# 电磁场小组研究报告

# ——探究电磁波在低层大气中的各种衰减情况

陈梓彧 龙文振 郭珺碧 冯国艺 邓皓阳 易俊哲

### 研究背景

电磁波在低层大气中的传播是无线通信、遥感探测和雷达系统等领域的关键因素。由于大气中的气体分子、水滴、冰晶等微粒的存在，电磁波在传播过程中会经历吸收、散射等衰减现象，从而影响信号的强度和质量。本研究旨在探究电磁波在低层大气中的衰减特性，以更加深入地了解电磁场与电磁波的相关知识。

根据参考资料给出的相关具体公式，我们将使用matlab建立数学模型来了解不同频率电磁波产生的衰减之间的差异。

### 基础概念

1. **比衰减**

气体比衰减‌是指气体在传播过程中能量逐渐减少的现象。这种现象主要由于气体分子的吸收和散射作用导致。在任何气压、任何温度和任何湿度下，累加氧气和水蒸气各自谱线，可以准确地计算主要由于干空气和水蒸气所造成的高达1 000 GHz频率上的比衰减。

1. **地面路径衰减**

地面路径衰减‌是指无线电波在地表附近传播时，由于大气中的物质（如气体分子、水蒸气、悬浮颗粒物等）对电磁波的吸收和散射作用而导致的信号强度减弱现象。

1. **倾斜路径衰减**

倾斜路径衰减是指无线电波在从地面站到卫星或其他高空平台的传播过程中，由于大气成分（主要是氧气和水蒸气）吸收而导致信号强度减弱的现象。

1. **色散**

是指波的相位速度或群速度随频率变化的现象。在物理学中，色散描述了波在介质中传播时，不同频率成分的传播速度不同，导致波形随时间扩展或压缩的现象。上述衰减均为基于频率相关复折射率虚部的衰减，色散则是基于频率相关复折射率实部的衰减。

### 参考文献

ITU-R P.676-13 建议书

ITU-R P.835 建议书

ITU-R P.453-13 建议书

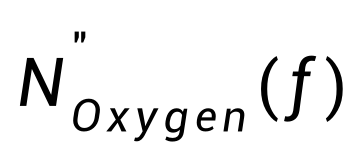
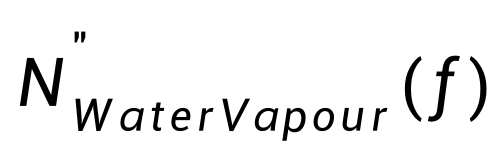
### 研究内容

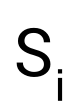
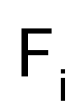
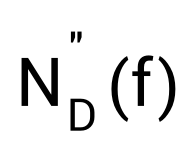
1. **比衰减**

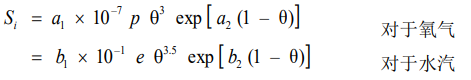
在任何气压、任何温度和任何湿度下，累加氧气和水蒸气各自谱线，可以准确地计算主 要由于干空气和水蒸气所造成的高达1 000 GHz频率上的比衰减。这一方法同时也考虑了一 些其他相对影响较小的因素，如10 GHz以下氧气的非共振的德拜频谱，100 GHz以上的主要 由气压造成的氮衰减和计算实验上发现的过多水蒸气吸收的潮湿连续带。并且在研究过程中使用标准的大气参数，即气压1 013.25 hPa、温度：15℃、水蒸气密度为7.5 g/m3。计算频率从1~1000GHz，步长为1GHz的比衰减。

**气体比衰减公式**如下：



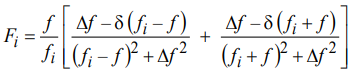
f（单位GHz）是频率，和是该频率相关的复折射率的虚部： 

是**第i条氧气或水蒸气谱线强度**，是**氧气或水蒸气谱线形状因子**，对表1氧气衰减的谱线数据（见附录）和表2水蒸气衰减的谱线数据（见附录）中的所有谱线进行求和；是由**气压造成的氮吸收**以及德拜频谱产生的干空气连续吸收谱。相关公式如下：

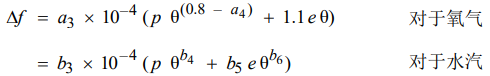


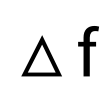
任何高度的水蒸气分压e可通过该高度的水蒸气密度和温度T计算得到

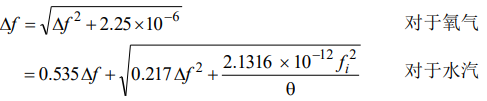




是**氧气或水蒸气谱线频率**，是**谱线宽度：**



修改**谱线宽度**，将氧气谱线的塞曼分裂和水蒸气谱线的多普勒展宽考虑在内公式变换如下：



由于在氧气线中的干扰影响，latexmath是修正因子：

**干空气分压p**：单位为hPa；**水蒸气分压e**，单位为hPa；**总气压=水蒸气分压+干空气气压**；**θ= 300/T** ；**T：温度**，单位为K。

1. **地面路径衰减**

对于**地面路径**，或者是微小倾斜的接近于地面的倾斜路径，**衰减公式**如下：



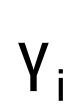
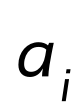
A为路径衰减值（dB），γ为比衰减（dB/km），r0 为路径长度（km）。

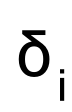
1. **倾斜路径衰减**

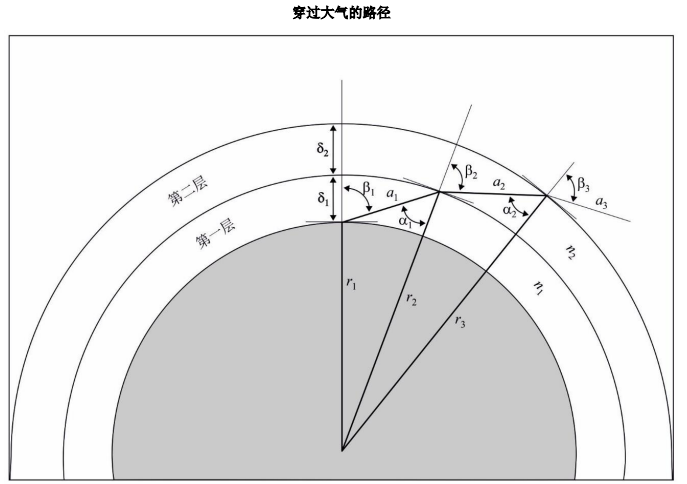
**（1）非负视在仰角**

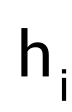
倾斜路径气体衰减可通过把大气分成指数级增长的层，确定每层的比衰减（dB/km）和通过每层的路径长度（千米），对每一层的比衰减和通过每一层的路径长度的乘积求和来进行十分近似的估计，公式如下。在实验中为了方便以及适用性我们选择了使用ITU-R P.835建议书中的全球年平均参考大气来作为我们的大气数据。

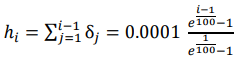


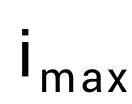
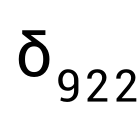
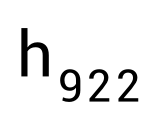
其中是第i层比衰减（dB/km），是通过第i层的路径长度（单位为km）。

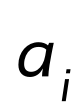
对于地球表面和空间之间的倾斜路径，参考下图的几何结构，各层的厚度从地表10厘米到100千米高度，呈指数级增加，以确保对总倾斜路径气体衰减的准确估算。**第i层的厚度**为： 



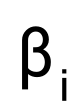
**第i层的底部高度**为：



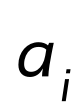
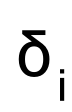
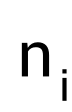
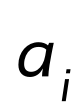
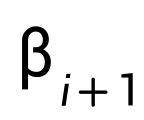
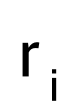
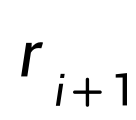
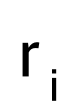
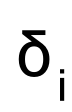
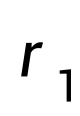
因为使用了ITU-R P.835建议书中的全球年平均参考大气，所以对高达100千米的几何高度规定大气剖面，在这种情况下， = 922,  = 0.999 66 km, = 99.457 km

