

# Modelling Electrical Conductivity in Cluster Networks

Emmanuel Ohieku Jonah (emmanuelj@aims.ac.za)  
African Institute for Mathematical Sciences (AIMS)

Supervised by M. Härting and D.T. Britton  
University of CapeTown

June 8, 2007

# Abstract

The addition of silicon into the active layer of thick-film circuits is currently an area of research interest. This work was done to study the contributions of silicon nanoparticles which form clusters of resistors in the active layer of thick-film circuits. A computer model was developed to compute the effective resistance between any two points in the clusters. Four resistor networks were studied which are possible natural configurations of the silicon particles. The results were compared with measured values from electrical circuits that were designed and constructed.

# Contents

<b>Abstract</b>	<b>i</b>
<b>1 Introduction</b>	<b>1</b>
<b>2 Clusters, Kirchhoff's Laws and Conductance</b>	<b>3</b>
2.1 Clusters . . . . .	3
2.1.1 Tetrahedral Cluster . . . . .	3
2.1.2 Hexahedral Cluster . . . . .	3
2.1.3 Octahedral Cluster . . . . .	6
2.1.4 Icosahedral Cluster . . . . .	6
2.2 Kirchhoff's Laws . . . . .	8
2.2.1 Example to Compute $R_{\text{eff}}$ . . . . .	8
2.3 Conductance . . . . .	9
<b>3 Computer Model and Electrical Circuits</b>	<b>11</b>
3.1 Computer Model . . . . .	11
3.1.1 First Method . . . . .	11
3.1.2 Second Method . . . . .	14
3.2 Electrical Circuits . . . . .	18
<b>4 Results and Discussions</b>	<b>20</b>
<b>5 Conclusion</b>	<b>25</b>
<b>A The Codes for the Models</b>	<b>26</b>
A.1 First Method . . . . .	26
A.2 Second Method . . . . .	30
A.3 Icosahedral Connections . . . . .	31
<b>References</b>	<b>34</b>

# List of Figures

2.1	Tetrahedral Cluster connected with equivalent resistance. . . . .	4
2.2	Hexahedral Cluster connected with equivalent resistance. . . . .	5
2.3	Octahedral Cluster connected with equivalent resistance. . . . .	6
2.4	Icosahedral Cluster connected with equivalent resistance. . . . .	7
2.5	Equivalent circuit of Fig. 2.1(b) . . . . .	9
3.1	A sample network . . . . .	15
3.2	Resistor connections of the tetrahedral, hexahedral and octahedral cluster networks . .	19
3.3	Resistor connections of the icosahedral cluster . . . . .	19
4.1	Tetrahedral network with the voltages at each node. . . . .	21
4.2	Hexahedral network with the voltages at each node. . . . .	22
4.3	Octahedral network with the voltages at each node. . . . .	23
4.4	Icosahedral network with the voltages at each node. . . . .	24

# 1. Introduction

There have been growing demands in the field of electronics for the reduction of the size and cost of electronic components. This has led engineers and scientists to diversify their research into devising materials that could be used to meet this demand. Much attention has now been focussed on research into the use of nanotechnology to achieve this goal [1].

The first to use the term “Nanotechnology”, Professor Norio Taniguchi of Tokyo Science University [2] defined it as “Techniques which mainly consists of the processing of, separation, consolidation, and deformation of materials by one atom or one molecule” [3]. This technology focuses on structures which are in the range 1 – 100nm, which is where most nanostructures demonstrate their greatest potential [1].

Questions have been raised over why much interest should be focussed on nanostructures. Landman and Luedtke [4] highlighted the fact that being small is different. They stressed that at small scales, the properties of materials no longer obey the laws that govern the bulk materials, indicating that a new level of science could evolve from this emerging field of technology. It has been observed that some of the properties of bulk materials like the colour or state (i.e solid or liquid), could be obtained by combining aggregates of their nanoparticles [5]. Also, scientists and engineers have found it possible to alter the properties of a material by adding particles which create a bulk material with the desired characteristics [6].

Nanomaterials made of metals, semiconductors or oxides have generated interest because of their electrical, optical and chemical properties. Semiconductor composites such as silicon, have been used in the active layers of printed electronic circuits [7]. They have been applied in thick-film printed circuits where the silicon particles form clusters in the semiconducting layer.

Thick-film technology is an additive process that uses screen printing methods to deposit conductive, resistive and insulating films on a circuit board. The screen printing inks deposited on the layers of the circuits are viscous pastes [8] which contains the functional material in an organic vehicle. When the first hybrid circuits were made using this technology, they were observed to be about three to ten times smaller compared to traditional circuit boards [9]. The electrical and thermal properties of these circuits are a lot better than those of traditional printed circuits [10]; this means that their ability to conduct heat is better. This (thick-film) technology is used as an efficient and cost-effective alternative to other methods of patterning passive electronic components and circuitry [7].

Silver has often been used as a resistor in active layers of thick-film printed circuits [9]. They are good conductors, less expensive than equivalent materials like gold, and some other properties that they display have made them a good choice in printed circuits. Hydrogenated amorphous silicon has also been used as the semiconducting layer in hybrid thick-film circuits but nanocrystalline silicon is seen as a potential replacement in the active layers of these circuits [7]. Layers of nanocrystalline silicon can also be grown using chemical vapour decomposition [7].

When the nanoparticulate silicon is deposited as the active layer of printed circuits, it forms an interconnecting backbone of silicon particles [7]. These interconnecting backbones (also known as clusters) of silicon contribute to the electrical characteristics of the material. These clusters

actually contribute to the active layer collectively and not individually [4]. In other words, the contribution is a feature of the structure of the cluster. The clusters act as resistors on the active layers of the printed circuits.

We will be focussing mainly on the resistivity of these clusters of silicon. The aim is to model known particulate silicon resistor networks using computer simulation. The networks to be focused upon are the tetrahedral, hexahedral, octahedral and icosahedral clusters of silicon. These clusters are the possible natural configurations of  $\text{Si}_4$ ,  $\text{Si}_5$ ,  $\text{Si}_6$  and  $\text{Si}_{13}$  respectively [11], where the numbers refer to the number of particles present in the cluster and are to be discussed in section 2.1.

The conductances of these clusters are to be computed as a function of the size scaling in the packing of the clusters. This is important in order to understand the contributions of these networks to the active layer of electronic materials, because to succeed in the reduction of the size of electronic circuits, the properties of these nanoparticles must be understood in their nanostate. These results are to be compared with measured values obtained from the electrical circuits built in the laboratory.

# 2. Clusters, Kirchhoff's Laws and Conductance

## 2.1 Clusters

The word *cluster* has many definitions depending on the sector in which it is used. We will define it as an aggregate of particles with dimensions of the order of 1 – 100nm, each containing hundreds to several thousand atoms. A cluster could contain between two and several hundred particles. A *cluster network* will refer to a combination of many similar clusters which allows them to act as a single unit. An example of a cluster and its equivalent circuit is shown in Fig. 2.1. If all the particles in a cluster are connected to each other, its equivalent circuit forms a complete graph (eg. Fig. 2.1(b)), and the cluster can be said to be saturated. In an unsaturated cluster not all particles have a direct connection to each other and the equivalent circuit is an incomplete graph (eg. Fig. 2.2(b)).

An electrical cluster network is important because it will help us understand how the properties of a nanoparticulate silicon thick film vary with size. Properties of bulk materials like the colour of the material, its electrical conductivity and some other properties can be understood from the study of clusters [12]. In this work we will be concerned with the way the electrical conductivity, varies with change in the size of the clusters. To obtain the conductivity of a cluster, Kirchhoff's laws will have to be used at both the junctions and nodes of the cluster network to compute the effective resistance ( $R_{\text{eff}}$ ). We will be concentrating on some known clusters which will be described below.

### 2.1.1 Tetrahedral Cluster

This is a cluster which is made up of 4 particles as shown in Fig. 2.1(a). It is called a tetrahedral cluster because it has four faces, six edges and four vertices. This could be said to be a regular polyhedron since this network has the properties of a regular tetrahedron.

Its network, shown in Fig. 2.1(b), consists of four nodes connected by six identical resistances each corresponding to the bulk and contact resistances for one particle. From Fig. 2.1(b), the network exhibits symmetry between any two connections. Each line in the figure is an equivalent resistance element.

### 2.1.2 Hexahedral Cluster

This cluster is made up of five particles with connections as shown in Fig. 2.2. This figure does not satisfy a regular hexahedron structure which is expected to have eight vertices and twelve edges, but it does have six faces which is the same in a regular structure.

