editors, *Planetary Radio Emissions III*, pages 13–33. Austria Acad. of Sci. Press, Vienna, 1992. URL: <https://austriaca.at/?arp=0x0015cd0e>.
- [Con07] J. E. P. Connerney. Planetary Magnetism. In Tilman Spohn, editor, *Treatise on Geophysics: Planets and Moons*, volume 10, pages 243–280. Elsevier, 2007.
- [CAN81] J. E. P. Connerney, M. H. Acuna, and N. F. Ness. Modeling the Jovian current sheet and inner magnetosphere. *J. Geophys. Res.*, 86:8370–8384, 1981. doi:10.1029/JA086iA10p08370.
- [CAN84] J. E. P. Connerney, M. H. Acuna, and N. F. Ness. The Z3 Model of Saturn’s Magnetic Field and the Pioneer 11 Vector Helium Magnetometer Observations. *J. Geophys. Res.*, 89:7541–7544, 1984. doi:10.1029/JA089iA09p07541.
- [CAN87] J. E. P. Connerney, M. H. Acuna, and N. F. Ness. The magnetic field of Uranus. *J. Geophys. Res.*, 92:15329–15336, 1987. doi:10.1029/JA092iA13p15329.
- [CANS98] J. E. P. Connerney, M. H. Acuna, N. F. Ness, and T. Satoh. New Models of Jupiter’s Magnetic Field Constrained by the Io Flux Tube Footprint. *J. Geophys. Res.*, 103:11929–11939, 1998. doi:10.1029/97JA03726.
- [CKO+18] J. E. P. Connerney, S. Kotsiaros, R. J. Oliverson, J. R. Espley, J. L. Joergensen, P. S. Joergensen, J. M. G. Merayo, and et al. A New Model of Jupiter’s Magnetic Field From Juno’s First Nine Orbits. *Geophys. Res. Lett.*, 45:2590–2596, 2018. doi:10.1007/s11214-009-9621-7.
- [CTHJ20] J. E. P. Connerney, S. Timmins, M. Hecceg, and J. L. Joergensen. A Jovian Magnetodisc Model for the Juno Era. *Journal of Geophysical Research (Space Physics)*, 125(10):e28138, October 2020. doi:10.1029/2020JA028138.

- [CTO+22] J. E. P. Connerney, S. Timmins, R. J. Oliverson, J. R. Espley, J. L. Joergensen, S. Kotsiaros, P. S. Joergensen, J. M. G. Merayo, M. Herceg, J. Bloxham, K. M. Moore, A. Mura, A. Moirano, S. J. Bolton, and S. M. Levin. A New Model of Jupiter's Magnetic Field at the Completion of Juno's Prime Mission. *Journal of Geophysical Research (Planets)*, 127(2):e07055, February 2022. doi:10.1029/2021JE007055.
- [DS90] Davis, L. Jr and E. Smith. A Model of Saturn's Magnetic Field Based on All Available Data. *J. Geophys. Res.*, 95:15257–15261, 1990. doi:10.1029/JA095iA09p15257.
- [Her09] F. Herbert. Aurora and magnetic field of uranus. *J. Geophys. Res.*, 2009. doi:10.1029/2009JA014394.
- [HBBL17] S. L. G. Hess, B. Bonfond, F. Bagenal, and L. Lamy. A Model of the Jovian Internal Field Derived from in-situ and Auroral Constraints. In *PRE8 Proceedings*, volume 8. Austrian Academy of Science, 2017. doi:10.1553/PRE8s157.
- [HBZG11] S. L. G. Hess, B. Bonfond, P. Zarka, and D. Grodent. Model of the Jovian Magnetic Field Topology Constrained by the Io Auroral Emissions. *J. Geophys. Res.*, 116:177, 2011. doi:10.1029/2010JA016262.
- [HCZ08] SLG Hess, B Cecconi, and Philippe Zarka. Modeling of Io-Jupiter Decameter Arcs, Emission Beaming and Energy Source. *Geophys. Res. Lett.*, 2008. doi:10.1029/2008GL033656.
- [HTK98] David P. Hinson, Joseph D. Twicken, and E. Tuna Karayel. Jupiter's ionosphere: new results from voyager 2 radio occultation measurements. *Journal of Geophysical Research: Space Physics*, 103(A5):9505–9520, 1998. doi:10.1029/97JA03689.
- [HBB19] P. C. Hinton, F. Bagenal, and B. Bonfond. Alfvén wave propagation in the io plasma torus. *Geophysical Research Letters*, 46(3):1242–1249, 2019. doi:10.1029/2018GL081472.
- [HGL+23] V. Hue, G. R. Gladstone, C. K. Louis, T. K. Greathouse, B. Bonfond, J. R. Szalay, A. Moirano, R. S. Giles, J. A. Kammer, M. Imai, A. Mura, M. H. Versteeg, G. Clark, J. -C. Gérard, D. C. Grodent, J. Rabia, A. H. Sulaiman, S. J. Bolton, and J. E. P. Connerney. The Io, Europa, and Ganymede Auroral Footprints at Jupiter in the Ultraviolet: Positions and Equatorial Lead Angles. *Journal of Geophysical Research (Space Physics)*, 128(5):e2023JA031363, May 2023. doi:10.1029/2023JA031363.
- [LHC+19] C K Louis, S L G Hess, B Cecconi, P Zarka, L Lamy, S Aicardi, and A Loh. ExPRES: an Exoplanetary and Planetary Radio Emissions Simulator. *Astronomy and Astrophysics*, 627:A30, 2019. doi:10.1051/0004-6361/201935161.
- [LLZ+17a] C K Louis, L Lamy, P Zarka, B Cecconi, and S L G Hess. Detection of Jupiter decametric emissions controlled by Europa and Ganymede with Voyager/PRA and Cassini/RPWS. *J. geophys. Res. Space Physics*, 122:1–20, September 2017. doi:10.1002/2016JA023779.
- [LLZ+17b] C K Louis, L Lamy, P Zarka, B Cecconi, M Imai, W S Kurth, G Hospodarsky, S L G Hess, X Bonnin, S J Bolton, J E P Connerney, and S M Levin. Io-Jupiter decametric arcs observed by Juno/Waves compared to ExPRES simulations. *Geophys. Res. Lett.*, pages 1–17, 2017. doi:10.1002/2017GL073036.
- [Wu85] Chi-Wei Wu. Kinetic cyclotron and synchrotron Maser instabilities: Radio emission processes by direct amplification of radiation. *Space Sci. Rev.*, 41:215–298, 1985.