**路径长度**为：



在极坐标中**使用Snell定律直接计算**，如下所示：



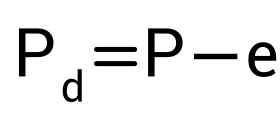
是穿过厚度为的第𝑖层的路径长度，是使用ITU-R P.453建议书公式计算的第𝑖层干空气气压、温度和水蒸气分压之间的函数。和是第𝑖层和第(𝑖 + 1)层之间界面的入射角和出射角，是从地球中心到第𝑖层起点的半径， =  + , 是从地球中心到最 底层起点的半径，通常是平均地球半径（6 371千米）。

与**折射率n相关公式**，**大气无线电折射率n**：



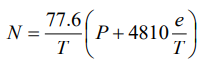
其中**N为无线电折射率**，计算公式为：



因为所以还可以表示为：



为了方便计算我们运用简化后的式子，尽管精确度有所降低，但符合要求：



**参考大气相关内容：**

大气温度和压力分布以两种高度形式定义：1) 从0公里至84.852公里′的位势高度（H），和2) 从86公里至100公里的几何高度（Z）。1976年美国标准大气采用的位势高度H（公里）与几何高度Z（公里）之间的**转换公式**为：

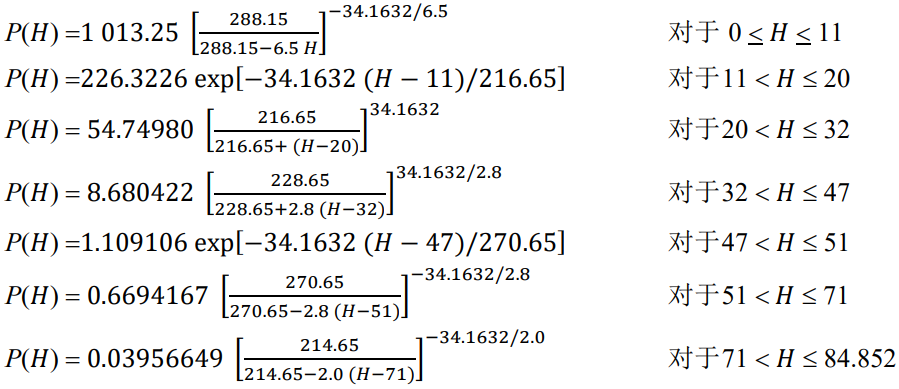
****

由于676参考书中的公式用的都是几何高度所以我们需要把公式里的位势高度H都转化为几何高度Z。

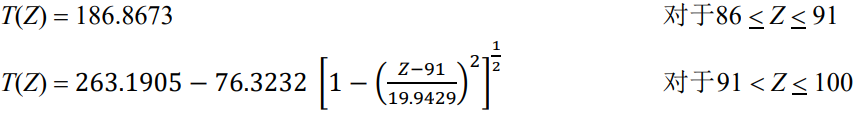
第一高度方案中，**位势高度公里′处的温度T（K）为**：



**位势高度𝐻（公里′）处的压力P（hPa）**：



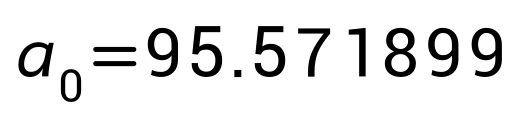
在第二高度区域中，**几何高度Z（公里）处的温度T (K)**为：

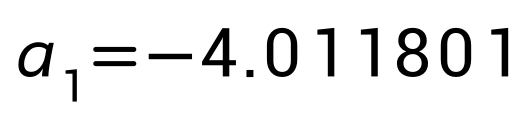


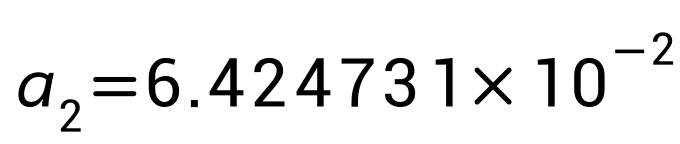
**几何高度Z（公里）处的压力P（hPa）**为：

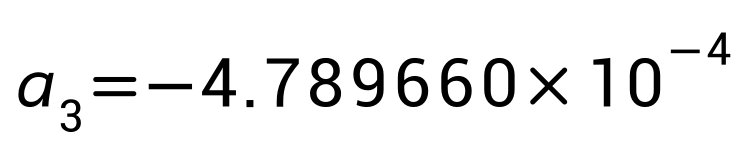


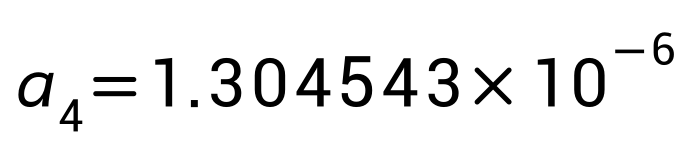
式中：









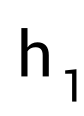
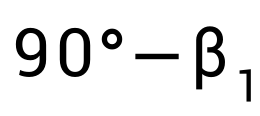


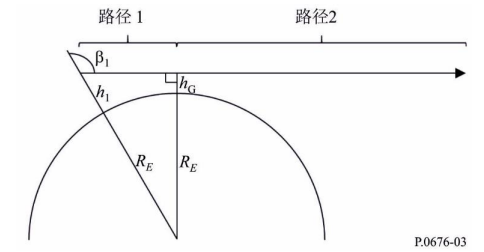
大气中的**水蒸气密度latexmath与几何高度的关系**可通过以下负指数分布得出近似值：

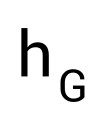


**水蒸气分压𝑒(𝑍): **

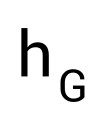
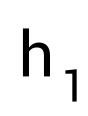
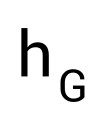
1. **负视在仰角**

下图显示了高度，视在仰角为的地球站的这一情况。



**掠射高度**可通过迭代求解下示公式确定。对于该特定大气剖面（通常是ITU-RP.835建议书的参考分布图之一），**折射率𝑛(ℎ)**可通过上节陈述的公式计算。

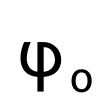
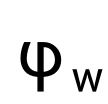
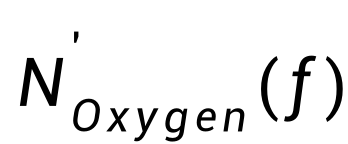
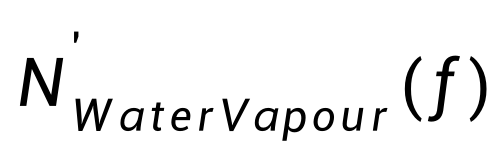


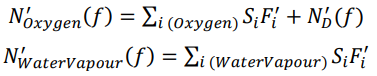
净气体衰减是路径1和路径2的气体衰减总和。路径1为视在仰角为0度，在高度千米的虚拟地球站和在高度千米的实际地球站之间的气体衰减，路径2为视在仰角为0度，在高度千米的虚拟地球站和最大大气高度（通常为100千米）之间的气体衰减。 这样就把路径分成了两个非负视在仰角的路径就可用上节的公式进行计算。

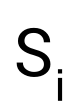
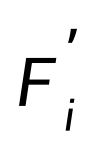
1. **色散**

除上段所述基于频率相关复折射率虚部的衰减外，氧气和水蒸气产生基于频率相关复折 射率实部的色散。通过相对于频率的相位色散（deg/km）或相对于频率的群延迟（ps/km） 对这一效应进行描述；与衰减类似，亦可计算倾斜路径色散。与比衰减的公式类似可以得到**色散的公式**如下：