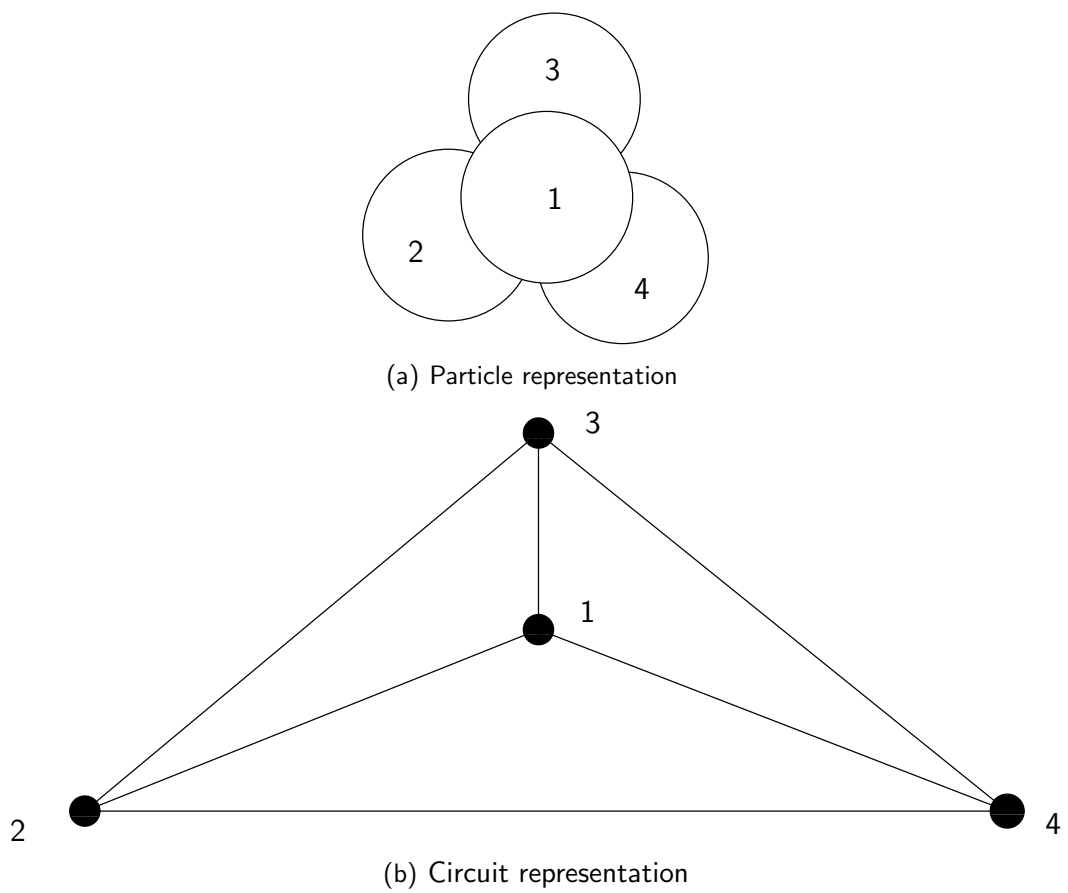


Figure 2.1: Tetrahedral Cluster connected with equivalent resistance.



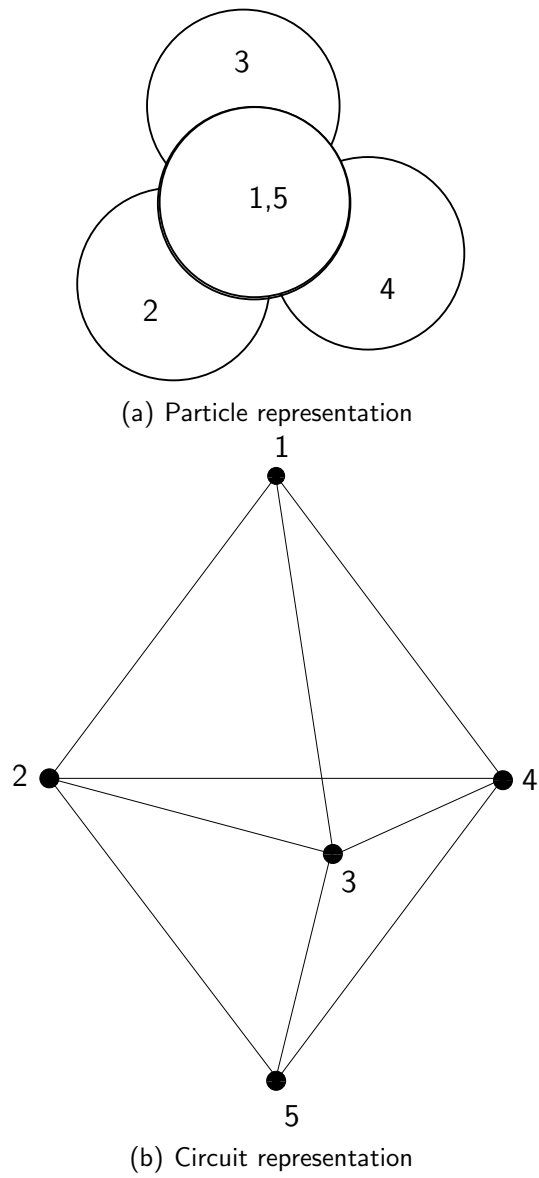


Figure 2.2: Hexahedral Cluster connected with equivalent resistance.

Each line in the figure is an equivalent resistance element. The line of symmetry in this case occurs between 1 and 2, 1 and 3 or 1 and 4. If we compute  $R_{\text{eff}}$  between 1 and 5 we will obtain a different result from the other connections. Take note that in this cluster, 1 and 5 are not connected with an equivalent resistance.

### 2.1.3 Octahedral Cluster

The clusters become more complex as we add particles to them. The octahedral cluster has six particles in it and some of its particles are not interconnected as we have seen with the first cluster. Figure 2.3(b) shows how the particles are connected. Unlike the hexahedral cluster, this is an example of a perfect octahedron structure with six vertices, eight faces and twelve edges. 1 and 6, exhibit symmetry with 2 and 4 and it can be seen that these points do not have an equivalent resistance connected between them.

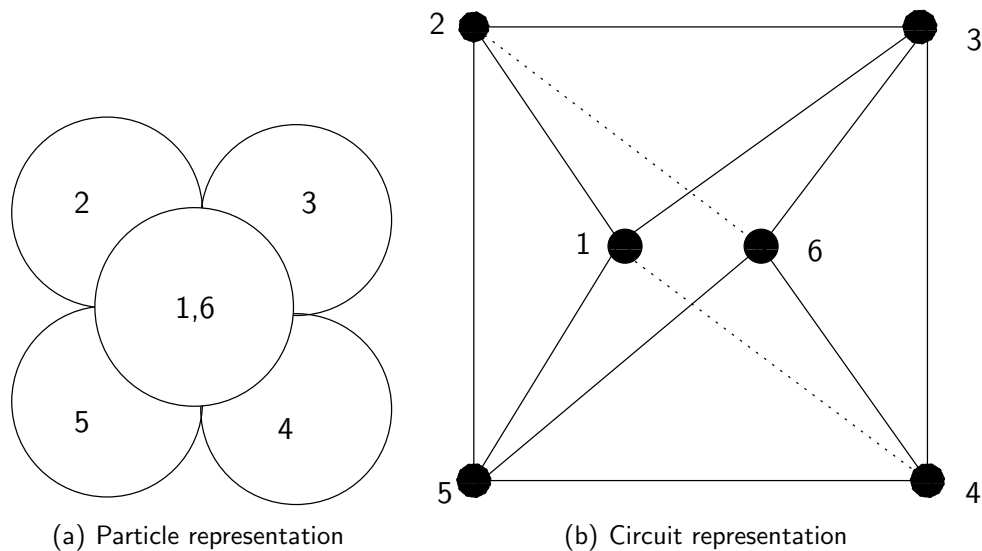


Figure 2.3: Octahedral Cluster connected with equivalent resistance.

### 2.1.4 Icosahedral Cluster

The icosahedral cluster is made up of thirteen particles and is, in essence a cluster of clusters. From Fig. 2.4(b), if we take away the particle labelled zero, we will get a perfect icosahedron structure with twelve vertices, thirty edges and twenty faces.

In this case, many particles are actually not saturated, i.e. they do not share connections with all the particles in the cluster besides close neighbours. Some of the connections which are expected to be symmetrical when a potential is connected across them are 1 and 4, 2 and 5 and 3 and 6. This is likely as they are all diagonally placed as shown in Fig. 2.4.

