EG4217/EG7217 Spacecraft Communications

Coursework 1

Introduction

The coursework assessment is intended to give students the opportunity to put the material in this module into practice and to gain a critical understanding of the models presented in the lectures. It is to be done in your own time but can be discussed in the Friday sessions.

You should download the Matlab programmes and run them to answer the questions. The results and conclusions arising from these exercises should be written in an electronic lab-book (MS Word or similar), along with any graphical output from the programmes, as required by the questions. If possible, please convert the finished lab-book to pdf format before uploading it to the EG7217 Blackboard site. **This is not a formal report**, it is a lab-book, and should therefore explain to an educated reader (e.g. one of your peers) exactly what you have modelled, what it showed and what you understood from the results, but no further explanation of the modelled phenomena is required and lengthy explanations will attract no further marks. Furthermore, **Matlab code should not be included** in the lab-book. The results should be clearly laid out and all variables should be defined. The results should be in sentence form, rather than just cut and pasted from the Matlab window. For example, the following is inadequate:

*'>> d=sqrt(2*a*ht)*
d =
99.5359'

and should be replaced by:

*'The horizon distance was calculated from the formula d=sqrt(2*a*ht), where a is the earth radius in km and h_t is the antenna height in km. It was found to be 99.5 km.'*

Each question is worth 2.5 marks, except for Q3, which is worth 5 marks. One mark will be awarded if there is evidence that the relevant programme has been run; two marks will be awarded if the answer is approximately correct and 2.5 marks are available if the answer is correct and well presented (clear text, axis labels etc.). These are doubled for Q3. The overall mark for the lab-book contributes 25% of the total marks available for the module.

You must upload the document by **12.00 noon on Monday 11th March 2024**. Late submissions will lose 10% of the total possible mark for the first 24 hours or part thereof, and an additional 5% for each subsequent 24 hour period or part thereof.

You are reminded that while the outputs of the software will be the same for all students, the discussions and conclusions must be **your own work**. All work that is not your own MUST be referenced in your logbook. When you submit your work you **must** include the following paragraph at the beginning of your lab-book:

"I declare that this assignment is my own work, that sources of reference are acknowledged and that it has not been submitted for any other course. I understand that plagiarism is a serious offence under the University's regulations and that appropriate penalties will be applied if I am found to have submitted plagiarised work."

How to run the software

The Matlab programs do not need to be altered or edited. To run them, first download them and make sure that Matlab's current directory is where they are kept. Then, put your cursor in the part of the Matlab window marked "Command Window" You should see the prompt ">>". Type the name of the program, followed by round brackets and the value of any input parameters (as numbers or defined variables), separated by commas. If you're unsure what the input parameters are, then double-click on the programme in "Current Folder" and the programme's code will appear in the part of the window marked "Editor". The first line of code will tell you the required parameters and their correct order. For Q8 and Q9 you'll need to ensure that you've installed the Matlab phased array toolbox, as arrayfactor.m and steervec.m are inbuilt functions.

For example, to run freespace.m type `freespace(1000,50)`

This will return the calculated value of free-space loss for a 1000 MHz radio wave that travels 50km. Some additional Matlab code may need to be written to present the results in the most appropriate format. Many of the Matlab functions accept one of the input parameters in vector form, allowing the output parameter to be plotted in one line, e.g.

```
plot([1:20],freespace(1000,[1:20]))
```

will plot out the freespace loss from 1 to 20km in 1km steps for a 1GHz signal. A slower, but a more C- or Fortran-like, alternative is

```
for range=1:20
loss(range)=freespace(1000,range);
end
plot([1:20],loss)
```

All of the functions are sufficiently well documented by comments to explain what they do and what inputs are required. You should have a reasonable grasp what each model does before running the code from revising the lectures. Please be careful to look at the units of each input variable and use appropriate values. A few extra Matlab commands may be useful (look up the details using the online help, e.g. at the command line type help subplot):

- hold on : enables multiple lines to be drawn on the same axes
- subplot : can be used to generate multiple axes on the same piece of paper
- title, xlabel and ylabel : add the appropriate axis labels and title to your graph
- print –dtiff <filename.tif> : saves the current figure in tiff format for importing into Word.
- plot : see the help for details of how to add plot symbols, introduce dashed lines, etc.