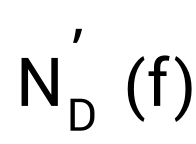


其中，是干空气造成的特定相位色散(deg/km)，是水蒸气造成的特定相位色散； f是频率（GHz）；和是基于**频率相关复折射率的实部**：

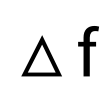


其中与比衰减中的是一致的，是**第i条氧气或水蒸气谱线强度**，直接使用比衰减处的公式，是**氧气或水蒸气谱线形状因子的实部**：



是由气压造成的氮吸收产生的干空气连续吸收谱的实部：



,latexmath,latexmath均已在比衰减处定义，运用相应的公式即可算出。

### 研究结果

1. **比衰减**

频率与衰减之间的关系如下图

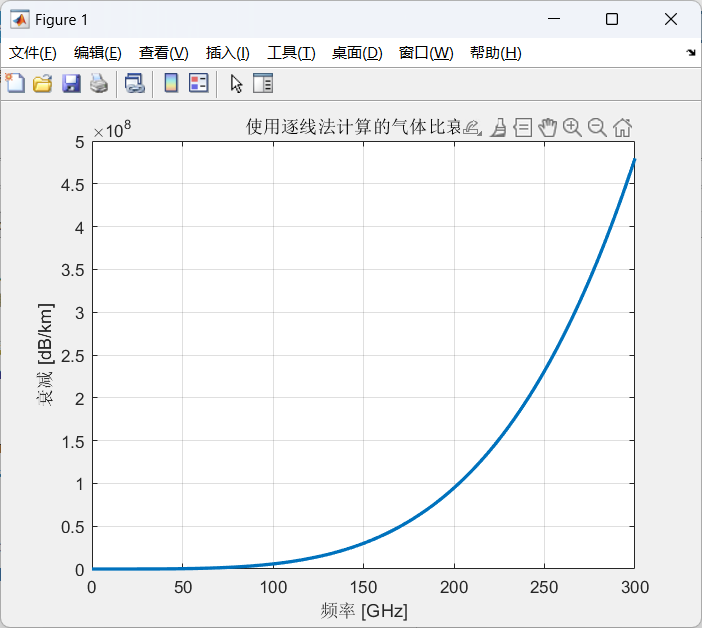


图1 比衰减随频率变化

1. **地面路径衰减**

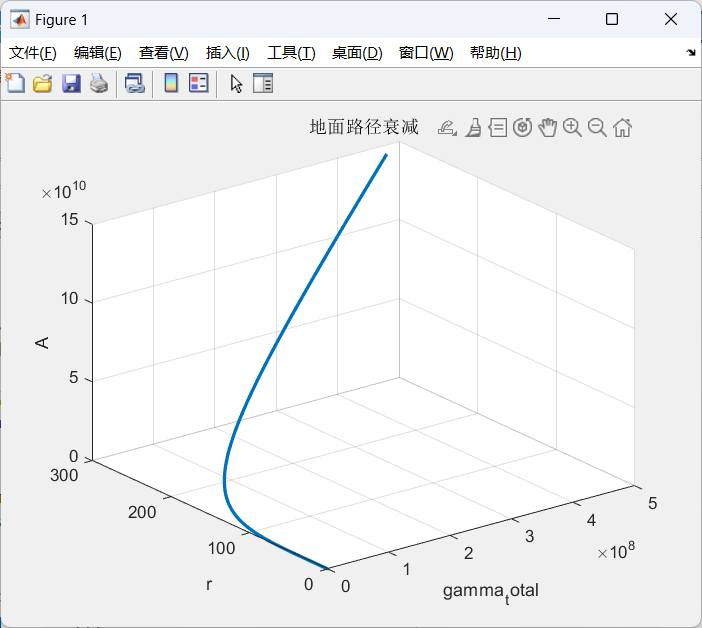
****

图2 地面路径衰减与传播距离和比衰减的关系

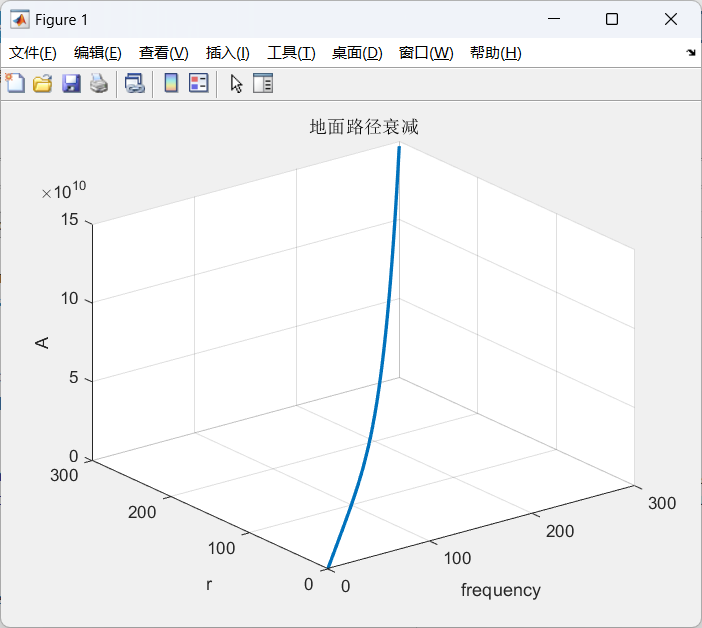


图3 地面路径衰减与传播距离和频率的关系

1. **倾斜路径衰减**

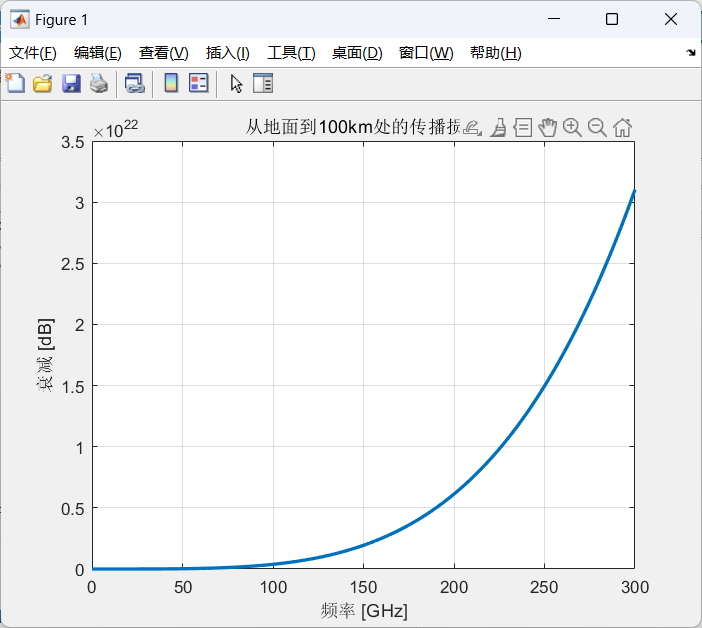
****

图4 倾斜路径衰减（高度固定100km算不同频率衰减）

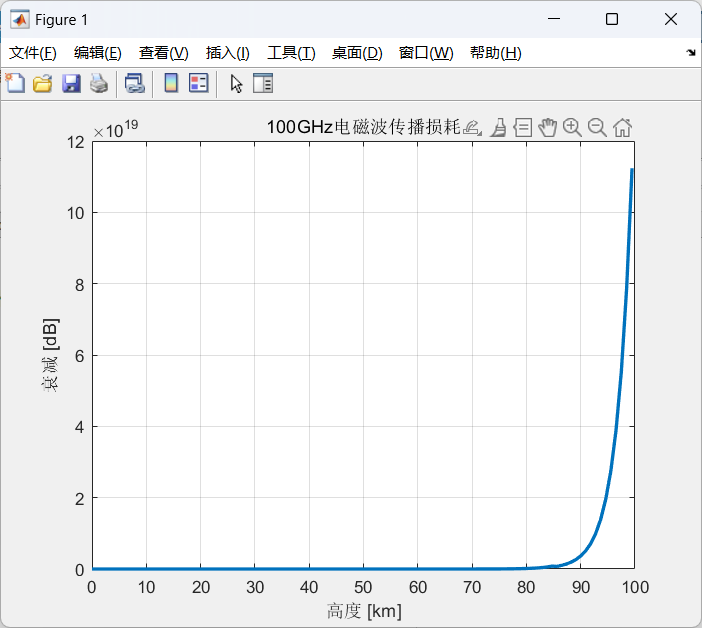
****

图5 倾斜路径衰减（固定100GHz算不同高度衰减）

1. **色散**

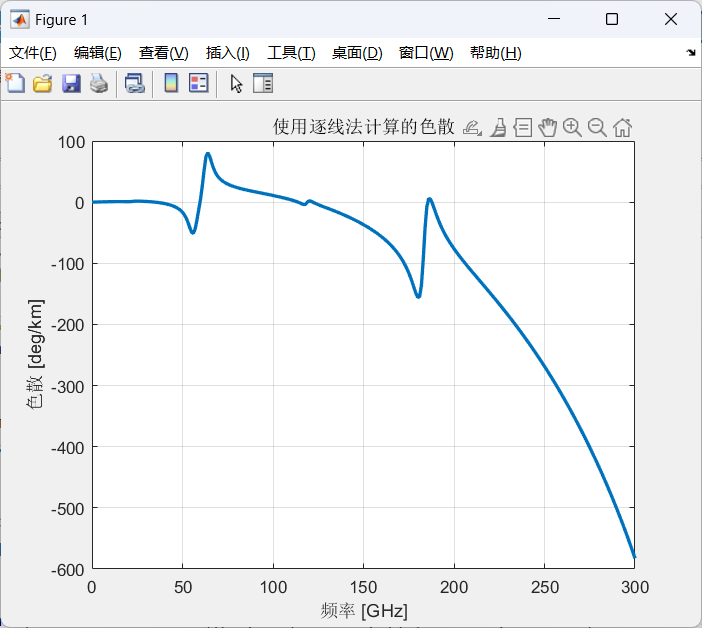


图6 色散随频率的变化

### 结论分析

在这次研究中我们讨论了电磁波在低层大气中的传播特性，特别是关注不同频率下电磁波所经历的各种衰减现象。通过使用MATLAB建立数学模型，并结合ITU-R发布的多个建议书（如P.676-13、P.835、P.453-13）中提供的标准公式和方法，我们对气体比衰减、地面路径衰减、倾斜路径衰减以及色散进行了分析和计算。

1. **比衰减**

由图一可知，在0到300GHz的频率范围内，随着频率的增加，比衰减呈现出指数级增长的趋势。这一结果表明，在高频段内，电磁波更容易受到大气成分的影响而发生能量损失。

1. **地面路径衰减**

由图二，图三两个三维图可知，地面路径传播的衰减与传播距离和信号频率都有关系。地面路径上的信号强度会随着传播距离的增加而逐渐减弱。对于较短的距离和较低的频率，衰减相对较小；但当频率升高或传播距离变长时，衰减显著加剧。

1. **倾斜路径衰减**

对于从地面站到卫星或其他高空平台之间的倾斜路径，氧气和水蒸气是导致信号强度减弱的主要因素。我们的研究表明，在非负视在仰角和负视在仰角的情况下，都存在明显的气体吸收效应，这需要在设计通信系统时加以考虑。由图四、图五可知，在固定传播距离时，信号衰减随着传播信号频率的增加指数级增长，在频率到达130GHz时快速增长，在固定传播频率时，衰减在传播距离到90km后快速增长。

1. **色散**

色散现象反映了不同频率成分在介质中的传播速度差异，我们在本次实验中展示了色散随频率变化的趋势。这对于理解信号波形的变化具有重要意义。由图六可知，在0到300GHz的频率范围内，随着频率的增加，色散衰减在60GHz和175GHz附近出现明显上下波动，但整体仍呈现下降的趋势。

通过对比衰减、地面路径衰减、倾斜路径衰减以及色散这四个方面的研究，我们提升了对电磁波在低层大气中传播行为的理解。并且基于这些模型我们可以对一定频率内倾斜路径气体衰减进行预测。

### 小组分工

龙文振：代码编写，部分研究报告编写。

郭珺碧：研究报告编写，部分代码编写

易俊哲：

陈梓彧：

冯国艺；

邓皓阳：

### 附录

**实验代码如下：**

1. **比衰减**

% 定义常量和参数

frequencies = 0:1:300; % 频率范围从0到1000 GHz，步长为1 GHz

pressure = 1013.25-waterVaporPressure; % 干空气气压，单位为hPa

temperature = 15 + 273.15; % 温度，单位为开尔文（15摄氏度）

waterVaporDensity = 7.5; % 水蒸气密度，单位为g/m^3

waterVaporPressure = waterVaporDensity \* temperature / 216.7; % 水蒸气分压

xiu = 0;

% 加载氧气和水蒸气的谱线数据

oxygenData = readSpectralLinesFromExcel('oxygen\_spectral\_lines.xlsx');

waterVaporData = readSpectralLinesFromExcel('water\_vapor\_spectral\_lines.xlsx');