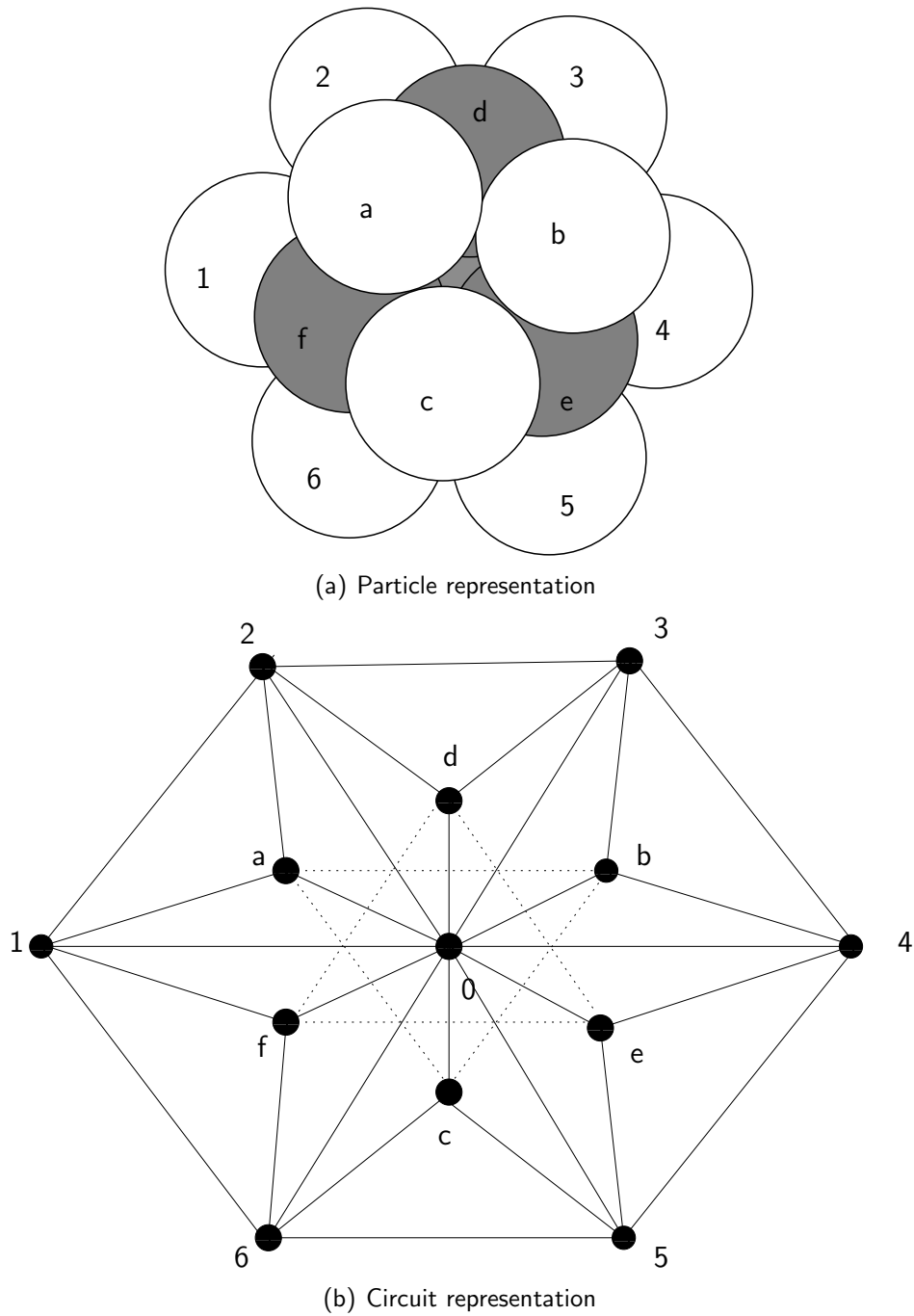


Figure 2.4: Icosahedral Cluster connected with equivalent resistance.

## 2.2 Kirchhoff's Laws

### Kirchhoff's Current Law (KCL)

The algebraic sum of the currents flowing into a node is zero. In other words, for all nodes  $i$ ,

$$\sum_j I_{ij} = 0 \quad (2.1)$$

where  $I_{ij} = -I_{ji}$  is the current flowing from node  $i$  to node  $j$ .

### Kirchhoff's Voltage Law (KVL)

Around a closed loop in a circuit, the sum of the potential differences across all elements is zero.

$$\sum_i \Delta V_i = 0. \quad (2.2)$$

The current law can be modified using Ohm's law and it can be written as

$$\sum_j G_{ij}(V_i - V_j) = J_i, \quad (2.3)$$

where

$$\sum I_{ij} = J_i. \quad (2.4)$$

Here, we are modifying equation 2.1 to include the external currents and its properties will be discussed later in section 3.1.2. Using these laws the effective resistance and subsequently the conductance of the clusters were computed. An example of how Kirchhoff's laws were used to compute  $R_{\text{eff}}$  for the circuit in Fig. 2.1(b) is discussed in the next section.

### 2.2.1 Example to Compute $R_{\text{eff}}$

If we draw the equivalent circuit of Fig. 2.1(b), we can compute the effective resistance between two points. We will use Fig. 2.5 as an example and compute the effective resistance between nodes A and B.

Using equation 2.1 at point A, the equation of the current going in and coming out of the junction can be written as

$$I = I_1 + I_2. \quad (2.5)$$

At the junction C the current is

$$I_2 = I_3 + I_4. \quad (2.6)$$

Using KVL we can deduce the following; at node E, the potential difference is  $V/2$  because we have gone halfway between the points A and B. The same result holds for node D. Thus no current flows between the nodes D and E since they are at the same potential and the resistance

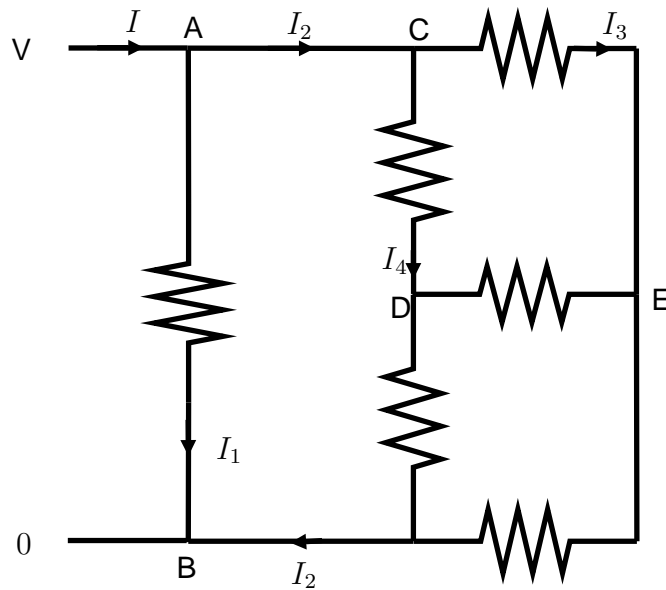


Figure 2.5: Equivalent circuit of Fig. 2.1(b)

between them does not contribute to the effective resistance of the circuit. The equations of the potentials at these nodes are:

$$V = I_1 R \quad (2.7)$$

$$\frac{V}{2} = I_4 R = I_3 R. \quad (2.8)$$

Combining equations 2.5, 2.6, 2.7 and 2.8 we find a result for the effective resistance which is

$$R_{\text{eff}} = \frac{V}{I} = \frac{R}{2}. \quad (2.9)$$

This result simply means that the total resistance due to a tetrahedral network between any two of its particles is half the value of the resistance connected across the particles.

## 2.3 Conductance

Conductance  $G$  can be defined as the measure of how easily electricity flows along a certain path through an electrical element. It is the reciprocal of resistance and its unit is called the Siemens  $S$ :

$$G = \frac{1}{R}. \quad (2.10)$$

In the case of an electric cluster network, measuring the conductivity will require some understanding of the system. Some connections of the network may not be conducting so they do not contribute to the total conductivity of the cluster [13]. The method used by Broadbent and Hammersley [14] to obtain the conductivity of a cluster network was the percolation method.

They computed the number of paths that contribute to the current flow by randomly assigning values to the conductance along the connected paths.

Kirkpatrick [15] tried to compute the conductivity  $\sigma$  for a 2D square lattice cluster and he did this by assigning a local conductivity  $\sigma(r)$  to each material at the position  $\mathbf{r}$  in the cluster. He computed the microscopic current to be

$$\mathbf{j}(r) = \sigma(r)\nabla V(r). \quad (2.11)$$

Equation 2.11 may be solved using a finite difference approximation. Introducing a regular cubic mesh of points  $\mathbf{r}_i$  with spacing  $\Delta r$ , one obtains a system of linear equations for the voltages  $V_i = V(r_i)$ ,

$$\sum_j g_{ij}(V_i - V_j) = 0 \quad (2.12)$$

$$g_{ij} = \Delta r \sigma[(r_i + r_j)/2], \quad (2.13)$$

where  $i, j$  are neighbouring sites on the mesh and  $g_{ij}$  represents random conductances assigned to various connections in the network. Equation 2.12 is the Kirchhoff's current law equation and it holds for all  $\mathbf{r}_i$  not at the surface.

For our own networks, we will compute the conductance using equation 2.10 where  $R$  is the effective resistance generated.

# 3. Computer Model and Electrical Circuits

## 3.1 Computer Model

To build the model, the circuit laws (i.e. Kirchhoff's laws) were taken into consideration. We shall use these laws to derive the equation with which the computer simulations were obtained. Two ways were used to compute the conductance of the circuits and the most efficient will be determined of the two models. The first model will be concerned with the nodes i.e, using Kirchhoff's laws to solve the circuits while the second model will use the conductance from the conducting connections in the circuit. All code is written in python.