Questions

Q1

freespace.m

This programme calculates free space loss assuming isotropic antennas.

Plot the free-space loss for ranges between 100 km and 1000 km for frequencies 1 GHz and 2 GHz.

To make this curve easy for the reader to view and understand:
Plot these curves in different colours on the same graph using 'hold on'.
Add a graticule using the command 'grid on'
Add a legend using the command 'legend'.
Add axis labels using 'xlabel' and 'ylabel', not forgetting units.
Add a figure number and caption to the plot
Add some text, referring to Figure 1 and describing the trend.
Explain why loss increases with distance.
(Subsequent plots should also all have these embellishments to make them clear.)

Q2

visible_time.m, freespace.m and **sat_dist.m**

visible_time.m calculates the time in seconds for which a satellite is visible from the earth
sat_dist calculates the distance from a satellite to the ground-station.

An Earth observation satellite at an altitude of 400 km collects data at a rate of 10 kbit/s as it goes round the Earth. It must download this data using a 5 GHz carrier wave as it flies over its mission control centre. An adequate SNR is only possible the when the satellite is above 10^{o} elevation, as seen from the mission control centre. Assume that it flies directly over the mission control centre.

a) Calculate the approximate free-space loss in the satellite-mission control centre link when data collection starts. Don't forget the units.
b) Calculate the data rate required to ensure that all the data from the latest orbit can be downloaded as the satellite passes over the mission control centre.

Q3

ITUdryair.m, sat_dist.m and **freespace.m**

The programme ITUdryair.m calculates the loss due to absorption by oxygen.

A satellite orbits at a height of 500 km above the earth. It transmits a signal with an EIRP of 100W at a carrier frequency of 50 GHz to a ground-station at sea-level has a 90 cm radio dish. The threshold value for the received signal is –120 dBW.The air pressure measured at the ground-station is 1013 hPa and at 10 km height, it is measured as 164 hPa. The temperature at the ground-station is 20^{o}C.

a. Calculate the scale height for oxygen pressure.
b. Use ITUdryair.m and this scale height to calculate the oxygen attenuation when the satellite is overhead.
c. Divide this answer by the sine of the elevation angles 10 to 90 degrees to create an array of the loss as a function the elevation angle. Plot the oxygen loss as a function of angle.
d. Calculate the receive antenna gain in dBi.
e. Calculate the distance of the satellite as a function of these same elevation angles and plot the free-space loss as a function of angle.
f. Add the free-space loss to the oxygen loss to get the total loss as a function the elevation angle. Plot this on a separate graph as a function of elevation angle.
g. Use the link budget to calculate the maximum total loss that the link can sustain and use the graph in part (f) to determine the minimum elevation angle at which the signal from the satellite can be received.

Q4

SAMrain.m

This function calculates the loss associated with rain; the former for 0.01% of the time.

A geostationary satellite has an elevation of 30^{o} from Leicester, which has an altitude of 62 m above mean sea-level and it has approximately the same elevation angle as seen from a few miles away in Buxton, Derbyshire, which has an altitude of 330 m. Use the figure opposite (taken from ITU-R P837 version 4) and the approximate latitude of 53^{o}N for both stations and calculate how much better or worse the signal would be in Buxton, compared to Leicester, for the worst 52 minutes of a typical year. The satellite's carrier signal has a frequency of 20 GHz.