% 使用逐线法计算特定频率下的比衰减

gamma\_oxygen = zeros(size(frequencies));

gamma\_waterVapor = zeros(size(frequencies));

for i = 1:length(frequencies)

gamma\_oxygen(i) = calculateLineByLineAttenuationoxygen(frequencies(i), pressure, waterVaporPressure, temperature, oxygenData);

gamma\_waterVapor(i) = calculateLineByLineAttenuationwater(frequencies(i), pressure, waterVaporPressure, temperature, waterVaporData);

end

% 总的比衰减

gamma\_total = 0.1820 .\* frequencies .\* (gamma\_oxygen + gamma\_waterVapor);

% 绘制比衰减图

figure;

plot(frequencies, gamma\_total, 'LineWidth', 2);

title('使用逐线法计算的气体比衰减');

xlabel('频率 [GHz]');

ylabel('衰减 [dB/km]');

grid on;

% 读取Excel文件中的谱线数据

function data = readSpectralLinesFromExcel(filename)

% 从指定的Excel文件读取谱线数据

% 输入：

% filename - Excel文件名

% 读取Excel文件中的数据，默认读取第一个工作表，并确保跳过标题行

tbl = readtable(filename);

% 将表格数据转换为结构体

data.f0 = tbl.f0; % 假设第一列为f0

data.b1 = tbl.b1; % 假设第二列为b1

data.b2 = tbl.b2; % 假设第三列为b2

data.b3 = tbl.b3; % 假设第四列为b3

data.b4 = tbl.b4; % 假设第五列为b4

data.b5 = tbl.b5; % 假设第六列为b5

data.b6 = tbl.b6; % 假设第七列为b6

end

% 基于逐线法计算氧气比衰减

function n\_prime = calculateLineByLineAttenuationoxygen(frequency, pressure, waterVaporPressure, temperature, spectralLines)

% 此函数基于逐线法计算复折射率的虚部。

% 输入：

% frequency - 频率，单位为GHz

% pressure - 压力，单位为hPa

% temperature - 温度，单位为K

% spectralLines - 包含谱线数据的结构体

theta = 300 / temperature;

% 初始化N''的求和变量

N\_doublePrime = 0;

% 遍历所有谱线

for j = 1:length(spectralLines.f0)

S = spectralLines.b1(j) \* 10^(-7) \* pressure \*theta^(3)\* exp(spectralLines.b2(j)\*(1 - theta));

fi = spectralLines.f0(j);

deltaF = spectralLines.b3(j)\*10^(-4)\*(pressure\*theta^(0.8-spectralLines.b4(j))+1.1\*waterVaporPressure\*theta);

deltaF = sqrt(deltaF^(2)+2.25\*10^(-6));

xiu = (spectralLines.b5(j)+spectralLines.b6(j)\*theta)\*10^(-4)\*(pressure+waterVaporPressure)\*theta^(0.8);

F = frequencyLineShapeFactor(frequency, fi, deltaF,xiu);

N\_doublePrime = N\_doublePrime + S \* F;

end

d =5.6\*10^(-4)\*(pressure+waterVaporPressure)\*theta^(0.8);

nitrogen = frequency\*pressure\*theta^(2)\*(6.14\*10^(-5)/d\*(1+(frequency/d)^(2))+ ...

