HDEER: A Distributed Routing Scheme for Energy-Efficient Networking

MODULE 1:

Wireless mobile topology of creation of simple packet transmission between nodes with default node configurations.

Flow of Implementation:

TCL Script, Default configurations of wireless,SPT protocol, NAM window.

EXISTING MECHANISM (PAPERS EXISTING METHOD)

MODULE 2:

Wireless mobile topology of creation of more number of nodes [50 nodes] complex topology and packet transmission will be done based on NORMAL SHORTEST PATH ALGORITHM TECHNIQUE [SPT PROTOCOL] and QOS performance metrics like end to end delay, energy spent, packet delivery ratio, throughput values are taken and graphs will be plotted in xgraph.

Flow of Implementation:

TCL Script, Default configurations of wireless ,SPT protocol,NAM window,awk file execution,graph plot.

PROPOSED MECHANISM (PAPERS PROPOSED METHOD)

MODULE 3:

Wireless mobile topology of creation of more number of nodes [50 nodes] complex topology and packet transmission will be done based on PROPOSED DISTRIBUTED ROUTING SCHERME FOR ENERGY EFFICIENT ROUTING TECHNIQUE (Please provide a name for the protocol) MOBILE COORDINATED WIRELESS SENSOR NETWORK SCHEME) HDEER PROTOCOL is suggested which is developed in c++ and

integrated in to NS2 package and Algorithms will be written as procedures in tcl scrpti

1.D_Routing

2.Routing in a static traffic scenario

3.Routing in a dynamic traffic scenario

QOS performance metrics like end to end delay, energy spent, packet delivery ratio, throughput values are taken and graphs will be plotted in xgraph.

Flow of Implementation:

TCL Script, Default configurations of wireless,Procedure for three algorithms,HDEER protocol,NAM window,awk file execution,graph plot.

MODULE 4:

Comparison of the existing NORMAL SHORTEST PATH ALGORITHM TECHNIQUE (NORMAL SPT PROTOCOL) and proposed (HDEER PROTOCOL) DISTRIBUTED ROUTING SCHERME FOR ENERGY EFFICIENT ROUTING TECHNIQUE with single trace file and graphs execution.

Flow of Implementation:

User generated trace files,graph plot.

NOTE:

SOFTWARES USED : REDHAT LINUX 9

Front End : TCL

Back End : C++

We can do the same in static and dynamic topology. Please confirm with the student as i gave only dynamic topology.

Enhancement (New work with the paper) has not given in the module break up. If the student has any idea on the same can be done else will suggest once we completed the paper work.