Broadcast algorithms in SINR model with random noise

June 17, 2012

Chapter 1

Introduction

Chapter 2

Model

2.1 Signal-to-interference ratio

$$\frac{\frac{P}{d^{\alpha}}}{I+R} \tag{2.1}$$

2.2 Random noise

In the real world the noise value is not constant. It depends on many physical factors like weather or even cosmic noise. Using random N values approximating real noise seems to be a good idea.

Random noise can be modelled quite as a Generalized Extreme Value [2] distribution with some parameters μ, σ, ξ. We assume $\xi \neq 0$. Cumulative distribution function for GEV(μ, σ, ξ) distribution is:

$$\mathbf{P}(N \leq x) = e^{-(1+\xi\frac{x-\mu}{\sigma})^{-\frac{1}{\xi}}} \tag{2.2}$$

The values of ξ, μ and σ can be found experimentally.

2.3 Range

We define transmission range r_i of a node i as a maximum distance at which the node can succesfully transmit its signal when there is no interference other than background noise. This is easy to obtain from (2.1):

$$\beta = \frac{\frac{P_i}{r_i^{\alpha}}}{0+N}$$

$$
\begin{aligned}
\beta N &= \frac{P_i}{r_i^\alpha} \\
r_i^\alpha &= \frac{P_i}{\beta N} \\
r_i &= \left(\frac{P_i}{\beta N}\right)^{\frac{1}{\alpha}}
\end{aligned}
$$

Considering random noise, transmission range of a node can vary due to current noise. We define probable transmission range $\tilde{r}_i$ as a distance at which we have 0.95 probability of a successful transmission. We have:

$$
\begin{aligned}
\mathbf{P}\left(\beta \leq \frac{\frac{P_i}{\tilde{r}_i^\alpha}}{0 + N}\right) &= 0.95 \\
\mathbf{P}\left(N \leq \frac{P_i}{\beta \tilde{r}_i^\alpha}\right) &= 0.95
\end{aligned}
$$

and applying (2.2) we have:

$$
\begin{aligned}
e^{-\left(1+\xi\frac{\frac{P_i}{\beta\tilde{r}_i^\alpha}-\mu}{\sigma}\right)^{-\frac{1}{\xi}}} &= 0.95 \\
\left(1+\xi\frac{\frac{P_i}{\beta\tilde{r}_i^\alpha}-\mu}{\sigma}\right)^{-\frac{1}{\xi}} &= (-\ln 0.95) \\
\xi\frac{\frac{P_i}{\beta\tilde{r}_i^\alpha}-\mu}{\sigma} &= (-\ln 0.95)^{-\xi} - 1 \\
\frac{P_i}{\beta\tilde{r}_i^\alpha} - \mu &= \frac{\sigma(-\ln 0.95)^{-\xi} - 1}{\xi} \\
\frac{1}{\tilde{r}_i^\alpha} &= \frac{\beta}{P_i}\left(\frac{\sigma(-\ln 0.95)^{-\xi} - 1}{\xi} + \mu\right) \\
\tilde{r}_i^\alpha &= \left[\frac{\beta}{P_i}\left(\frac{\sigma(-\ln 0.95)^{-\xi} - 1}{\xi} + \mu\right)\right]^{-1} \\
\tilde{r}_i &= \left[\frac{\beta}{P_i}\left(\frac{\sigma(-\ln 0.95)^{-\xi} - 1}{\xi} + \mu\right)\right]^{-\frac{1}{\alpha}} \qquad (2.3)
\end{aligned}
$$

Chapter 3

Network graph

Network graph is a undirected graph $G = (V, E)$, where $V = \{v_1, \dots, v_n\}$ is a set of nodes (points) and $E \in V^2$ is a set of links between nodes. For each node v_i we define $E(v_i)$ to be a set of v_i neighbors, ie. $E(v_i) = \{v_j \in V : (v_i, v_j) \in E\}$.

We used three methods for generating network graphs:

1. uniform networks,
2. social networks,
3. gadget networks

Each generated network is strongly connected.

3.1 Uniform networks

Uniforms networks are random graphs generated using a uniform distribution. Each node has a position in $[0, S] \times [0, S]$ square.

3.2 Social networks

Social networks tries to model human behaviour of connecting into larger group. Nodes are generated within a square $[0, S] \times [0, S]$. The square is divided into subsquares of size ϵ. We define $s_{i,j}$ to be a square in i-th row and j-th column. To a subsquare $s_{i,j}$ we assign a weight:

$$w_{i,j} = \left| \bigcup_{v \in s_{i,j}} E(E(v)) \right|$$

With probability $1 - \gamma$ we choose a subsquare s according to the weights, and put new node with random position (using uniform distribution) within the choosen subsquare. With probablity γ we use uniform network algorithm to generate new node.

3.3 Gadget networks

Gadget network is a special class of network designed to be “hard” for algorithms.

Chapter 4

Algorithms

Bibliography

[1] Olga Goussevskaia , Yvonne-Anne Pignolet, Roger Wattenhofer, *Efficiency of Wireless Networks: Approximation Algorithms for the Physical Interference Model*, 2010

[2] Xu Su, Rajendra V. Boppana, *On the impact of noise on mobile ad hoc networks*, 2007, ACM