UNIVERSITI TEKNOLOGI MARA
University Publication Centre (UPENA)

ESTEEM

Academic Journal UiTM Pulau Pinang

Volume 4, Number 2 2008 ISSN 1675-7939

ESTEEM Academic Journal is jointly published by the Universiti Teknologi MARA, Pulau Pinang and University Publication Centre (UPENA), Universiti Teknologi MARA, 40450 Shah Alam, Selangor, Malaysia.

ESTEEM
Academic Journal UiTM Pulau Pinang

Volume 4, Number 2 | 2008 | ISSN 1675-7939

SOCIAL SCIENCES

Foreword

This is the first time that ESTEEM Academic Journal UiTM Pulau Pinang has come up with 2 publications in a year! Previously, ESTEEM was published once biennially.

For these publications to materialise, I would like to thank Associate Professor Mohd Zaki Abdullah, the Director of UiTM Pulau Pinang for his unflinching support and who always told me, "Go ahead, don't worry about the money!".

Both the Associate Professor Mohd Zaki Abdullah and Dr. Mohamad Abdullah Hemdi, the Deputy Director of Academic Affairs really provided me with a great deal of assistance in ensuring that there are sufficient articles for publishing. Both of them have emphasized the need for lecturers to embark on journal writing. Incidentally this is one of the prerequisites for promotion among the academic staff members of UiTM Pulau Pinang.

I do not think I can run the show alone without the help from the editorial board, reviewers and the cooperation from University Publication Centre (UPENA) of UiTM Malaysia. My special thanks to Mr. Mohd Aminudin Murad for his efficiency in editing articles and to Dr. Khairil Iskandar Othman for speeding up the final stage of printing process.

Since writing is an important criterion in rating a university, I feel it is a great responsibility for me to produce a good journal. Fellow colleagues, let's work closely to put UiTM Pulau Pinang in the final list of Anugerah Kualiti Naib Canselor (AKNC) and Anugerah Kualiti Perdana Menteri (AKPM) by submitting more quality articles to ESTEEM!

Lastly, let me end by thanking all of you for giving your unwavering support to UPENA.

The Chief Editor
November, 2008

ESTEEM, Vol. 4, No. 2, 2008, 39–49

Simulation of Routing Probability in Ad Hoc Networks

Ahmad Zia Ul-Saufie Mohamad Japeri
Muhammad Hisyam Lee
Shaharuddin Salleh

ABSTRACT

An ad hoc network is a group of wireless mobile hosts forming a temporary network without the aid of any established infrastructure or centralized administration. In such an environment, it may be necessary for one mobile host to enlist the aid of other hosts in forwarding a packet to its destination due to the limited range of each mobile host's wireless transmissions. This paper presents the simulation and programming of a probability model for route selection of an ad hoc network for mobile computing. The route selection is based on the probability that the route is still available when a packet has to be sent through it. It is also based on the knowledge at each node of the geographic position of all the other nodes in the network and takes into account the mobility of the nodes and the dependencies between links in a computed route. We derive a simple closed from expression for the computation of the availability of a route, which can then be selected as the best route to serve the purposes of a specific application. Microsoft Visual C++ was used for the purposes of running the simulation.

Keywords: *Microsoft Visual C++, mobile ad hoc network, probability, routing*

Introduction

An ad hoc network is the cooperative engagement of a collection of mobile nodes without the intervention of any centralized access point, and in the absence of a fixed infrastructure. Routes between two mobile nodes in ad hoc network may consist of hops through other mobile nodes in the network. Node mobility can cause frequent unpredictable topology changes. This paper is continuation from the paper work of Muhammad Hisyam, Shaharuddin, and Stephen Oalriu (2001, September). We simulate and program this problem using Microsoft Visual C++ with FMC windows.

In this project, we present a program on simulation to determine the probability model of a wireless radio link that exists between any two mobiles nodes in a computed route, such that the probability of the link between those two nodes is considered available. Furthermore, we compute the probability of the route availability to any given destination from the source node. Specifically, we derive a simple closed form expression for the computation of the availability of a route that takes into account node mobility and dependencies between links, which can then be used to choose the route that best meets the strict requirements of applications.

When a source node needs to route a packet to a given destination based on the location information available locally, it will associate with each route a value that bounds the probability that the route exists for the duration that it is traversed by the packet. The destination can be one or more depending on the routing scheme considered. Thus, any source node can then choose the route based on this probability that satisfies the requirements of applications.