1.4\*10^(-12)\*pressure\*theta^(1.5)/(1+1.9\*10^(-5)\*frequency^(15)));

n\_prime = N\_doublePrime+nitrogen;

end

% 基于逐线法计算水汽比衰减

function n\_prime = calculateLineByLineAttenuationwater(frequency, pressure, waterVaporPressure,temperature, spectralLines)

% 此函数基于逐线法计算复折射率的虚部。

% 输入：

% frequency - 频率，单位为GHz

% pressure - 压力，单位为hPa

% temperature - 温度，单位为K

% spectralLines - 包含谱线数据的结构体

theta = 300 / temperature;

% 初始化N''的求和变量

N\_doublePrime = 0;

% 遍历所有谱线

for j = 1:length(spectralLines.f0)

S = spectralLines.b1(j) \* 10^(-1) \* waterVaporPressure \*theta^(3.5)\* exp(spectralLines.b2(j)\*(1 - theta));

fi = spectralLines.f0(j);

deltaF = spectralLines.b3(j)\*10^(-4)\*(pressure\*theta^(spectralLines.b4(j))+ ...

spectralLines.b5(j)\*waterVaporPressure\*theta^(spectralLines.b6(j)));

deltaF = 0.535\*deltaF+sqrt(0.217\*deltaF^(2)+(2.1316\*10^(-12)\* fi^(2)/theta));

xiu = 0;

F = frequencyLineShapeFactor(frequency, fi, deltaF, xiu);

N\_doublePrime = N\_doublePrime + S \* F;

end

n\_prime = N\_doublePrime;

end

% 计算频率谱线形状因子

function F = frequencyLineShapeFactor(frequency, fi, deltaF, xiu)

% 计算频率谱线形状因子

F = (frequency/fi)\*((deltaF-xiu\*(fi-frequency))/((fi-frequency)^2 + deltaF^2) +...

(deltaF-xiu\*(fi+frequency))/((fi+frequency)^2 + deltaF^2));

end

**2、地面路径衰减**

% 定义常量和参数

r = 0:1:1000;% 距离范围从0到1000km，步长为1km

frequencies = 0:1:300; % 频率范围从0到300 GHz，步长为1 GHz

pressure = 1013.25-waterVaporPressure; % 干空气气压，单位为hPa

temperature = 15 + 273.15; % 温度，单位为开尔文（15摄氏度）

waterVaporDensity = 7.5; % 水蒸气密度，单位为g/m^3

waterVaporPressure = waterVaporDensity \* temperature / 216.7; % 水蒸气分压

xiu = 0;

% 加载氧气和水蒸气的谱线数据

oxygenData = readSpectralLinesFromExcel('oxygen\_spectral\_lines.xlsx');

waterVaporData = readSpectralLinesFromExcel('water\_vapor\_spectral\_lines.xlsx');

% 使用逐线法计算特定频率下的比衰减

gamma\_oxygen = zeros(size(frequencies));

gamma\_waterVapor = zeros(size(frequencies));

for i = 1:length(frequencies)

gamma\_oxygen(i) = calculateLineByLineAttenuationoxygen(frequencies(i), pressure, waterVaporPressure, temperature, oxygenData);

gamma\_waterVapor(i) = calculateLineByLineAttenuationwater(frequencies(i), pressure, waterVaporPressure, temperature, waterVaporData);

end

% 总的比衰减

gamma\_total = 0.1820 .\* frequencies .\* (gamma\_oxygen + gamma\_waterVapor);

%地面路径衰减

A = gamma\_total.\*r;

figure;

plot3(gamma\_total,r ,A , 'LineWidth', 2);

xlabel('gamma\_total');

ylabel('r');

zlabel('A');

title('地面路径衰减');

grid on;

% 读取Excel文件中的谱线数据

function data = readSpectralLinesFromExcel(filename)

% 读取Excel文件中的数据，默认读取第一个工作表，并确保跳过标题行

tbl = readtable(filename);

% 将表格数据转换为结构体

data.f0 = tbl.f0; % 假设第一列为f0

data.b1 = tbl.b1; % 假设第二列为b1

data.b2 = tbl.b2; % 假设第三列为b2

data.b3 = tbl.b3; % 假设第四列为b3

data.b4 = tbl.b4; % 假设第五列为b4

data.b5 = tbl.b5; % 假设第六列为b5

data.b6 = tbl.b6; % 假设第七列为b6

end

% 基于逐线法计算氧气比衰减

function n\_prime = calculateLineByLineAttenuationoxygen(frequency, pressure, waterVaporPressure, temperature, spectralLines)

% 此函数基于逐线法计算复折射率的虚部。

% 输入：

% frequency - 频率，单位为GHz

% pressure - 压力，单位为hPa

% temperature - 温度，单位为K

% spectralLines - 包含谱线数据的结构体

theta = 300 / temperature;

% 初始化N''的求和变量

N\_doublePrime = 0;

% 遍历所有谱线

for j = 1:length(spectralLines.f0)

S = spectralLines.b1(j) \* 10^(-7) \* pressure \*theta^(3)\* exp(spectralLines.b2(j)\*(1 - theta));

fi = spectralLines.f0(j);

deltaF = spectralLines.b3(j)\*10^(-4)\*(pressure\*theta^(0.8-spectralLines.b4(j))+1.1\*waterVaporPressure\*theta);

deltaF = sqrt(deltaF^(2)+2.25\*10^(-6));

xiu = (spectralLines.b5(j)+spectralLines.b6(j)\*theta)\*10^(-4)\*(pressure+waterVaporPressure)\*theta^(0.8);

F = frequencyLineShapeFactor(frequency, fi, deltaF,xiu);

N\_doublePrime = N\_doublePrime + S \* F;

end

d =5.6\*10^(-4)\*(pressure+waterVaporPressure)\*theta^(0.8);

nitrogen = frequency\*pressure\*theta^(2)\*(6.14\*10^(-5)/d\*(1+(frequency/d)^(2))+ ...

1.4\*10^(-12)\*pressure\*theta^(1.5)/(1+1.9\*10^(-5)\*frequency^(15)));

n\_prime = N\_doublePrime+nitrogen;

end

% 基于逐线法计算水汽比衰减

function n\_prime = calculateLineByLineAttenuationwater(frequency, pressure, waterVaporPressure,temperature, spectralLines)

% 此函数基于逐线法计算复折射率的虚部。

% 输入：

% frequency - 频率，单位为GHz

% pressure - 压力，单位为hPa

% temperature - 温度，单位为K

% spectralLines - 包含谱线数据的结构体

theta = 300 / temperature;

% 初始化N''的求和变量

N\_doublePrime = 0;

% 遍历所有谱线

for j = 1:length(spectralLines.f0)

S = spectralLines.b1(j) \* 10^(-1) \* waterVaporPressure \*theta^(3.5)\* exp(spectralLines.b2(j)\*(1 - theta));

fi = spectralLines.f0(j);

deltaF = spectralLines.b3(j)\*10^(-4)\*(pressure\*theta^(spectralLines.b4(j))+ ...

spectralLines.b5(j)\*waterVaporPressure\*theta^(spectralLines.b6(j)));

deltaF = 0.535\*deltaF+sqrt(0.217\*deltaF^(2)+(2.1316\*10^(-12)\* fi^(2)/theta));

xiu = 0;

F = frequencyLineShapeFactor(frequency, fi, deltaF, xiu);

N\_doublePrime = N\_doublePrime + S \* F;

end

n\_prime = N\_doublePrime;

end

% 计算频率谱线形状因子

function F = frequencyLineShapeFactor(frequency, fi, deltaF, xiu)

% 计算频率谱线形状因子

F = (frequency/fi)\*((deltaF-xiu\*(fi-frequency))/((fi-frequency)^2 + deltaF^2) +...