### 3.1.1 First Method

This method takes the nodes of the circuit and uses the Kirchhoff's junction and nodal laws to compute the effective resistance between two points. This method operates in a similar way to what Kirkpatrick [15] did for site percolation. We will explain in detail how the program works.

#### The Program

The program operates to solve the Kirchhoff's laws using a system of equation as shown in section 2.2.1. It has the following segments; the input, the body and the output. We shall explain the function of each of these segments as they apply to the solution of the circuit.

#### Input

The code requires some known parameters like the value of the resistance used, the potential and the connections to be inputed before it can solve for the resistance. The code below shows the section which request for the known parameters.

```
particles = input('What is the number of particles in this circuit?')
volts = input('What is the value of the potential?')
R = input('Input the resistance:')
resistors = asarray(read_array("icosahedral_cluster.dat"),Int)
vertices = input('Input the two connected points as a list:')
```

The 'particles' in the first line of the code means the number of particles required in any of the circuits discussed in section 2.1 that is being considered. The resistors in the fourth line is an

$m \times 2$  matrix which records all the possible combinations that we have in a circuit where  $m$  is the number of connections in the circuit; appendix A.3 shows the combinations available for the icosahedral cluster and how they are arranged in the file read by resistors in the code. Note that the combination must be numbers from 1 to  $n$ , where  $n$  is the number of particles present; for example in the icosahedral cluster, 0 was represented with 13 and the other alphabets took different numbers. Vertices is a list  $[a,b]$  where  $a$  and  $b$  are the points where the potential is connected across.

## The Body

What we seek to do is to solve a system of linear equations. Since we do not know the value of any of the current passing through the various combinations, then we will have same number of unknowns as there are connections in the circuit. With this idea in mind, we built the body of the code which is shown below.

```
n = len(resistors)
K = zeros((n,n),Float)
b = zeros(n)
N = particles

'''Solve equations for the current law. This takes the matrix K and
replace the zeros with the coefficient of currents. For all
junctions not in vertices the KCL is solved.
'''
for i in xrange(n):
    if (i+1) in resistors[:,0] :
        if (i+1) not in vertices:
            for j in xrange(len(resistors[:,0])):
                if (resistors[j][0] == (i+1)):
                    h=resistors[j][:]
                    for d in xrange(n):
                        if alltrue(h== resistors[d][:]):
                            K[int(h[0])-1,d] = 1
    if (i+1) in resistors[:,1]:
        if (i+1) not in vertices:
            for j in xrange(len(resistors[:,0])):
                if (resistors[j][1] == (i+1)):
                    h=resistors[j][:]
                    for d in xrange(n):
                        if alltrue(h== resistors[d][:]):
                            K[int(h[1])-1,d] = -1

'''solve equations for the voltage law. This loop takes all the
sufficient parts and replaces the zeros in the matrix K with
the coefficients of the currents I ie, the resistance R. It
also take the matrix b and replaces the zeros with the V.
```



```

'''
M = resistors

MM=[]
for elt in M: MM.append(list(elt))

T = sufficient_paths(vertices[0],vertices[1],M)

u = 0
for i in xrange(n):
    if alltrue(K[i] == 0):
        b[i] = -volts
        for j in xrange(len(T[u])):
            if [T[u][1],T[u][0]] == vertices:
                for q in xrange(len(MM)):
                    if MM[q] == vertices:
                        K[i,q] = -R
            else:
                if j != len(T[u]) - 1:
                    if [T[u][j],T[u][j+1]] in MM:
                        for s in xrange(len(MM)):
                            if [T[u][j],T[u][j+1]] == MM[s]:
                                K[i,s] = -R
                    elif [T[u][j+1],T[u][j]] in MM:
                        for s in xrange(len(MM)):
                            if [T[u][j+1],T[u][j]] == MM[s]:
                                K[i,s] = R

        u += 1
'''Using the two matrix obtained for K and b, the currents are
    computed as a linear system of equations and they are stored
    in X
'''
X = linalg.solve(K,b)
I = []
for r in xrange(n):
    if vertices[1] in MM[r]:
        I.append(abs(X[r]))
I = sum(array(I))

```

To solve the linear equations, we solve the matrix equation

$$Ax = b$$

where  $A$  is an  $(m \times m)$  matrix containing the resistors,  $b$  is a  $(1 \times m)$  matrix containing the potentials,  $x$  is the matrix containing the currents which are not known and  $m$  is the number of connections present in the circuit. The first part of the body of the equation creates the matrices

where the known parameters are to be stored.  $K$  and  $b$  have zeros ( $0s$ ) as their initial entries which will later be replaced with the values of the resistors and potentials respectively.

The second part of the code creates the equations of the currents using KCL. For all the junctions beside the two junctions where the potential was connected, the current equations were formed. Since there are no resistors in the current equations as seen in equation 2.5, the zeros in the resistor matrix were replaced with  $1s$  or  $-1s$  (depending on the direction chosen) at the position where the currents are connected.

The next part uses the nodes to form the voltage equations. It searches for possible closed loops and the result is used to fill the rest of the matrix  $K$  containing the resistors and this time also fills the matrix  $b$  for the potentials based on how many voltage equations were used to fill  $K$ .

The last part of the code above which involves  $X$  is the part where the currents are stored in a  $(1 \times m)$  matrix. Then the currents which contributes to the current entering the circuit are sorted out and summed.

## Output

The output expected is the conductance which is given by equation 2.10. Using that equation we computed the output of the code using the commands shown in the code below.

```
print 'the effective resistance is', volts/I , 'ohms'  
print 'the conductance is', I/volts, 'Siemens'
```

The code which was able to solve for the conductances of the 4 clusters used in this work is generalised to solve for any network provided all the connections are inputted in the file that contains the connection.

### 3.1.2 Second Method

This method was concerned with the individual conductances in the circuit. It followed the model of bond percolation as used by Kirkpatrick [15].

The method made use of an  $n \times n$  square matrix where  $n$  is the number of particles present in the circuit. To write the code of this model, we solved a general equation for the networks that requires the use of the individual conductances in the circuit. We shall derive the equation using the circuit in Fig. 4.1.

We shall define the following terms:

$$I_{ij} = \text{current from } i \text{ to } j$$

$$G_{ij} = \text{conductance between } i \text{ and } j$$

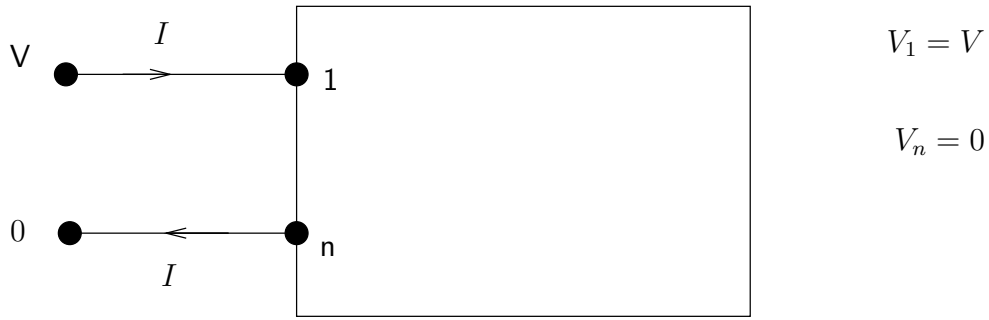


Figure 3.1: A sample network

with  $I_{ii} = G_{ii} = 0$ . By definition,  $I_{ij} = -I_{ji}$ ,  $G_{ji} = G_{ij}$ .  $J_i$  is the external current into node  $i$ . From Fig. 4.1, applying external connection at two points defined by 1 and  $n$ , we will have the values of  $J_i$  for all the nodes to be

$$\mathbf{J} = (I, 0, \dots, 0, -I).$$

Thus from equation 2.1, we have the relationship

$$\sum_j I_{ij} = J_i. \quad (3.1)$$

But

$$I_{ij} = \frac{V_i - V_j}{R_{ij}} = G_{ij}(V_i - V_j).$$

Equation 3.1 can then be written as

$$\sum_j G_{ij}(V_i - V_j) = J_i. \quad (3.2)$$

Rewriting equation 3.2,

$$\sum_j A_{ij}V_j = J_i, \quad (3.3)$$

where

$$A_{ij} = -G_{ij} + \left(\sum_{k=1}^n G_{ik}\right)\delta_{ij}. \quad (3.4)$$

Our aim is to obtain the equivalent resistance  $R_{\text{eff}} = V/I$ . Consider the linear system in equation 3.3, for  $i = 1$  we have

