PH20018: Coursework Assignment 2
2020-21

There are 3 questions in this assignment, which is worth 50% of your unit mark. For each question, you will write a program and generate some results. You will write a report that contains a printout of each program, the results you obtained, and a short discussion of the results. You should explain how you designed your program and how it works, and demonstrate that it is robust. There are no specific rules about report format, but please aim to present your work clearly and logically.

Your coursework should be in the form of **a word-processed report** and submitted either **in PDF format or word document**. **The electronic copy of the report, plus text files containing your C programs** should be submitted via the online submission box below **by 30-April**.

Coursework will be submitted and marked anonymously.

Conductivity of disordered materials

Computers are increasingly used to understand the physical processes that take place in complicated (and imperfect) systems in everyday life. For example, what happens if we build a disordered material made of small conducting particles, embedded in an insulating plastic sheet. Does the material conduct electricity? Can we mix the flexibility and low cost of plastic with the electrical properties we need for displays or solar cells?

Figure 1(a) is a schematic picture of a disordered material, while and figure 1(b) shows a simplified representation of the system, suitable for computational modelling. In this case, you can see the particles form a path by which current can flow across the system. Figure 1(c) shows how the computer actually represents the system. This assignment concerns this simple model of a disordered material.

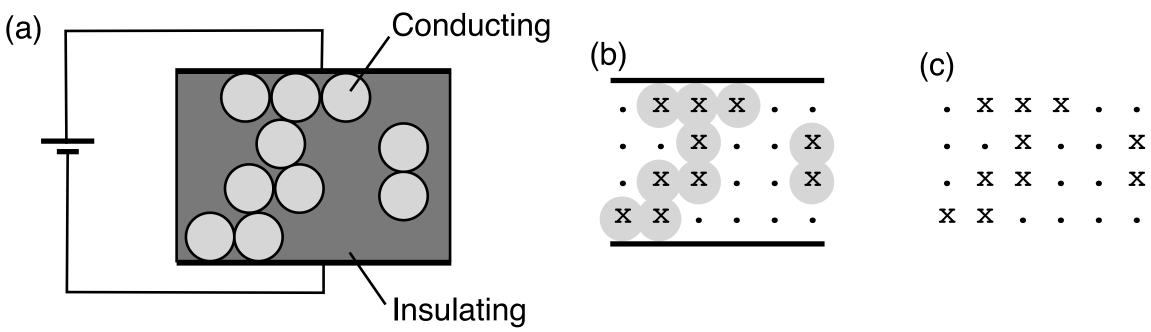

Fig. 1. (a) A disordered material consisting of conducting particles in an insulating plastic environment. (b) Simplified picture, where particles are represented by x symbols that lie on a square grid. (c) Computational representation of the grid alone. Neighbouring x symbols indicate particles that are touching, so current can flow between them.

Question 1 (20 marks)

One feature of disordered materials is that making them is a random process, and each one comes out slightly different. Describing such processes on a computer requires 'random numbers'. The Appendix gives you some functions that help with this.

a) Write a program that generates and prints grids like the one in Fig. 1(c), of size $L_x \times L_y$, with N particles randomly distributed through the grid. The program should allow the user to enter L_x and L_y and N from the keyboard, and each site on the grid can have at most one particle. (Each site has either one particle or no particle, as in Fig. 1(c). In that figure, $L_x = 6$, $L_y = 4$ and $N = 10$.). Use your program to generate 3 different grids, all with the same L_x, L_y and N.

Note: In 'random grids', every position in the grid should be equally likely to have a particle. If your program does not achieve this then it is not working properly.

b) Imagine now that the conductive properties of some fraction of the particles in your grid can be hightened such that conduction can occur through diagonal as well as adjacent particles. For example, in the grid shown below, those particles drawn with a '`+`' are only conductive along vertical or horizontal paths, whereas those particles drawn with a '`x`' are conductive along vertical, horizontal and diagonal directions.

```
.  +  x  +  .  .
.  .  +  .  .  +
.  +  x  .  .  +
+  +  .  .  .  .
```

Modify your program to allow the user to input the fraction of particles, f_{SC}, that are super-conductors (i.e. of `x` type). Print a grid of fixed size $L_x \times L_y$, with N randomly distributed particles, but with the fraction of super-conducting particles set to f_{SC}=10%, 20% and 30% of the total number of particles.

Note: You may want to consider using structs to define whether your particles are super-conductors, normal conductors, or insulators.

Question 2 (40 marks)

An important question for these grids is whether paths exist that connect the two electrodes shown in Fig. 1. (For example, this determines whether the sample you have made acts as a resistor or a capacitor.) To test this, one must consider clusters of connected particles.

a) Develop further the program that you wrote for question 1 so that, after generating a grid, the code selects an initial particle in the grid at random, and then identifies the corresponding cluster of connected particles. Generate and print 3 different grids, all with the same L_x, L_y and N, and with $f_{SC} = 10\%$, making it clear in the printed output which was the initial particle, and which particles do and do not make up part of the cluster. For example, you could identify cluster particles with a '1' and non-cluster particles with a '-1', and set the initial particle fo '0'.

Notes: When finding particles that are adjacent to some other particle, you need to take care with the edges of your system.

b) Now modify your program to work out whether the cluster you have found contains a path between the two electrodes. Use your program to analyse 100 grids with $L_x = 6$, $L_y = 4$, $N = 10$ and $f_{SC} = 10\%$, and output the number (or fraction) of grids for which you find a connecting path.

c) Try increasing the size of your grid to $L_x = 100$, $L_y = 100$, with $N = 6000$ and $f_{SC} = 10\%$, again running this 100 times and outputting the fraction of grids with a connecting path. If it takes longer than 5 minutes to run 100 grids, try analysing a smaller number of grids and calculate how long you would need to analyse 100 of them.

Investigate the behaviour of grids of size 100×100, for different numbers of particles N and different fractions of super-conducting particles. What trends to you observe for the fraction of grids for which a path exists between the electrodes? Why?

Question 3 (40 marks)

Imagine now that you have a 3D material of size $L_x \times L_y \times L_z$ where $L_x = L_y = L_z = 10$. How large does N have to be (if $f_{SC} = 10\%$) in order to find a connecting path 50% of the time that you run your code? How does this vary if you increase the size of your 3D material, and the fraction of super-conducting particles, f_{SC}?

Appendix: function for generating random numbers

The following program generates a sequence of 20 random integers and prints them to the screen. The integers are in the range 0...9.

```
#include <stdio.h>

// declaration of a function that returns a random integer
// in the range 0...(max-1)
int random_i(int max);

main() {
  int i;
  int nmax=10, nn=20;
  int seed=3;

  // setup the random number generator
  srandom(seed);

  for (i=0;i<nn;i++) {
    printf("%d\n",random_i(nmax));
  }
}

// function definition
int random_i(int max) {
  // note random() is a built-in C function
  // the % operator divides by max and takes the remainder
  return (random() % max);
}
```

This program is available on moodle. You probably want to use similar functions to these in your answer to the questions in this assignment.

Things to notice:

- The program prints an apparently random sequence. But if you run it several times, it always prints the *same* random sequence. This is called ‘pseudo-randomness’.
- If you want to generate different sequences, you should change the argument `(seed)` of `srandom` from 3 to some other integer.
- The `C` function `random()` returns a random integer between 0 and $(2^{31} - 1)$. Dividing by 10 and taking the remainder gives a random integer between 0 and 9.
- In most `C` programs, you should call the function `srandom` at most once. If you are tempted to call this function several times in the same program, you should ask yourself if this is really necessary.