(deltaF-xiu\*(fi+frequency))/((fi+frequency)^2 + deltaF^2));

end

% 绘制三维曲线

1. **倾斜路径衰减**

**（1）高度固定100km算不同频率衰减**

% 定义常量和参数

frequencies = 0:1:300; % 频率范围从0到300 GHz，步长为1 GHz

RE = 6371; % 平均地球半径，单位为千米

phi1 = 60; % 视在仰角（假设60度）

% 初始化变量

imax = 922;

delta = zeros(imax, 1);

heights = zeros(imax, 1);

pressure = zeros(imax, 1);

temperature = zeros(imax, 1);

waterVaporDensity = zeros(imax, 1);

gamma\_total = zeros(imax, length(frequencies));

beta = zeros(imax, 1);

alpha = zeros(imax, 1);

pathLength = zeros(imax, 1);

attenuation = zeros(1, length(frequencies));

waterVaporPressure = zeros(imax, 1);

n = zeros(imax, 1);

gamma\_oxygen = zeros(imax, length(frequencies));

gamma\_waterVapor = zeros(imax, length(frequencies));

r = zeros(imax, 1);

% 加载氧气和水蒸气的谱线数据

oxygenData = readSpectralLinesFromExcel('oxygen\_spectral\_lines.xlsx');

waterVaporData = readSpectralLinesFromExcel('water\_vapor\_spectral\_lines.xlsx');

% 使用ITU-R P.835建议书中提供的标准大气参数

for i = 1:imax

delta(i) = 0.0001 \* exp((i-1)/100); % 计算第i层的厚度

heights(i) = sum(delta(1:i-1)); % 计算第i层的底部高度

% 根据高度选择合适的公式计算温度、压力和水蒸气密度

[temperature(i), pressure(i), waterVaporDensity(i)] = getAtmosphericParameters(heights(i));

waterVaporPressure(i) = waterVaporDensity(i) \* temperature(i) / 216.7;

pressure(i) = pressure(i) - waterVaporPressure(i);

% 使用ITU-R P.453建议书公式(1)和(2)计算折射率n

n(i) = 1 + (77.6 \* pressure(i) + 72 \* waterVaporPressure(i) + 3.75e5 \* waterVaporPressure(i) / temperature(i)^2) \* 1e-6;

% 使用ITU-R P.676建议书中的方法计算比衰减γ

for j = 1:length(frequencies)

gamma\_oxygen(i, j) = calculateLineByLineAttenuationOxygen(frequencies(j), pressure(i), waterVaporPressure(i), temperature(i), oxygenData);

gamma\_waterVapor(i, j) = calculateLineByLineAttenuationWater(frequencies(j), pressure(i), waterVaporPressure(i), temperature(i), waterVaporData);

gamma\_total(i, j) = 0.1820 \* frequencies(j) \* (gamma\_oxygen(i, j) + gamma\_waterVapor(i, j));

end

% 计算路径长度和角度

r(i) = RE + heights(i);

beta(i) = asind((n(1) \* RE / (n(i)\*r(i))) \* sind(phi1));

pathLength(i) = -r(i) \* cosd(beta(i)) + sqrt(r(i)^2 \* cosd(beta(i))^2 + 2 \* r(i) \* delta(i) + delta(i)^2);

if i > 1

alpha(i-1) = asind(r(i) / (r(i) + delta(i)) \* sind(beta(i)));

end

end

% 计算总的传播损耗

for j = 1:length(frequencies)

for i = 1:imax

attenuation(j) = attenuation(j) + pathLength(i) \* gamma\_total(i, j);

end

end

% 绘制传播损耗图

figure;

plot(frequencies, attenuation, 'LineWidth', 2);

title('从地面到100km处的传播损耗');

xlabel('频率 [GHz]');

ylabel('衰减 [dB]');

grid on;

function [T, P, rho] = getAtmosphericParameters(Z)

if Z < 11

T = 288.15 - 6.5 \* Z;

P = 1013.25 \* (288.15 / (288.15 - 6.5 \* Z))^(-34.1632/6.5);

rho = 7.5 \* exp(-Z / 2);

elseif Z < 20

T = 216.65;

P = 226.3226 \* exp(-34.1632 \* (Z - 11) / 216.65);

rho = 7.5 \* exp(-Z / 2);

elseif Z < 32

T = 216.65 + (Z - 20);

P = 54.74980 \* (216.65 / (216.65 + (Z - 20)))^(34.1632);

rho = 7.5 \* exp(-Z / 2);

elseif Z < 47

T = 228.65 + 2.8 \* (Z - 32);

P = 8.680422 \* (228.65 / (228.65 + 2.8 \* (Z - 32)))^(34.1632/2.8);

rho = 7.5 \* exp(-Z / 2);

elseif Z < 51

T = 270.65;

P = 1.109106 \* exp(-34.1632 \* (Z - 47) / 270.65);

rho = 7.5 \* exp(-Z / 2);

elseif Z < 71

T = 270.65 - 2.8 \* (Z - 51);

P = 0.6694167 \* (270.65 / (270.65 - 2.8 \* (Z - 51)))^(-34.1632/2.8);

rho = 7.5 \* exp(-Z / 2);

elseif Z < 84.852

T = 214.65 - 2.0 \* (Z - 71);

P = 0.03956649 \* (214.65 / (214.65 - 2.0 \* (Z - 71)))^(-34.1632/2.0);

rho = 7.5 \* exp(-Z / 2);

elseif Z < 91

T = 186.8673;

P = exp(95.571899 - 4.011801 \* Z + 6.424731e-2 \* Z^2 - 4.789660e-4 \* Z^3 + 1.340543e-6 \* Z^4);

rho = 7.5 \* exp(-Z / 2);

else

T = 263.1905 - 76.3232 \* (1 - ((Z - 91) / 19.9429)^2);

P = exp(95.571899 - 4.011801 \* Z + 6.424731e-2 \* Z^2 - 4.789660e-4 \* Z^3 + 1.340543e-6 \* Z^4);

rho = 7.5 \* exp(-Z / 2);

end

end

% 基于逐线法计算氧气比衰减

function n\_prime = calculateLineByLineAttenuationOxygen(frequency, pressure, waterVaporPressure, temperature, spectralLines)

% 此函数基于逐线法计算复折射率的虚部。

N\_doublePrime = 0;

theta = 300 / temperature;

% 遍历所有谱线

for j = 1:length(spectralLines.f0)

S = spectralLines.b1(j) \* 10^(-7) \* pressure \* theta^3 \* exp(spectralLines.b2(j) \* (1 - theta));

fi = spectralLines.f0(j);

deltaF = spectralLines.b3(j) \* 10^(-4) \* (pressure \* theta^(0.8 - spectralLines.b4(j)) + 1.1 \* waterVaporPressure \* theta);

deltaF = sqrt(deltaF^2 + 2.25e-6);

xiu = (spectralLines.b5(j) + spectralLines.b6(j) \* theta) \* 10^(-4) \* (pressure + waterVaporPressure) \* theta^0.8;

F = frequencyLineShapeFactor(frequency, fi, deltaF, xiu);

N\_doublePrime = N\_doublePrime + S \* F;

end

d = 5.6e-4 \* (pressure + waterVaporPressure) \* theta^0.8;

nitrogen = frequency \* pressure \* theta^2 \* (6.14e-5 / d \* (1 + (frequency / d)^2) + ...

1.4e-12 \* pressure \* theta^1.5 / (1 + 1.9e-5 \* frequency^15));

n\_prime = N\_doublePrime + nitrogen;

end

% 基于逐线法计算水汽比衰减

function n\_prime = calculateLineByLineAttenuationWater(frequency, pressure, waterVaporPressure, temperature, spectralLines)

% 此函数基于逐线法计算复折射率的虚部。

N\_doublePrime = 0;

theta = 300 / temperature;

% 遍历所有谱线

for j = 1:length(spectralLines.f0)

S = spectralLines.b1(j) \* 10^(-1) \* waterVaporPressure \* theta^3.5 \* exp(spectralLines.b2(j) \* (1 - theta));

fi = spectralLines.f0(j);

deltaF = spectralLines.b3(j) \* 10^(-4) \* (pressure \* theta^(spectralLines.b4(j)) + ...

spectralLines.b5(j) \* waterVaporPressure \* theta^(spectralLines.b6(j)));

deltaF = 0.535 \* deltaF + sqrt(0.217 \* deltaF^2 + 2.1316e-12 \* fi^2 / theta);

xiu = 0;

F = frequencyLineShapeFactor(frequency, fi, deltaF, xiu);

N\_doublePrime = N\_doublePrime + S \* F;

end

n\_prime = N\_doublePrime;

end

% 计算频率谱线形状因子

function F = frequencyLineShapeFactor(frequency, fi, deltaF, xiu)

F = (frequency / fi) \* ((deltaF - xiu \* (fi - frequency)) / ((fi - frequency)^2 + deltaF^2) + ...

