

# Proposta de Padronização Para Entrada e Saída de Dados

Grupo de Propagação - GAPTEM

May 28, 2021

# Objeto

Definir uma padronização a ser utilizada no Grupo de Propagação do GAPTEM, que permita o intercâmbio de informações entre os diversos *solvers* que venham a ser desenvolvidos dentro do grupo. Devendo contemplar:

- Configuração da simulação
- Entrada de dados
- Resultados da simulação

*\*Propõem-se que todas as variáveis sejam apresentadas utilizando a unidade padronizada pelo S.I., no formato inteiro ou ponto flutuante*

## Configuração da simulação

Propõe-se a adoção do formato Json pelos seguintes motivos:

- **JSON** (*JavaScript Object Notation*) é um modelo para armazenamento e transmissão de informações no formato texto.
- Permite a criação de forma compacta e estruturada dos parâmetros desejados, preservando o tipo da variável.
- Sua leitura é de fácil implementação e bastante difundida em várias linguagens de programação.

# Importar *.json* no Matlab ®

- No Matlab® um arquivo *.json* pode ser convertido para um objeto do tipo *struct*
- Os arquivos *.json* são importados através do comando:  

```
var_x = jsondecode( fileread ( 'file_name.json' ))
```

# Importar *.json* no Matlab®

Exemplo: conf\_teste.json

conf\_teste.json

```
1 {  
2   "a" : 123,  
3   "b" : "teste",  
4   "c" : 1.5E+3  
5 }
```

```
>> var_teste = jsondecode(fileread('conf_teste.json'))  
  
var_teste =  
  
    struct with fields:  
  
        a: 123  
        b: 'teste'  
        c: 1500  
  
>> var_teste.a  
  
ans =  
  
    123  
  
>> var_teste.b  
  
ans =  
  
    'teste'  
  
>> var_teste.c  
  
ans =  
  
    1500
```

# Importar *.json* em Python

- Em Python um arquivo *.json* pode ser convertido para um objeto do tipo *dictionary*.
- A estrutura dos arquivos *.json* é compreendida utilizando a biblioteca "json":

```
import json
f = 'file_name.json'
with open(f) as json_conf :
    var_x = json.load(f)
```

# Importar *.json* em Python

Exemplo: conf\_teste.json

conf\_teste.json

```
1 {  
2   "a" : 123,  
3   "b" : "teste",  
4   "c" : 1.5E+3  
5 }
```

```
In [20]: import json
```

```
In [21]: with open(conf_file) as json_conf :  
...:     var_teste = json.load(json_conf)  
...:
```

```
In [22]: var_teste['a']  
Out[22]: 123
```

```
In [23]: var_teste['b']  
Out[23]: 'teste'
```

```
In [24]: var_teste['c']  
Out[24]: 1500.0
```

# Arquivo de configuração

- Propõe-se que o nome do arquivo seja:  
*conf\_{**frequencia central**}.json*  
p.ex: conf\_400MHz.json, conf\_15GHz.json
- As informações dos parâmetros de simulação sejam apresentados em blocos.



# Estrutura do arquivo de configuração

```
{
  "general" : {
    "length" : 15000,
    "height" : 300,
    "center frequency" : 500.0E+6,
    "bandwidth" : 50E3,
    "elevation profile" : "location_A_elev.mat",
    "buildings_heights" : "location_A_bh.mat",
    "atmospheric reflectivity" : "location_A_ra.mat",
    "clutter" : "location_A_clutter.mat"
  },
  "principal" : {
    "x" : 10,
    "y" : 60,
    "tx power" : 1E-3,
    "antenna pattern" : "factoryX_model50b.mat",
    "azimuth" : 10,
    "tilt" : 4
  },
  "probe" : {
    "horizontal" : {
      "y" : 3,
      "threshold_dbm" : -115,
      "x_start_interval" : 1000,
      "x_end_interval" : 3000
    },
    "vertical" : {
      "x" : 3000,
      "threshold_dbm" : -115,
      "y_start_interval" : 50,
      "y_end_interval" : 300
    }
  },
  "optional" : {
    "clutter_epsr" : "location_A_epsr.mat",
    "clutter_sigma" : "location_A_sigma.mat",
    "window_size" : 1000
  }
}
```

# Entrada de Dados

São considerados dados as informações descritivas de:

- Elevação do Terreno
- Altura das Edificações
- Tipo de Cobertura do Terreno
- Diagramas de irradiação das antena
- Comportamento do Índice de Refratividade da Atmosfera

# Entrada de Dados

Propõem-se a utilização de *m-files*.