Link Existence Probability

Let γ_Q represents a link from node P to node Q. Probability of link existence between these two nodes, $P(\gamma_Q)$ can be computed by intersecting these two circles. The first circle is centered at node *P* and the second circle centered at node *Q*. We define the first circle to be the transmission radius of node P, and denote by *Tx*(*P*). This is the larger circle shown in Figure 1. As for the second circle, we define it as the movement circle of node Q relative to P, and denote it by *Mv*(*Q*).

We assume that node P to be stationary and centered at the origin (0, 0) of a Cartesian system, such that node Q is centered at (d, 0), where d is the Euclidean distance between node P and node Q. We then assume node Q can move freely in any direction at a maximum velocity of $v = v_P + v_Q$, where v_P and v_Q are the maximum velocities of nodes P and node Q, respectively. We take into account that the two nodes can move closer or away from each other $r_Q = v.\Delta t\, .\Delta t$ Thus, node Q can be found anywhere inside the circle whose radius. is the time elapsed between the time last GPS packet was received at the source from node Q and the current time.

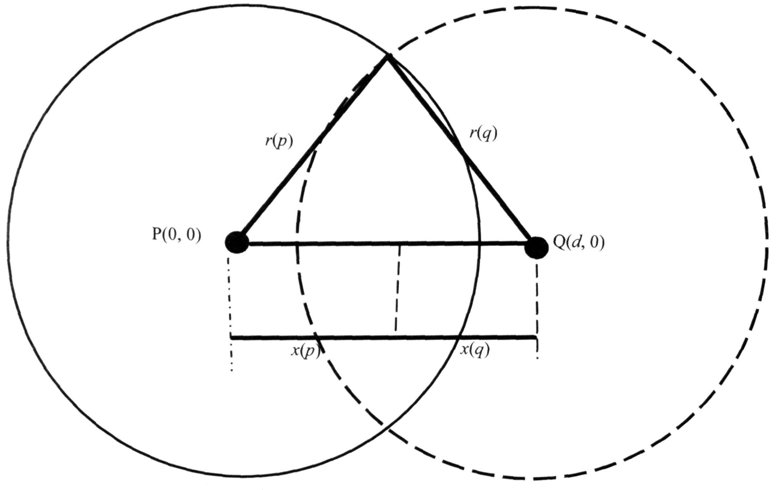

Figure 1: Computing the Probability of Link Existence between Two Nodes (P and Q) from the Intersected Area

If node Q moves into the intersected area (see Figure 1), then link γ_Q exists between these two nodes. Thus the probability $P(\gamma_Q)$ can be computed through the classical approach probability, that is,

$$P(\gamma_Q) = \frac{A_Q}{\pi r_Q^2} \tag{1}$$

where A_Q is the intersected area between these two nodes.

Clearly, the area A_Q can be computed through the two circular segments of radius r_P and r_Q, and triangles x_P and x_Q as shown in Figure 1. The steps to calculate A_Q are as follows:

Step 1: Area for segment of radius r_p is easily computed through the formula $\frac{\theta r^2}{2}$. Thus,

$$r_P^2 \cos^{-1}\left(\frac{x_P}{r_p}\right) \tag{2}$$

is the respective segment area for node P, where $\theta = \cos^{-1}\left({}^{x_P}\!/_{r_P}\right)$. Similarly,

$$r_Q^2 \cos^{-1}\left(\frac{x_Q}{r_Q}\right) \tag{3}$$

is the respective segment area for node P, where $\theta = \cos^{-1}\left({x_Q}/{r_Q} \right)$.

Step 2: Areas for two segment of triangle are $x_P\sqrt{r_P^2 - x_P^2}$ and, $x_Q\sqrt{r_Q^2 - x_Q^2}$ respectively.

Step 3: Thus, the resulting area, A_Q in which two circles intersect can be computed by minus step 2 from step 1, that is,

$$A_Q = r_P^2 \cos^{-1}\left(\frac{x_P}{r_P} \right) + r_P^2 \cos^{-1}\left(\frac{x_Q}{r_Q} \right) - x_P\sqrt{r_P^2 - x_P^2} - x_Q\sqrt{r_P^2 - x_Q^2} \tag{4}$$

where $x_P = \left(d^2 + r_P^2 - r_Q^2\right)/\,2d$ and $x_Q = \left(d^2 - r_P^2 + r_Q^2\right)/\,2d$.

Since the source node P knows all values involved, computation can use standard library function.