(deltaF - xiu \* (fi + frequency)) / ((fi + frequency)^2 + deltaF^2));

end

% 读取Excel文件中的谱线数据

function data = readSpectralLinesFromExcel(filename)

tbl = readtable(filename);

data.f0 = tbl.f0; % 假设第一列为f0

data.b1 = tbl.b1; % 假设第二列为b1

data.b2 = tbl.b2; % 假设第三列为b2

data.b3 = tbl.b3; % 假设第四列为b3

data.b4 = tbl.b4; % 假设第五列为b4

data.b5 = tbl.b5; % 假设第六列为b5

data.b6 = tbl.b6; % 假设第七列为b6

end

1. **固定100GHz算不同高度衰减**

% 定义常量和参数

frequency = 100; % 频率设为100GHz

RE = 6371; % 平均地球半径，单位为千米

phi1 = 60; % 视在仰角（假设60度）

% 初始化变量

imax = 922;

delta = zeros(imax, 1);

heights = zeros(imax, 1);

pressure = zeros(imax, 1);

temperature = zeros(imax, 1);

waterVaporDensity = zeros(imax, 1);

gamma\_total = zeros(imax, 1);

beta = zeros(imax, 1);

alpha = zeros(imax, 1);

pathLength = zeros(imax, 1);

attenuation = zeros(imax, 1);

waterVaporPressure = zeros(imax, 1);

n = zeros(imax, 1);

gamma\_oxygen = zeros(imax, 1);

gamma\_waterVapor = zeros(imax, 1);

r = zeros(imax, 1);

% 加载氧气和水蒸气的谱线数据

oxygenData = readSpectralLinesFromExcel('oxygen\_spectral\_lines.xlsx');

waterVaporData = readSpectralLinesFromExcel('water\_vapor\_spectral\_lines.xlsx');

% 使用ITU-R P.835建议书中提供的标准大气参数

for i = 1:imax

delta(i) = 0.0001 \* exp((i-1)/100); % 计算第i层的厚度

heights(i) = sum(delta(1:i-1)); % 计算第i层的底部高度

% 根据高度选择合适的公式计算温度、压力和水蒸气密度

[temperature(i), pressure(i), waterVaporDensity(i)] = getAtmosphericParameters(heights(i));

waterVaporPressure(i) = waterVaporDensity(i) \* temperature(i) / 216.7;

pressure(i) = pressure(i) - waterVaporPressure(i);

% 使用ITU-R P.453建议书公式(1)和(2)计算折射率n

n(i) = 1 + (77.6 \* pressure(i) + 72 \* waterVaporPressure(i) + 3.75e5 \* waterVaporPressure(i) / temperature(i)^2) \* 1e-6;

% 使用ITU-R P.676建议书中的方法计算比衰减γ

gamma\_oxygen(i) = calculateLineByLineAttenuationOxygen(frequency, pressure(i), waterVaporPressure(i), temperature(i), oxygenData);

gamma\_waterVapor(i) = calculateLineByLineAttenuationWater(frequency, pressure(i), waterVaporPressure(i), temperature(i), waterVaporData);

gamma\_total(i) = 0.1820 \* frequency \* (gamma\_oxygen(i) + gamma\_waterVapor(i));

% 计算路径长度和角度

r(i) = RE + heights(i);

beta(i) = asind((n(1) \* RE / (n(i)\*r(i))) \* sind(phi1));

pathLength(i) = -r(i) \* cosd(beta(i)) + sqrt(r(i)^2 \* cosd(beta(i))^2 + 2 \* r(i) \* delta(i) + delta(i)^2);

if i > 1

alpha(i-1) = asind(r(i) / (r(i) + delta(i)) \* sind(beta(i)));

end

end

% 计算总的传播损耗

for i = 1:imax

attenuation(i) = attenuation(i) + pathLength(i) \* gamma\_total(i);

end

% 绘制传播损耗图

figure;

plot(heights, attenuation, 'LineWidth', 2);

title('100GHz电磁波传播损耗');

xlabel('高度 [km]');

ylabel('衰减 [dB]');

grid on;

function [T, P, rho] = getAtmosphericParameters(Z)

if Z < 11

T = 288.15 - 6.5 \* Z;

P = 1013.25 \* (288.15 / (288.15 - 6.5 \* Z))^(-34.1632/6.5);

rho = 7.5 \* exp(-Z / 2);

elseif Z < 20

T = 216.65;

P = 226.3226 \* exp(-34.1632 \* (Z - 11) / 216.65);

rho = 7.5 \* exp(-Z / 2);

elseif Z < 32

T = 216.65 + (Z - 20);

P = 54.74980 \* (216.65 / (216.65 + (Z - 20)))^(34.1632);

rho = 7.5 \* exp(-Z / 2);

elseif Z < 47

T = 228.65 + 2.8 \* (Z - 32);

P = 8.680422 \* (228.65 / (228.65 + 2.8 \* (Z - 32)))^(34.1632/2.8);

rho = 7.5 \* exp(-Z / 2);

elseif Z < 51

T = 270.65;

P = 1.109106 \* exp(-34.1632 \* (Z - 47) / 270.65);

rho = 7.5 \* exp(-Z / 2);

elseif Z < 71

T = 270.65 - 2.8 \* (Z - 51);

P = 0.6694167 \* (270.65 / (270.65 - 2.8 \* (Z - 51)))^(-34.1632/2.8);

rho = 7.5 \* exp(-Z / 2);

elseif Z < 84.852

T = 214.65 - 2.0 \* (Z - 71);

P = 0.03956649 \* (214.65 / (214.65 - 2.0 \* (Z - 71)))^(-34.1632/2.0);

rho = 7.5 \* exp(-Z / 2);

elseif Z < 91

T = 186.8673;

P = exp(95.571899 - 4.011801 \* Z + 6.424731e-2 \* Z^2 - 4.789660e-4 \* Z^3 + 1.340543e-6 \* Z^4);

rho = 7.5 \* exp(-Z / 2);

else

T = 263.1905 - 76.3232 \* (1 - ((Z - 91) / 19.9429)^2);

P = exp(95.571899 - 4.011801 \* Z + 6.424731e-2 \* Z^2 - 4.789660e-4 \* Z^3 + 1.340543e-6 \* Z^4);

rho = 7.5 \* exp(-Z / 2);

end

end

% 基于逐线法计算氧气比衰减

function n\_prime = calculateLineByLineAttenuationOxygen(frequency, pressure, waterVaporPressure, temperature, spectralLines)

% 此函数基于逐线法计算复折射率的虚部。

N\_doublePrime = 0;

theta = 300 / temperature;

% 遍历所有谱线

for j = 1:length(spectralLines.f0)

S = spectralLines.b1(j) \* 10^(-7) \* pressure \* theta^3 \* exp(spectralLines.b2(j) \* (1 - theta));

fi = spectralLines.f0(j);

deltaF = spectralLines.b3(j) \* 10^(-4) \* (pressure \* theta^(0.8 - spectralLines.b4(j)) + 1.1 \* waterVaporPressure \* theta);

deltaF = sqrt(deltaF^2 + 2.25e-6);

xiu = (spectralLines.b5(j) + spectralLines.b6(j) \* theta) \* 10^(-4) \* (pressure + waterVaporPressure) \* theta^0.8;

F = frequencyLineShapeFactor(frequency, fi, deltaF, xiu);

N\_doublePrime = N\_doublePrime + S \* F;

end

d = 5.6e-4 \* (pressure + waterVaporPressure) \* theta^0.8;

nitrogen = frequency \* pressure \* theta^2 \* (6.14e-5 / d \* (1 + (frequency / d)^2) + ...