Cada informação em um arquivo distinto contendo:

- Uma matriz  $1 \times N$  contendo a posição relativa da informação
- Uma matriz  $1 \times N$  contendo a informação relativa ao ponto de coleta
- Uma variável de dados informando o passo utilizado

## Dados ao longo do eixo X

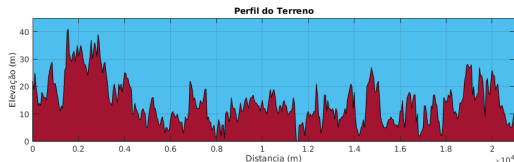
Para as informações ao longo do eixo X, p.ex. elevação, clutter, altura de edificações, a referência é a estação principal (p.ex eNB, transmissora de radiodifusão).

O *m-file* deve conter três variáveis:

- uma matriz  $1 \times N$  contendo as distâncias relativas das amostras tomando a estação principal como referência denominada *distancia*
- uma matriz  $1 \times N$  contendo os valores do parâmetro amostrado nos pontos cujas distâncias estão indicadas pelo mesmo índice na matriz *distancia*. O nome desta matriz segue o nome do parâmetro observado em formato *ASCII* (caixa baixa, sem acentos).
- O passo utilizado para amostragem.

Para as informações de classificação de cobertura do terreno, sugere-se adoção dos códigos do *Land Cover Classification System* (LCCS).

# Dados ao longo do eixo X - exemplo



**Select variables to import using checkboxes**

☒ Create variables matching preview.  
☐ Create vectors from each column using column names.  
☐ Create vectors from each row using row names.

**Variables in /home/mlnunes/Documents/doutorado/Grupo de Propagação/Padronização/eleva...**

Import	Name	Size	Bytes	Class
<input checked="" type="checkbox"/>	distancia	1x422	3376	double
<input checked="" type="checkbox"/>	elevacao	1x422	3376	double
<input checked="" type="checkbox"/>	passo	1x1	8	double

No variable selected for preview.

☐ Generate MATLAB code

## Dados ao longo do eixo Y

Para as informações ao longo do eixo Y, p.ex. índice de refratividade da atmosfera, a referência é nível do mar.

O *m-file* deve conter três variáveis:

- uma matriz  $1 \times N$  contendo a faixa de altitude de coleta das amostras, denominada *altitude*
- uma matriz  $1 \times N$  contendo os valores do parâmetro amostrado nos pontos cujas altitudes estão indicadas pelo mesmo índice na matriz *altitude*. O nome desta matriz segue o nome do parâmetro observado em formato *ASCII* (caixa baixa, sem acentos).
- O passo utilizado para amostragem.

# Diagramas de irradiação de antenas

O *m-file* deve conter três variáveis:

- uma matriz  $1 \times N$  contendo o azimuth denominado *azimute*
- uma matriz  $1 \times N$  contendo os valores do ganho no plano H amostrados nos pontos cujos azimuths estão indicadas pelo mesmo índice na matriz *azimute*, denominada *plano\_h*.
- uma matriz  $1 \times N$  contendo os valores do ganho no plano V amostrados nos pontos cujos azimuths estão indicadas pelo mesmo índice na matriz *azimute*, denominada *plano\_v*.
- O passo utilizado para amostragem.

## Saída de dados

Propõe-se que os dados de saída sejam disponibilizados em formato *M-file* contendo na descrição a variável amostrada e a grandeza utilizada.



# Aguardo sugestões.

mlnunes@ufmg.br

O arquivo fonte desta apresentação em formato  $\text{\LaTeX}$  está disponível em [https://github.com/mlnunes/padronizacao\\_grupo\\_propagacao.git](https://github.com/mlnunes/padronizacao_grupo_propagacao.git).