$$\sum_j^n A_{1j}V_j = J_1.$$

Using the properties of the circuit in Fig. 3.1, we have

$$A_{11}V + \sum_{j=2}^n A_{1j}V_j = I, \quad (3.5)$$

and from equation 3.4

$$A_{11} = \sum_{i=1}^n G_{1i}; \quad A_{1j} = -G_{1j}.$$

Therefore, equation 3.5 becomes

$$\sum_{i=1}^n G_{1i}V - \sum_{j=2}^n G_{1j}V_j = I. \quad (3.6)$$

Since  $V_n = 0$  from Fig. 3.1, we can rewrite equation 3.6 as

$$\sum_{i=1}^n G_{1i}V - \sum_{j=2}^{n-1} G_{1j}V_j = I. \quad (3.7)$$

For  $i = \alpha = 2, 3, \dots, (n-1)$  and  $j = \beta$  in equation 3.3, then,

$$\sum_{\beta} A_{\alpha\beta}V_{\beta} = J_{\alpha}$$

$$A_{\alpha 1}V + \sum_{\beta=2}^{n-1} A_{\alpha\beta}V_{\beta} = 0 \quad (3.8)$$

where  $A_{\alpha 1} = -G_{\alpha 1} = -G_{1\alpha}$  from 3.4. Equation 3.8 becomes,

$$\sum_{\beta=2}^{n-1} A_{\alpha\beta}V_{\beta} = G_{1\alpha}V. \quad (3.9)$$

Let

$$\tilde{A} = \left( \begin{array}{c|cc} 0 & 0 & 0 \\ \hline 0 & A_{\alpha\beta} & 0 \\ \hline 0 & 0 & 0 \end{array} \right), \quad \tilde{V} = \begin{pmatrix} 0 \\ V_{\alpha} \\ 0 \end{pmatrix}, \quad \text{and} \quad \tilde{G} = \begin{pmatrix} 0 \\ G_{1\alpha} \\ 0 \end{pmatrix}.$$

We can now write equation 3.9 as

$$\begin{aligned} \tilde{A}\tilde{V} &= V\tilde{G} \\ \tilde{V} &= V\tilde{A}^{-1}\tilde{G} \end{aligned} \quad (3.10)$$

where  $\tilde{A}^{-1}$  is an  $(n-2) \times (n-2)$  matrix. Substituting 3.10 into 3.7 we get the result

$$V \sum_{k=1}^n G_{1k} - \tilde{G}\tilde{A}^{-1}\tilde{G}^T V = I. \quad (3.11)$$

Simplifying equation 3.11 we will obtain the equation for the resistance which is,

$$R_{\text{eff}} = \frac{V}{I} = \frac{1}{\sum_k G_{1k} - \tilde{G} \cdot \tilde{A}^{-1} \cdot \tilde{G}^T}. \quad (3.12)$$

Thus the conductance is given by

$$G = \frac{1}{R_{\text{eff}}} = \sum_k G_{1k} - \tilde{G} \cdot \tilde{A}^{-1} \cdot \tilde{G}^T. \quad (3.13)$$

With the expression of equation 3.13 the second model was built. This model at the end looked simpler than the first because it only takes into consideration the connections that contributes to the flow of currents in the network. Also it is independent of the potential used. We shall explain the details of the code in sections which include the input, body and output.

### Input

Similar to the first model, the known parameters are given to the code. The known parameters are the connections in the circuit, the resistance used and the two points the potential will be connected across. The codes which requires these are shown below.

```
fnet = asarray(array_import.read_array("tetra_cluster.dat"), Int)
R = input('Input the value of the resistance needed:')
terminal1 = 1
terminal2 = 2
```

The file of the connections is read and stored as an  $m \times 2$  matrix where  $m$  is the number of connections available in the circuit being considered. The terminal keeps the records of the two points connected to the potential.

### Body

The idea for the body is to form an  $(n-2) \times (n-2)$  matrix where  $n$  is the number of particles in the circuit. This is required to form the matrix  $\tilde{A}^{-1}$  which is in equation 3.10. The codes below shows how this matrix is formed and how it was used to solve equation 3.11.

```
vertices = list(Set(fnet[:,0]).union(fnet[:,1]))
vertices2 = vertices[:]
N = len(vertices)

G = zeros((N,N), Float)

vertices2.remove(terminal1);
vertices2.remove(terminal2);
new_vertices = [terminal1] + vertices2 + [terminal2]

for link in fnet:
```

```

i, j = new_vertices.index(link[0]), new_vertices.index(link[1])
G[i, j] = 1.
G[j, i] = 1.

A = - G + diag(sum(G))
A_AB = A[1:N-1, 1:N-1]
G1 = reshape(G[0, 1:N-1], (1, N-2))
R=1/(sum(sum(G1))+G[0, N-1]-dot(G1, dot(inv(A_AB), transpose(G1))))[0, 0]

```

The matrix  $G$  which represents  $G_{ij}$  in equation 3.4 is of size  $n \times n$  (where  $n$  is the number of particles), was initially set at zeros. The loop afterwards was used to fill the matrix with the corresponding values of the conductances connected between nodes. The matrix  $A_{AB}$  is the matrix of size  $(n-2) \times (n-2)$  which represents  $\tilde{A}$  in equation 3.10. The last line of the code above was used to solve equation 3.12 and the effective resistance between the connected points is obtained from it.

## Output

The output of this code only requires that the conductance be printed out.

```

print "Equivalent Resistance:", R, 'ohms'
print "Conductance:", 1/R, 'siemens'

```

At the end of the computations, the two models were compared and their differences obtained. The first model ran for a total time of 6.3435 secs while the second ran for  $995\mu s$ . It can be seen that the second model is about 6000 times faster than the first. Also, the lengths of the codes vary greatly. As seen in appendix A.1 and A.2, the first model is about three times longer than the second. This is to say that using the second code is far more efficient as it saves time and uses less computer memory.

## 3.2 Electrical Circuits

Clusters of resistors were built for the four different structures and the effective resistances measured across some of the nodes. The apparatus used were, Fluke III True RMS Multimeter,  $10k\Omega$  ( $\frac{1}{4}W$ , 1% tolerance) resistors, breadboards, Topward DC power supply 6303D. The pictures of all the experimental setups are as shown below with the first three clusters built together and the icosahedral cluster built separately.

For the tetrahedral cluster, a total of six resistors were used, nine for the hexahedral and twelve for the octahedral, the icosahedral cluster has thirty six resistors connected. The pictures of the networks and their connections are as shown in figures 3.2 and 3.3.

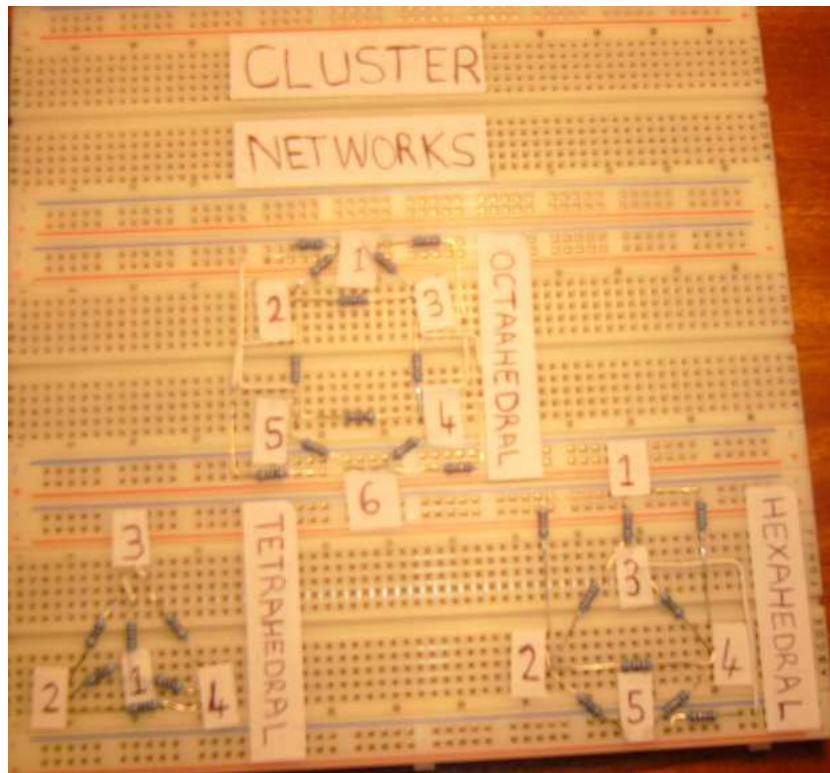


Figure 3.2: Resistor connections of the tetrahedral, hexahedral and octahedral cluster networks

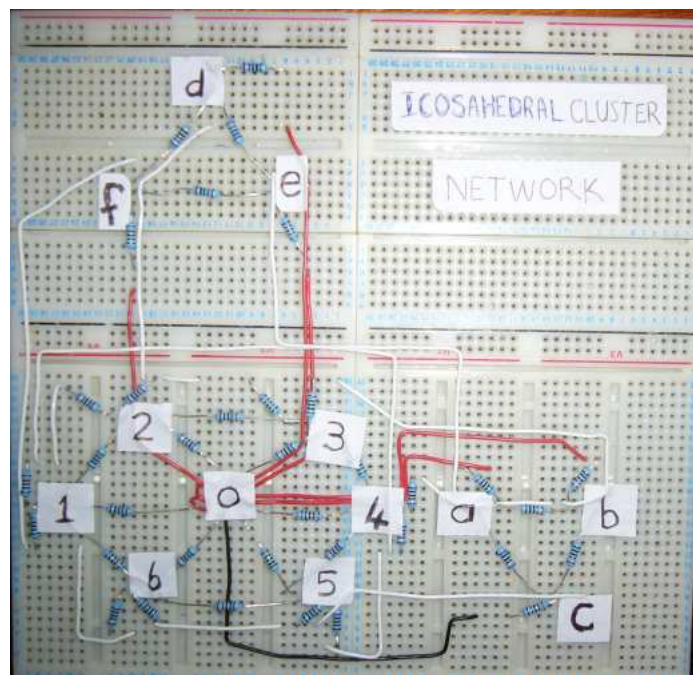


Figure 3.3: Resistor connections of the icosahedral cluster

## 4. Results and Discussions

The results for the computer model and those obtained from the actual experiments are presented here. Table 4.1 is the result obtained for the computer simulation. The conductances between some connections are shown for all the clusters.