A link γ is said to be available if there exists a $P(\gamma) > 0$. If the probability equals to zero, link are not available between two nodes. On the contrary, if $P(\gamma) > 1$, links are definite exist and we can consider this link to be a stationary link.

Important Programming Codes

We include some codes written for the program in this paper.

Some Important Codes

The following codes are for drawing and displaying the node with the defined radius using the right mouse button:

```
void CMainFrame:: OnRButtonDown(UINT nFlags, CPoint pt)
{
    CClientDC dc(this);
    CString s;
    CRect rc;
    CPen penGray(PS_SOLID,1,RGB(50,50,255));

                CFont fontCourier;
    fontCourier.CreatePointFont (100,"Arial");
    dc.SetTextColor(RGB(250,250,250));
    dc.SelectStockObject(HOLLOW_BRUSH);
```

```
    dc.SelectObject(penGray);

if (pt.x>xGMIN && pt.x<xGMAX && pt.y>yGMIN && pt.y<yGMAX)
    if (counter<=N)
    {
         mPt[counter].x=pt.x; mPt[counter].y=pt.y;
         x[counter]=pt.x; y[counter]=pt.y;

mRect [unter]=CRect(pt.x,pt.y,pt.x+5,pt.y+5);
    rc=CRect(mRect[counter]);
         dc.Rectangle(&rc);
         dc.Rectangle(&rc);
         dc.SelectObject(fontCourier);
         dc.SetBkColor(RGB(0,0,0));

         int rp;
         rpx[counter] = rp;
         int t =0;
         while (t<=360)
              {
              x1[counter]=rp*sin(t);
              y1[counter]=rp*cos(t);
         dc.SetPixel((int)x1[counter]+pt.x,(int)y1
         [counter]+pt.y, RGB(250,250,55));
                   t+=1;
              }
         s.Format("%d %d,%d",counter,pt.x,pt.y);
         dc.TextOut(x[counter]-20,y[counter]-15,s);
         s.Format("XY Point %d:(%d,%d):%d",counter,
         pt.x,pt.y,rp);
         dc.TextOut(610,100+30*counter,s);
         counter++;
              }
    }
```

The following codes are used to draw and display the nodes with the random radius using the left mouse button. The coding is similar to the defined radius with additional simple coding for random radius.

```
void CMainFrame:: OnLButtonDown(UINT nFlags, CPoint pt

                        int rp;
                  do{
                  rp =rand();
                  }while(rp<49 || rp > 150);
```

```
            rpx[counter] = rp;
```

The next codes are written to calculate the routing probability in ad hoc networks. Equation 4 must be converted to Microsoft Visual C++ format. Furthermore, these properties can also be used to print out the result.

```
void CMainFrame::OnCompute(){
    CClientDC dc(this);
    CClientDC dc1(this);
    CPen penGray(PS_SOLID,2,RGB(200,200,200));
    dc.SelectObject(penGray);
    CPen penG(PS_SOLID,1,RGB(0,0,0));

        CFont fontCourier1;
    fontCourier1.CreatePointFont (50,"Arial");
    dc1.SetTextColor(RGB(250,250,250));
    dc1.SetBkColor(RGB(0,0,0));
    dc1.SelectStockObject(HOLLOW_BRUSH);
    dc1.SelectObject(penG);

    CString s;
    CRect rc;
    for (int i=1;i<=counter-1;i++){
        int mx,my, counterX;
        mx = mPt[i].x;
        my = mPt[i].y;
        counterX = 0;
        for (int j=1;j<=counter;j++){
            int mx1,my1,x1,x2,lTr ,xd , yd;
            double xp1,xp2,d , a1, a2, b1, b2, ax,py;
            int m = 1;
                if(i == counter){
                    lTr = 1;
                } else if(i == j){
                    continue;
                    }else if(j == counter){
                        lTr =1;
                    }else{
                        lTr =j+1;
                    }
                mx1 = mPt[lTr].x;
                my1 = mPt[lTr].y;
```

```
            xd = mx1 - mx;
            yd = my1 - my;

      d = sqrt(pow((double)(xd),2) + pow((double)(yd),2));
            x1 = rpx[i];
            x2 = rpx[lTr];

p1 = (pow(d,2)+pow((double)(x1),2)-pow((double)(x2),2)) / (2*d);

xp2 = (pow(d,2)-pow((double)(x1),2)+pow((double)(x2),2)) / (2*d);

a1 = pow((double)(x1),2) * acos(xp1/(double)(x1));

a2 = pow((double)(x2),2)* acos(xp2/(double)(x2));

b1 = xp1*(sqrt(pow((double)(x1),2) - pow(xp1,2)));

b2 = xp2*(sqrt(pow((double)(x2),2) - pow(xp2,2)));

            ax = a1+a2-b1-b2;
            py = ax / (PI*pow((double)(x2),2));

            m1 = mPt[i];
            m2 = mPt[lTr];

               if(py == 0){}
               else if(py > 0 && py < 1){
               dc.MoveTo(m1); dc.LineTo(m2);
               counterX++;
                            }
               else if(py == 1){
               dc.MoveTo(m1); dc.LineTo(m2);
               counterX++;
                            }
         }
            if(counterX > 0 ){
            dc1.TextOut(810,100+i*30,"Node Available");
            }
            else if(counterX == 0){
            dc1.TextOut(810,100+i*30,"Not Available");
            }
            else if(counterX == 1){
      dc1.TextOut(810,100+i*30,"link are definite
      exist");
      }
      else{
      dc1.TextOut(820,100+i*30,"Cannot Define nodes");
      }
   }
}
```

