# PRINCIPLES OF COMPUTER NETWORKS

COMP 3203

Evangelos Kranakis
Assignment 2: **Programming**

September 17, 2011

# 1 Directional Antennae Connectivity

This project concerns connectivity of wireless sensor networks using omnidirectional and directional antennae.

## 1.1 Motivation and Problem Definition

Although connectivity in wireless sensor networks can be established using either omnidirectional or directional antennae, the latter can be more efficient and transmit further in a given direction for the same amount of energy than the former. We are interested in the following question: what is the "connectivity" cost when replacing omnidirectional with directional antennae? More specifically we are interested in the following problem:

> Consider a set $S$ of $n$ points in the plane that can be identified with sensors having a range $r > 0$. For a given angle $0 \leq \varphi \leq 2\pi$ each sensor is allowed to use at most one directional antennae of angle at most $\varphi$. How do we rotate the antennae and at what range $r$ (same for all sensors) so that by a directed, strongly connected network on $S$ is formed.

## 1.2 Problem to be considered in this project

Simulate the algorithm that constructs the strongly connected network of directional antennae and provides a solution to this problem. As a source, use the publication "Maintaining Connectivity in Sensor Networks Using Directional Antennae." by E. Kranakis, D. Krizanc, O. Morales. Chapter 4, pages 59-84. In Theoretical Aspects of Distributed Computing in Sensor Networks, S. Nikoletseas and J. Rolim, editors, Springer, ISBN 978-3-642-14848-4. You can download this publication from the course web page. (You are free to use other publications that you may find in my web page or elsewhere in the internet.)

# 2 Simulation Details

In due time, following the class lectures on wireless networks and location awareness, I will give a lecture in class on this topic so as to explain terms and consepts. However, your simulations should try to address the following issues detailed in three parts below.

## 2.1 Part 1

Starting from a set of pints in the plane, you will need to implement the following core algorithms.

1. Minimum Spanning Tree (MST).
2. Maximal Matching of an MST.
3. Antenna Orientation Algorithm.

## 2.2 Part 2

Sensors should have identical range.

1. Choose initial positions of sensors.
2. Draw the omnidirectional network.
3. Draw the strongly connected directional network.
4. Vary circle radius (i.e. $10, 100, 1000$)
5. Vary the number of sensors (i.e. $10, 20, 50, 100, \ldots$)
6. Vary the transmission range of sensors (i.e. $10, 20, 50, 100$).

## 2.3 Part 3

Compare the performance (hop-count) of

1. Shortest paths
2. Length of Routes
3. Diameter of network
4. Draw statistical averages for various ranges and angles.

in the omnidirectional and directional sensor networks by varying the angle and range.

## 3 Documentation and Code

It is a good programming practice to assume you are developing this application as a member of a team and therefore you should ensure that it will take minimum amount of time for others to understand the code and the overall functionality of the application. Documentation should include:

1. class diagram,
2. implementation details (i.e., how were the specific elements implemented - frames, queues, send and receive routines, protocol, etc. and which classes were used),
3. log file (to permit students/TAs to assess the functionality of the protocol),
4. test cases (black box and white box - stating the objective, results expected, conditions, assumptions and if possible, screen shots).
5. Your programming code should be written in Java or C++.

## 4 What to submit

By the specified deadline submit:

1. A detailed document describing operational details of the protocols as well as details of your implementation. Number of pages can vary, but about twenty pages should be sufficient. Create a PS or PDF document of your write-up.
2. Instructions and batch files to compile and run the application.
3. Groups should email the TA a softcopy of all their work: code, tests, logs, documentation in a single compressed file (.zip, .tar.gz) by the due date.
4. In addition, a CD-ROM with your source code as well as a paper copy of the document should be put in an envelope with your name, and student identity in the location specified in the course web page.

Perform the programming on your own and acknowledge your sources in case you used library material.

## 5 Grading

You are permitted to vary in your definition of the model, data structures, and documentation. However, your grade will be based on the following

1. Correctness of code supplied (functionality, test cases, log file(s)).
2. Quality of document and code (different sections, i.e. class diagram, adequate documentation of code).
3. User interface (sufficiently demonstrate the functionality of the protocol).

Write your project neatly and carefully. Be precise, concise and clear!

# 6 Form a Team (soon!)

1. For this assignment you can work **in a team of up to three**.
2. Select your collaborator(s) on your own.
3. Arrange all necessary deadlines with your team mates.
4. You are responsible for smooth functioning of team work.
5. There will be a single grade for the team.
6. You do not get extra points for forming a smaller team.