Table 4.1: Conductances obtained for the clusters from computer simulation

Cluster	Connections	Effective Resistance(k $\Omega$ ) Upper – Lower	Conductances (mS) Upper – Lower
Tetrahedral	any two points	5.005 – 4.995	0.1998 – 0.2002
Hexahedral	1 & 2 or 1 & 3 or 1& 4	4.671 – 4.662	0.2141 – 0.2145
Octahedral	1 & 6 or 2 & 4 or 3& 5	5.005 – 4.995	0.1998 – 0.2002
Icosahedral	1 & 4 or 2& 5 or 3& 6	4.767 – 4.757	0.2098 – 0.2102

The resistors used have a tolerance of 1%, so their values could be in the range  $10000 \pm 10\Omega$ . Due to this, table 4.1 shows a record of the boundary limit of the error that can be measured with such resistors. The upper boundary limits of the resistors were calculated by adding the tolerance to the supposed value of the resistance, while the lower were by subtracting the tolerance value. This choice of error value is too large, but it was adopted since the actual errors can not be determined. For the conductances, results obtained from the resistors were utilised.

Table 4.2 shows the measured values obtained for all the possible connections in the tetrahedral network. The results obtained shows that the measured effective resistances lies between the upper and lower boundaries as calculated from the computer simulation.

Table 4.2: Measured values for the tetrahedral network

Connection	Resistance (k $\Omega$ )	Conductance (mS)
1 - 2	4.990	0.200401
1 - 3	4.987	0.200521
1 - 4	4.994	0.20024
2 - 3	5.000	0.2000
2 - 4	5.004	0.19984
3 - 4	5.001	0.19996

To verify if both results from the computed and measured were correct, a potential of  $10V$  was connected between nodes 2 and 3 of the tetrahedral circuit. The results obtained at each node of the circuit is as shown in Fig.4.1.

To compute the total current flowing from node 2 to node 3 we compute the currents flowing out of node 2 into the circuit and these are  $I_{23}$ ,  $I_{24}$  and  $I_{21}$ . Thus, using Ohm's law, the total



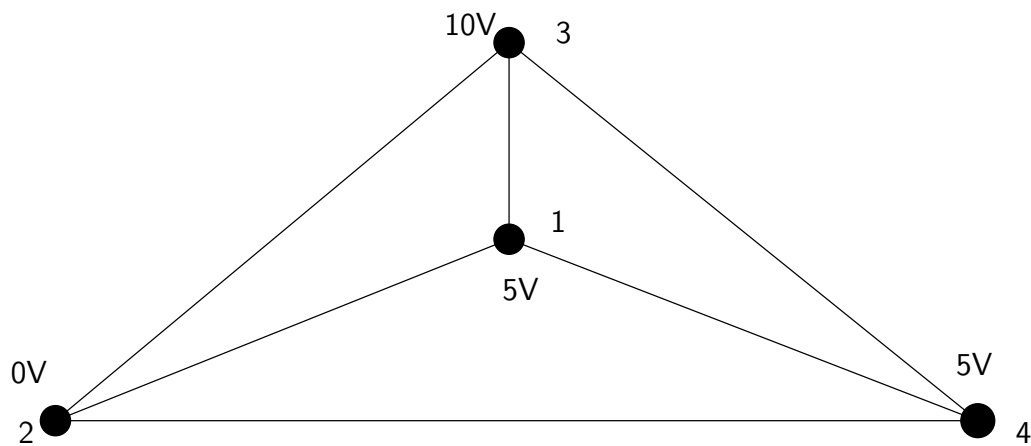


Figure 4.1: Tetrahedral network with the voltages at each node.

current  $I_T$  is given as:

$$\begin{aligned}
 I_T &= I_{23} + I_{24} + I_{21} \\
 &= \frac{10}{R} + \frac{5}{R} + \frac{5}{R} \\
 &= \frac{20}{10 \times 10^3} \\
 &= 2\text{mA}.
 \end{aligned}$$

The total resistance (or effective resistance  $R_{\text{eff}}$ ) between nodes 2 and 3 was then computed using the method shown below,

$$\begin{aligned}
 R_{\text{eff}_{2-3}} &= \frac{V_T}{I_T} \\
 &= \frac{10}{2 \times 10^{-3}} \\
 &= 5\text{k}\Omega.
 \end{aligned}$$

It can be seen that the calculated value of the total resistance lies between the error limit obtained from the computer model. So, if the values of the voltage across the nodes in a circuit are known, then  $R_{\text{eff}}$  between any two points can be easily computed.

Table 4.3 is a summary of the measured values of the conductances and resistances across connections in the hexahedral cluster. Some few points which are considered to be equivalent (from the circuit) are chosen and compared with the computed values of the computer model.

Connecting a potential of  $10V$  across 1 – 2 in the hexahedral circuit, the results obtained are as shown in Fig. 4.2.

The effective resistance obtained between these points was  $4.665 \text{ k}\Omega$ . This value also lies between the expected range of the computer model.

Table 4.3: Measured values for the hexahedral network

Connection	Resistance ( $k\Omega$ )	Conductance (mS)
1 - 2	4.673	0.213995
1 - 3	4.671	0.214087
1 - 4	4.667	0.21427

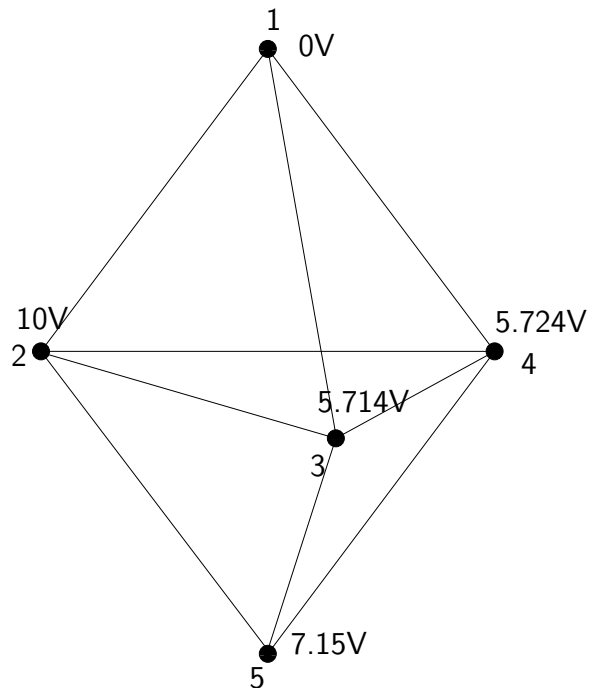


Figure 4.2: Hexahedral network with the voltages at each node.

Tables 4.4 and 4.5 are the measured results for the octahedral and icosahedral circuits respectively. Similar investigations were carried out to verify their results as was done for the first two circuits.

Table 4.4: Measured values for the octahedral network

Connection	Resistance ( $k\Omega$ )	Conductance (mS)
1 - 6	4.997	0.20012
2 - 4	5.006	0.19976
3 - 5	5.018	0.199283

Fig. 4.3 shows all the measured potentials at each node for the octahedral circuit. Using the same principle as above, the effective resistance between nodes 1 and 6 was calculated to be  $5.006k\Omega$  when a potential of  $10V$  was connected across them. This value is a little bit above the computed range of the computer model and the reason could be attributed to the fact that we may not be able to measure the exact error in each connection.

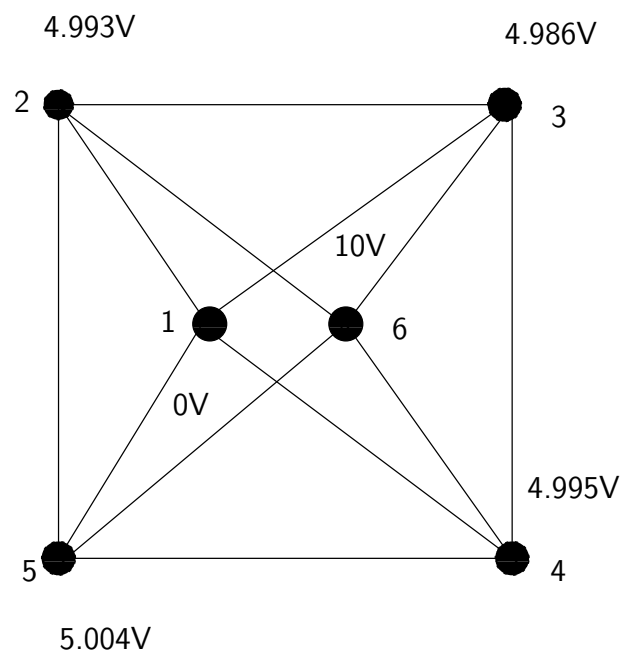


Figure 4.3: Octahedral network with the voltages at each node.