The Output

This section displays how the interface of the program looks like and also displays some outputs.

Interface

Figure 2 shows the interface of the program.

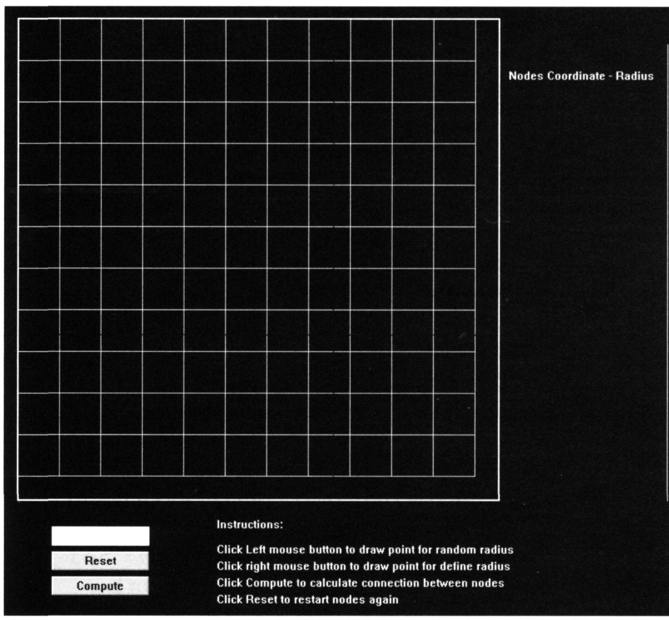

Figure 2: Routing Probability in Ad Hoc Networks Simulation Program Interface

Some Outputs

Figure 3 shows the output displayed in the interface when a user inputs a radius of 80 and clicks the 'Compute' button. Figure 4 shows the output for random radius (49 < radius < 150).