1.4e-12 \* pressure \* theta^1.5 / (1 + 1.9e-5 \* frequency^15));

n\_prime = N\_doublePrime + nitrogen;

end

% 基于逐线法计算水汽比衰减

function n\_prime = calculateLineByLineAttenuationWater(frequency, pressure, waterVaporPressure, temperature, spectralLines)

% 此函数基于逐线法计算复折射率的虚部。

N\_doublePrime = 0;

theta = 300 / temperature;

% 遍历所有谱线

for j = 1:length(spectralLines.f0)

S = spectralLines.b1(j) \* 10^(-1) \* waterVaporPressure \* theta^3.5 \* exp(spectralLines.b2(j) \* (1 - theta));

fi = spectralLines.f0(j);

deltaF = spectralLines.b3(j) \* 10^(-4) \* (pressure \* theta^(spectralLines.b4(j)) + ...

spectralLines.b5(j) \* waterVaporPressure \* theta^(spectralLines.b6(j)));

deltaF = 0.535 \* deltaF + sqrt(0.217 \* deltaF^2 + 2.1316e-12 \* fi^2 / theta);

xiu = 0;

F = frequencyLineShapeFactor(frequency, fi, deltaF, xiu);

N\_doublePrime = N\_doublePrime + S \* F;

end

n\_prime = N\_doublePrime;

end

% 计算频率谱线形状因子

function F = frequencyLineShapeFactor(frequency, fi, deltaF, xiu)

F = (frequency / fi) \* ((deltaF - xiu \* (fi - frequency)) / ((fi - frequency)^2 + deltaF^2) + ...

(deltaF - xiu \* (fi + frequency)) / ((fi + frequency)^2 + deltaF^2));

end

% 读取Excel文件中的谱线数据

function data = readSpectralLinesFromExcel(filename)

tbl = readtable(filename);

data.f0 = tbl.f0; % 假设第一列为f0

data.b1 = tbl.b1; % 假设第二列为b1

data.b2 = tbl.b2; % 假设第三列为b2

data.b3 = tbl.b3; % 假设第四列为b3

data.b4 = tbl.b4; % 假设第五列为b4

data.b5 = tbl.b5; % 假设第六列为b5

data.b6 = tbl.b6; % 假设第七列为b6

end

**4、色散**

frequencies = 0:1:300; % 频率范围从0到300 GHz，步长为1 GHz

pressure = 1013.25-waterVaporPressure; % 干空气气压，单位为hPa

temperature = 15 + 273.15; % 温度，单位为开尔文（15摄氏度）

waterVaporDensity = 7.5; % 水蒸气密度，单位为g/m^3

waterVaporPressure = waterVaporDensity \* temperature / 216.7; % 水蒸气分压

xiu = 0;

% 加载氧气和水蒸气的谱线数据

oxygenData = readSpectralLinesFromExcel('oxygen\_spectral\_lines.xlsx');

waterVaporData = readSpectralLinesFromExcel('water\_vapor\_spectral\_lines.xlsx');

% 使用逐线法计算特定频率下的比衰减

gamma\_oxygen = zeros(size(frequencies));

gamma\_waterVapor = zeros(size(frequencies));

for i = 1:length(frequencies)

gamma\_oxygen(i) = calculateLineByLineAttenuationoxygen(frequencies(i), pressure, waterVaporPressure, temperature, oxygenData);

gamma\_waterVapor(i) = calculateLineByLineAttenuationwater(frequencies(i), pressure, waterVaporPressure, temperature, waterVaporData);

end

% 总的比衰减

gamma\_total = -1.2008 .\* frequencies .\* (gamma\_oxygen + gamma\_waterVapor);

% 绘制比衰减图

figure;

plot(frequencies, gamma\_total, 'LineWidth', 2);

title('使用逐线法计算的色散');

xlabel('频率 [GHz]');

ylabel('色散 [deg/km]');

grid on;

% 读取Excel文件中的谱线数据

function data = readSpectralLinesFromExcel(filename)

% 从指定的Excel文件读取谱线数据

% 输入：

% filename - Excel文件名

% 读取Excel文件中的数据，默认读取第一个工作表，并确保跳过标题行

tbl = readtable(filename);

% 将表格数据转换为结构体

data.f0 = tbl.f0; % 假设第一列为f0

data.b1 = tbl.b1; % 假设第二列为b1

data.b2 = tbl.b2; % 假设第三列为b2

data.b3 = tbl.b3; % 假设第四列为b3

data.b4 = tbl.b4; % 假设第五列为b4

data.b5 = tbl.b5; % 假设第六列为b5

data.b6 = tbl.b6; % 假设第七列为b6

end

% 基于逐线法计算氧气比衰减

function n\_prime = calculateLineByLineAttenuationoxygen(frequency, pressure, waterVaporPressure, temperature, spectralLines)

% 此函数基于逐线法计算复折射率的虚部。

% 输入：

% frequency - 频率，单位为GHz

% pressure - 压力，单位为hPa

% temperature - 温度，单位为K

% spectralLines - 包含谱线数据的结构体

theta = 300 / temperature;

% 初始化N''的求和变量

N\_doublePrime = 0;

% 遍历所有谱线

for j = 1:length(spectralLines.f0)

S = spectralLines.b1(j) \* 10^(-7) \* pressure \*theta^(3)\* exp(spectralLines.b2(j)\*(1 - theta));

fi = spectralLines.f0(j);

deltaF = spectralLines.b3(j)\*10^(-4)\*(pressure\*theta^(0.8-spectralLines.b4(j))+1.1\*waterVaporPressure\*theta);

deltaF = sqrt(deltaF^(2)+2.25\*10^(-6));

xiu = (spectralLines.b5(j)+spectralLines.b6(j)\*theta)\*10^(-4)\*(pressure+waterVaporPressure)\*theta^(0.8);

F = frequencyLineShapeFactor(frequency, fi, deltaF,xiu);

N\_doublePrime = N\_doublePrime + S \* F;

end

d =5.6\*10^(-4)\*(pressure+waterVaporPressure)\*theta^(0.8);

nitrogen = (-6.14\*10^(-5)\*pressure\*theta^(2)\*frequency^(2))/(frequency^(2)+d^(2));

n\_prime = N\_doublePrime+nitrogen;

end

% 基于逐线法计算水汽比衰减

function n\_prime = calculateLineByLineAttenuationwater(frequency, pressure, waterVaporPressure,temperature, spectralLines)

% 此函数基于逐线法计算复折射率的虚部。

% 输入：

% frequency - 频率，单位为GHz

% pressure - 压力，单位为hPa

% temperature - 温度，单位为K

% spectralLines - 包含谱线数据的结构体

theta = 300 / temperature;

% 初始化N''的求和变量

N\_doublePrime = 0;

% 遍历所有谱线

for j = 1:length(spectralLines.f0)

S = spectralLines.b1(j) \* 10^(-1) \* waterVaporPressure \*theta^(3.5)\* exp(spectralLines.b2(j)\*(1 - theta));

fi = spectralLines.f0(j);

deltaF = spectralLines.b3(j)\*10^(-4)\*(pressure\*theta^(spectralLines.b4(j))+ ...

spectralLines.b5(j)\*waterVaporPressure\*theta^(spectralLines.b6(j)));

deltaF = 0.535\*deltaF+sqrt(0.217\*deltaF^(2)+(2.1316\*10^(-12)\* fi^(2)/theta));

xiu = 0;

F = frequencyLineShapeFactor(frequency, fi, deltaF, xiu);

N\_doublePrime = N\_doublePrime + S \* F;

end

n\_prime = N\_doublePrime;

end

% 计算频率谱线形状因子

function F = frequencyLineShapeFactor(frequency, fi, deltaF, xiu)

% 计算频率谱线形状因子

F = (frequency/fi)\*((deltaF\*xiu+(fi-frequency))/((fi-frequency)^2 + deltaF^2) -...

(deltaF\*xiu+(fi+frequency))/((fi+frequency)^2 + deltaF^2));

end

逐线法计算需要的参数表格如下：