For the Icosahedral network, the potential of  $10V$  was connected between nodes 1 and 4. Fig. 4.4 shows all the measured values across each node.  $R_{\text{eff}}$  between these points was calculated to be  $4.791k\Omega$ . This value is also a bit large because the potentials were approximated to nearest whole numbers and also actual errors may not be easily determined across the nodes.

Conductances for other points could be computed from the simulation and also measured directly from the experiment, but we discussed just some few. In the icosahedral cluster, only the diagonal connections were reported here and same applies to the octahedral cluster. This is because when these clusters will interact with other clusters, they will have to interact across the diagonal

Table 4.5: Measured values for the icosahedral network

Connection	Resistance ( $k\Omega$ )	Conductance (mS)
1 - 4	4.756	0.210261
2 - 5	4.759	0.210128
3 - 6	4.761	0.21004

connections. Also, it should be noted that for simplicity we have chosen the resistors to be of equal values. This could be modified to accommodate resistors of different values and the effective resistance can then be calculated.

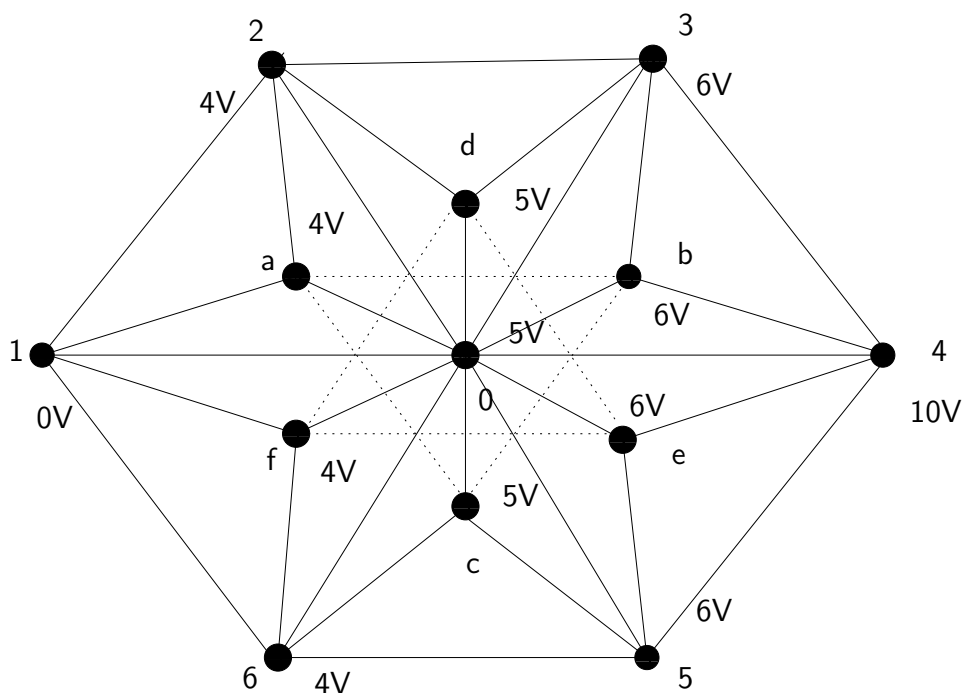


Figure 4.4: Icosahedral network with the voltages at each node.

## 5. Conclusion

We have been able to use a computer simulation to compute the conductances of different resistor clusters. For simplicity, the connections across each cluster were made of same resistance. This could be adjusted to use resistors of different values. Two computer models (both written in python) that used different approaches were built and the most efficient of the two was obtained. The first model considered the potentials at the nodes of the circuits while the second used the conductances of the conducting connections. The second model was about 6000 times faster in running time than the first and it also uses less computer memory because of its length.

Results obtained from electrical circuits built, were within the range of the computed values. The tolerance present in the resistors was used to determine the limit of experimental error that could possibly occur in the measurements. The measured values actually fell within the given limits of the errors. A third way was used to verify the results. This was done by connecting a potential of  $10V$  across two points in the circuits, using Ohm's law the total currents were calculated. The effective resistances were then calculated using the total potential and currents. Results obtained from the third method were in agreement with both the measured and computed values.

With these models, the contributions of more complicated clusters to electronic circuits could be studied without having to build the clusters. This work could be extended to contain capacitors, so that AC circuits could also be studied.

# Appendix A. The Codes for the Models

## A.1 First Method

```
''' Program to compute equivalent resistance for cluster networks using
    the potentials and currents across each node.
'''

''' Import necessary modules
'''
from __future__ import division
from scipy import *
from scipy.io.array_import import *
from time import *

'''Define a function that takes two entries; a=a number in the connection, M = matrix
    containing the entire connections and at the end returns a list of
    the neighbours of 'a'
'''
def connected_to(a,M):
    g = []
    if a in M[:,0]:
        for i in xrange(len(M)):
            if M[i][0]==a:
                g.append(M[i][1])
    if a in M[:,1]:
        for i in xrange(len(M)):
            if M[i][1]==a:
                g.append(M[i][0])
    return array(g)

'''Define a function that takes in four entries(v = vertices[1],
    a = vertices[0], avoided = empty list, net = transpose of MM).
    The function returns all possible paths beginning from v and
    terminating at a
'''

def path_to(v, a, avoided, M):
    paths=[]
    sel_nb = []
    nb = connected_to(v,M)
    for node in nb:
        if not node in avoided:
```

```

        sel_nb.append(node)
    for node in sel_nb:
        if node == a:
            path = [[a]]
        else:
            new_avoided = avoided + [v]
            path = path_to(node, a, new_avoided,M)
        for elt in path:
            paths.append(elt + [v])
    return paths

def paths_between(terminal1, terminal2, M):
    #print "Computing paths ..."
    return path_to(terminal2, terminal1, [], M)

'''Define a function that takes in 3 entries(vertices[0],vertices[1],the matrix
of the connections, M). The function returns the lists of paths or loops that
are necessary to compute the Kirchhoff KVL equations
'''

def links(path):
    lnks = []
    nlinks = len(path) - 1
    i = 0
    while i < nlinks :
        lnks.append([path[i], path[i+1]])
        i+=1
    return lnks

def is_in(net, link):
    return ([link[0], link[1]] in net) or ([link[1], link[0]] in net)

def sufficient_paths(terminal1, terminal2, M):
    #print "Determining minimum number of sufficient paths ..."
    net = transpose(M)
    list_of_paths = paths_between(terminal1, terminal2, M)
    print "Number of Paths from %d to %d:" % (terminal1, terminal2),len(list_of_paths)
    # print "List of Paths from %d to %d:" % (terminal1, terminal2),list_of_paths
    reg_links = []
    good_paths = []
    is_good_path = False
    net_size = len(M)
    for path in list_of_paths:
        path_links = links(path)
        for link in path_links:

```

```

        if not is_in(reg_links, link):
            reg_links.append(link)
            is_good_path = True
    if is_good_path:
        good_paths.append(path)
        is_good_path = False
        if len(reg_links) == net_size:
            break
    return good_paths

'''input the known parameters; the number of nodes, the value of the
resistors and the voltage across the points
'''
nodes = input('How many nodes do you need?')
volts = 1 # input('What is the value of the potential?')
R = 1 # input('Input the resistance:')
#resistors = asarray(read_array("tetra_cluster.dat"),Int)
#resistors = asarray(read_array("hexa_cluster.dat"),Int)
#resistors = asarray(read_array("octa_cluster.dat"),Int)
resistors = asarray(read_array("icosahedral_cluster.dat"),Int)
n = len(resistors)
K = zeros((n,n),Float)
b = zeros(n)
N = nodes

'''create a system of equations using Kirchhoff's laws
'''
vertices = input('Input the two points to compute the potential as a list:')
begin = time()

'''Solve equations for the current law. This takes the matrix K and replace
the zeros with the coefficient of currents. For all junctions not in vertices
the KCL is solved.
'''
for i in xrange(n):
    if (i+1) in resistors[:,0] :
        if (i+1) not in vertices:
            for j in xrange(len(resistors[:,0])):
                if (resistors[j][0] == (i+1)):
                    h=resistors[j][:]
                    for d in xrange(n):
                        if alltrue(h== resistors[d][:]):
                            K[int(h[0])-1,d] = 1
    if (i+1) in resistors[:,1]:

```



```

    if (i+1) not in vertices:
        for j in xrange(len(resistors[:,0])):
            if (resistors[j][1]) == (i+1):
                h=resistors[j][:]
                for d in xrange(n):
                    if alltrue(h== resistors[d][:]):
                        K[int(h[1])-1,d] = -1

'''solve equations for the voltage law. This loop takes all the
sufficient parts and replaces the zeros in the matrix K with
the coefficients of the currents I ie, the resistance R. It also take
the matrix b and replaces the zeros with the potential V.
'''
M = resistors

MM=[]
for elt in M: MM.append(list(elt))

T = sufficient_paths(vertices[0],vertices[1],M)

u = 0
for i in xrange(n):
    if alltrue(K[i] == 0):
        b[i] = -volts
        for j in xrange(len(T[u])):
            if [T[u][1],T[u][0]] == vertices:
                for q in xrange(len(MM)):
                    if MM[q] == vertices:
                        K[i,q] = -R
            else:
                if j != len(T[u]) - 1:
                    if [T[u][j],T[u][j+1]] in MM:
                        for s in xrange(len(MM)):
                            if [T[u][j],T[u][j+1]] == MM[s]:
                                K[i,s] = -R
                    elif [T[u][j+1],T[u][j]] in MM:
                        for s in xrange(len(MM)):
                            if [T[u][j+1],T[u][j]] == MM[s]:
                                K[i,s] = R

        u += 1

'''Using the two matrix obtained for K and b, the currents are
computed as a linear system of equations and they are stored in X
'''

```

```

X = linalg.solve(K,b)
I = []
for r in xrange(n):
    if vertices[1] in MM[r]:
        I.append(abs(X[r]))
I = sum(array(I))
end = time()
print 'the effective resistance is', volts/I , 'ohms'
print 'the conductance is', I/volts, 'siemens'
print end - begin,'seconds'

```

## A.2 Second Method

```

''' Program to compute equivalent resistance for cluster networks using
    the conductances across each bonds.
'''

''' Import the necessary modules
'''
from __future__ import division
from sets import Set
from scipy import *
from scipy.io import *
from scipy.linalg import *
from time import *

''' Read the connections from a file
'''
fnet = asarray(array_import.read_array("icosahedral_cluster.dat"), Int)
begin = time()

''' Input the two points where the potential is connected
'''
terminal1 = 1
terminal2 = 2

vertices = list(Set(fnet[:,0]).union(fnet[:,1])) # put all the numbers in
                                                # the connection in a list

vertices2 = vertices[:] # copy the vertices list to a separate location
N = len(vertices)

G = zeros((N,N), Float) # an (N x N) matrix containing zeroes

```

```

'''Remove the two points connected from the list of vertices
'''
vertices2.remove(terminal1);
vertices2.remove(terminal2);

'''Rearrange the vertices so that the two connected points occupy
positions 1 and N of the vertice list
'''
new_vertices = [terminal1] + vertices2 + [terminal2]

'''Fill the portion of the matrix that contains the conducting bonds
with 1s
'''
for link in fnet:
    i, j = new_vertices.index(link[0]), new_vertices.index(link[1])
    G[i, j] = 1.
    G[j, i] = 1.

A = - G + diag(sum(G))
A_AB = A[1:N-1,1:N-1]    #Create a matrix of the conducting resistances

G1 = reshape(G[0, 1:N-1], (1,N-2))
R = 1/(sum(sum(G1))+G[0,N-1] - dot(G1, dot(inv(A_AB),transpose(G1))))[0,0]
end = time()
print "Equivalent Resistance:", R
print end - begin,'seconds'

```

## A.3 Icosahedral Connections

The two columns shows the connection between the particles of the icosahedral cluster network. The connections implies where the resistors in the circuit will be connected.

```

1 13
1 2
1 6
1 7
1 12
2 13
2 3
2 7
2 8
3 13

```

3 4  
3 8  
3 9  
4 13  
4 5  
4 9  
4 10  
5 13  
5 6  
5 10  
5 11  
6 13  
6 11  
6 12  
7 13  
7 9  
7 11  
9 13  
9 11  
11 13  
8 13  
8 10  
8 12  
10 13  
10 12  
12 13

# Acknowledgements

All thanks be to God who enabled me go through this study without problems.

I will like to appreciate those who helped see to it that this work came out the way it is. My supervisors, Prof. M. Härting and Prof. D.T. Britton, your advice and care to ensure that this work meet an acceptable standard will always be cherished. Also, the contributions of the members of the solid state group, University of Cape Town has been of great value. Daniel Brink was helpful in the building of the circuits, tirelessly he made effort to ensure that even though I was an armature in electronic designs, you could hardly see it. Ayodele Odo was helpful in the explanation of some of the concepts used frequently in the group. Mr I. Khan, Mr. Girma Goro and Miss Ziyanda Sigcav also in no little way made great contributions when we discussed about this work in our meetings.

My parents Mr M.A. Jonah and Mrs Fatima who made effort to ensure that I made it to South Africa for this course, I will always be grateful. My siblings Mrs Elizabeth Abdulkarim, Joseph, David, Jacob, Martha, Samuel and Endurance, I just want to say thank you for your love and care to ensure my stay miles away from you have been a blessing and not a struggle. My appreciation also goes to the Lady in my life that has given me courage to stay far and work without pressure, Veronica Amlabu, thanks for being there always. Also to the family of Mr J. Amlabu, his wife and the children, Mercy, Lucy, Blessing and David, thanks for all the encouraging calls.

My colleagues at AIMS have been too helpfull psychologically and otherwise. People like Wole and Lois have been true friends and I will ever be grateful for knowing you guys. Victoria, Joy, Ndubuisi, Saheed, Magaret, Veronica and others that space will not allow me to include your names, thanks for the interesting times we all had at AIMS.

In my entire education having tutors to guide me during my studies was quite strange, but that was what we were treated with when we came to AIMS. Henry has been a wonderful tutor, a great friend and a brother. Thanks for been a listening ear when we struggled with understanding how to program with python and how to solve some stormbreaking mathematics or physics. Also, Sam helped a lot with my Essay, thanks for carefully going through my writings and correcting the errors. Jan, Igsaan, Laure, Jean-Marie, Paul, Eman, Ambrose, Andy and Anahita have been helpful during my time at AIMS. Thanks also to the Director of AIMS and his board that gave me the privilege to study here at AIMS.

To all who quietly have had me in mind all through my studies, thank you all for your efforts.

# References

- [1] M.J. Pitkethly. Nanomaterials - the driving force, 2004. URL: [www.iom3.org/divisions/mst/nano/Nanoart1.pdf](http://www.iom3.org/divisions/mst/nano/Nanoart1.pdf).
- [2] <http://www.euspen.org>. Life time achievement award, 2002. URL: <http://www.euspen.org/default.asp?contentid=407&ForPrinting=Y>.
- [3] N. Taniguchi. On the basic concept of 'nano-technology'. *Proc. Intl. Conf. Prod. Eng. Tokyo, Path II*, 1974.
- [4] U Landman and W.D. Luedtke. "small is different: Energetic, structural, thermal, and mechanical properties of passivated nanocluster assemblies". *Discussion of the Faraday Society*, 125(1), June 2003.
- [5] K. Satler. The energy gap of clusters nanoparticles, and quantum dots. *Nanomaterials and Magnetic Thick Films*, 5, 2002.
- [6] B.I.; L. Qi; T. Copeland Lee. Nanoparticles for materials design: Present & future. *Journal of Ceramic Processing Research*, 6(1):31–40, Oct 2005.
- [7] D. T. Britton and M. Härting. Printed nanoparticulate composites for silicon thick-film electronics. *Pure Appl. Chem.*, 78(9):1723–1739, 2006. URL: [http://iupac.org/publications/pac/new/article?article\\_id=7809x1723](http://iupac.org/publications/pac/new/article?article_id=7809x1723).
- [8] J.C. Whitaker. *The Electronics Handbook*. IEEE Press, 1996.
- [9] Pitt K.E.G.(ED.). *Handbook of Thick-Film Technology, 2nd Edition*. Electrochemical Publishing Ltd., Ayr, Scotland, 2005.
- [10] RHe. Thick film technology, 2007. URL: [http://www.rhe.de/en/Thick\\_Film\\_Technology.html](http://www.rhe.de/en/Thick_Film_Technology.html).
- [11] J.C. Grossman and L. Mitas. Silicon clusters, 1995. URL: <http://altair.physics.ncsu.edu/projects/sismall/siclust.html>.
- [12] A.G. Gantef. Every atom counts..., 2003. URL: <http://www.cluterphysik.uni-konstanz.de/en/forschung/index.html>.
- [13] J.A. Blackman. A theory of conductivity in disordered resistor networks. *J. Phys. C: Solid State Phys.*, 9:2049–2071, 1976.
- [14] S.R. Broadbent and J.M. Hammersley. *Proc. Camb. Philos. Soc.*, 53:629–41, 1957.
- [15] S. Kirkpatrick. Percolation and conduction. *Rev. Mod. Phys.*, 45(4):574–588, Oct 1973.