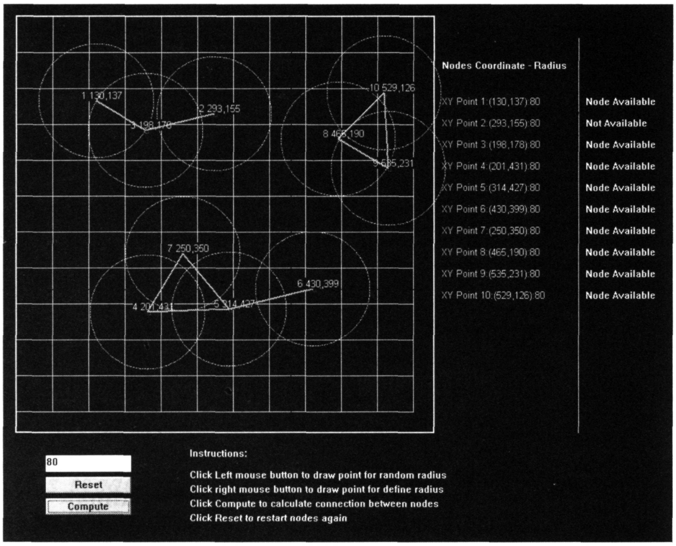

Figure 3: Sample Simulation Output at Radius of 80

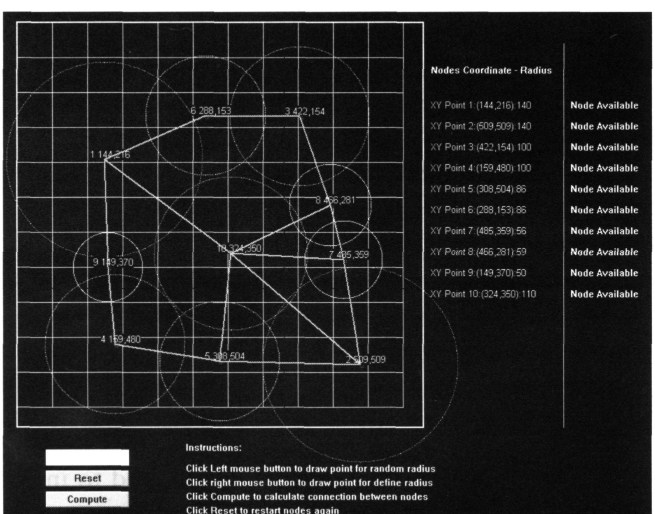

Figure 4: Simulation Output for Random Radius (49 < radius < 150)

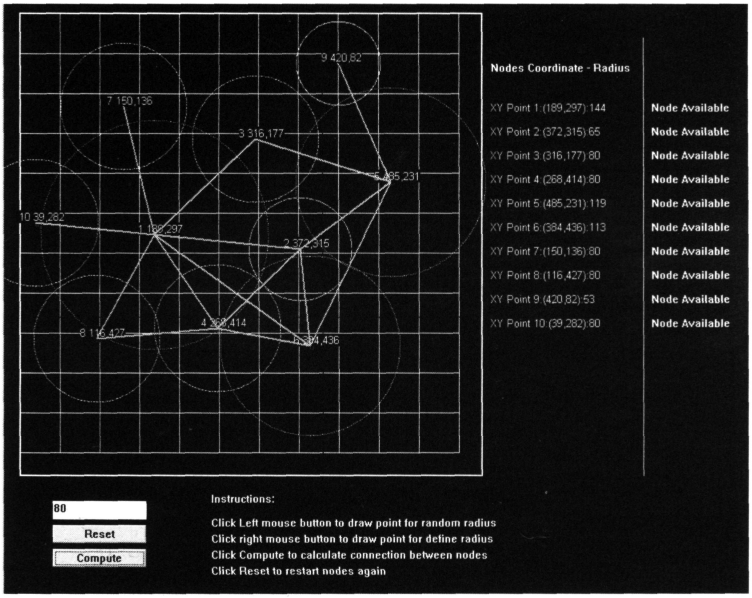

Figure 5: Simulation Output for Random and Defined Radius

Figure 5 shows the output for random and defined radius.

Conclusion

This program demonstrates how wireless radio link existence probability can be determined from location attributes, transmission radius, and node's velocity for any two mobile nodes. Furthermore, we determine the probability of the route availability that takes into account of the node movement and dependencies between links, and have been derived in a simple closed expression.

Simulation tests are evaluated in order to demonstrate the performance of the predicted route with respect to various schemes and parameters in mobile ad hoc networks.

Suggestions

Further work for simulation and program is required to include other measures to each route such as bandwidth, delay, and others in determining the probability of route existence. Simulation test is very important to be evaluated in order to demonstrate the performance of the predicted route with respect to various schemes and parameters in mobile ad hoc networks.

References

Jones, R. M. (1999). *Introduction to MFC programming with Visual C++*. New Jersey: Prentice Hall.

Muhammad Hisyam Lee, Shaharuddin Salleh, & Stephan Oalriu. (2001, September). *Computation of the availability of a route in a simple closed form expression for mobile ad hoc network*. Paper presented at the 7th Asia Pacific Conferences on Communications (APCC2001), Tokyo, Japan.

Sakhinah Abu Bakar, Shaharuddin Salleh, & Muhammad Hisyam Lee. (2003). FRECAST: Message transmission model for dense graphs in ad hoc networks. *International Journal of Simulation and Process Modeling*, *2*(1/2), 4–11.

Salleh, S. (2003). *SSM3323 lecture notes: C++ programming with MFC*. Malaysia: Universiti Teknologi Malaysia.

AHMAD ZIA UL-SAUFIE MOHAMAD JAPERI, MUHAMMAD HISYAM LEE, & SHAHARUDDIN SALLEH, Department of Mathematics, Faculty of Science, Universiti Teknologi Malaysia, 81310 UTM Skudai, Johor, MALAYSIA. E-mail: dzia_math@yahoo.com, hisyam@mel.fs.utm.my, ss@mel.fs.utm.my