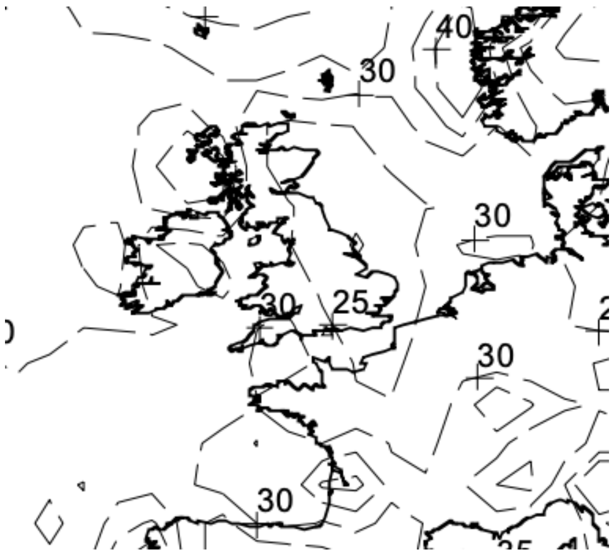

Fig 1Rain rate in mm/hr for the UK

Q5

SAMrain.m and **ITUrainattenuation.m**

ITUrainattenuation.m uses the ITU model to calculate rain loss for various percentages of the time.

The rain rate is 90 mm/hour for the rainiest 0.01% of the time in Kolkata (Calcutta), India. A geostationary TV satellite with the same longitude as Kolkata has an elevation of 64^{o}, as seen from Kolkata, which is at sea-level. A 30 GHz signal is received by a large antenna and distributed to subscribers. In dry weather, the signal-to-noise ratio is 12 dB, which is sufficient to ensure a good TV signal. However, the operators are worried that if it goes below 7 dB, viewers will complain.

a. For how many hours per year would you expect to see this level of degradation? *(To help you answer this question, use the inbuilt Matlab command p=logspace(-1,-3,10) to create a range of percentages and then use the loglog command to plot rain attenuation as a function of p on a log-log graph.)*
b. What other effect (that is not taken into consideration in the Matlab programme) might make this value inaccurate?

Q6

ITUxpd.m

This programme implements the model in ITU Recommendation P618 to determine the cross-polar discrimination seen when signals are sent simultaneously on different polarisations of the same wave.

A 12 GHz radio wave with left-handed circular polarisation is beamed from ground-station to a satellite at 30^{O} elevation. When the weather at the ground station is fine, the satellite receives the signal with a signal-to-noise ratio of 25 dB, but in the worst 0.01% of times, the SNR reduces to 20 dB. To increase throughput, the satellite operator decides to duplex two signals together; one with left-handed circular polarisation and one with right-handed circular polarisation.

a. What will the value of the cross-polar discrimination for these signals?
b. What will be the SINR (the signal-to-(interferer plus noise)-ratio) for this signal for 0.01% of the time?

Q7

SAMrain.m and **ITUdiversitygain.m**

ITUdiversitygain.m implements the extra gain available from having two Earth stations, according to the model in ITU Recommendation P618.

A firm provides internet services using a geostationary satellite situated at 1 degree W. It has a ground station in Reading, England (altitude 61 m above mean sea-level) but experiences rain attenuation of 13 dB in the worst hour of the year. The company is considering installing a second ground station either in Bristol (113 km west of Reading) or in Leicester (132 km North of Reading). Use these programmes to estimate which of these gives the better diversity gain, showing your working.

Q8

arrayfactor.m

This is an inbuilt function that can be found in the Matlab phased array toolbox. It produces the antenna pattern for a phased array of antennas.

Use the Matlab to determine the full 3dB beamwidth of the main lobe for an array of eight dipoles, spaced half a wavelength apart. Show the antenna pattern (power vs direction) in a plot.

Typing 'help arrayfactor' should tell you that the inputs to this function are:
position:a 1-D array describing the positions of the antenna elements **in units of the wavelength**,
angle: the angles in degrees to scan over (= theta in the lectures) – try -90 to + 90 to begin with and
phases (= weightings) an array of all ones will give zero phase difference between elements, to produce a beam at 0 degrees.

Q9

arrayfactor.m and **steervec.m**

The inbuilt routine, steervec will give the relative phases of the elements needed to steer the beam in the wanted direction.

The relative phases of the elements were all ones in the last question, giving no phase difference, but here they will be created by steervec.m. Its inputs are the position array (as used for arrayfactor) and the intended angle of the main beam in degrees. Use these functions to create an array for use with a 3 GHz carrier with dipoles 0.1 m apart. Show the beam pattern and find:

a) the successive phase in degrees of each element with respect to its preceding element required to give the correct beam orientation,
b) the power at the peak of the main beam relative to the power at the peak of the nearest side-lobe.

End of questions