ANOVA-INFORMED DECISION TREES FOR VOICE APPLICATIONS OVER MANETS*

Mouna Benaissa,[1] Vincent Lecuire,[1] D.W. McClary,[2] and Violet R. Syrotiuk[2]

[1] *CRAN, CNRS UMR 7039*
Henri Poincare-Nancy I University
Campus Scientifique, BP 239
54506 Vandoeuvre-les-Nancy Cedex, France
{mouna.benaissa,vincent.lecuire}@cran.uhp-nancy.fr

[2] *Department of Computer Science & Engineering*
Arizona State University
P.O. Box 878809
Tempe, AZ U.S.A. 85287-8809
{daniel.mcclary,syrotiuk}@asu.edu

Abstract   Both real-time multimedia and mobile networks present challenges ripe for new analysis techniques. We examine the applicability of statistical design of experiments and inductive learning theory in the prediction of delay for real-time audio transmissions over mobile ad hoc networks. Utilizing analysis of variance methods and simple decision tree agents, we find both significant factor interaction between traffic load and node mobility as well as a dramatic reduction in error percentage in prediction of end-to-end delay.

1.   Introduction

Real-time multimedia transmissions and mobile ad hoc networks each provide distinct challenges. Jitter and end-to-end delay are significant factors of interest for multimedia transmissions. Likewise, end-to-end delay is often a primary concern within *mobile ad hoc networks* (MANETs). When these fields meet, the importance of better understanding the domains increases significantly.

The reliability and consistency required by real-time audio transmission becomes much harder to ensure when dealing with the constantly varying condi-

---

tions of a mobile ad hoc network. Thus, there is a need to effectively understand and predict end-to-end delay.

In light of this need, we show the validity of techniques from outside the domain of common networking. Specifically, we examine *statistical design of experiments* (DoE) and *analysis of variance* (ANOVA) to isolate and quantify factor interactions among variables in a MANET. Additionally, we adapt artificial intelligence techniques from basic decision-tree learning to provide a mechanism for predicting delay within a mobile environment. By fusing these two techniques, we find significant measures of predictability for end-to-end delay for audio transmissions over MANETs.

We first present a performance analysis of end-to-end delay for audio transmissions over MANETs followed by an ANOVA analysis for accurate prediction of end-to-end delay. We begin with a discussion of related works and an introduction to the simulation methods used. From this, we present the simulation results followed by an introduction to DoE and ANOVA analysis, learning theory and decision trees, and the results produced by these techniques.

2.   Simulation Analysis of Audio Packet Delays

The results in this section are based on simulations using the network simulator ns-2, version 2.26 [6, 9]. This package includes extensions for mobile ad hoc network simulation, including a set of routing protocols, an IEEE 802.11 MAC layer, a radio propagation model, and a node mobility model. However, we extended this package to generate voice traffic over the wireless environment.

Technical Considerations

All nodes communicate with a wireless radio based on the IEEE 802.11 standard [7]. The radio propagation range for all nodes is 250 meters and the channel bandwidth is $11 Mbps$. The specific *medium access control* (MAC) scheme is CSMA/CA with acknowledgments. At the link layer, we leave most of the 802.11 parameters set to default values. Thus, the RTS threshold is set to 250, the short retry limit (SRL) is set to 7, and long retry limit (LRL) is set to 4.

At the network layer, we use the *Ad hoc On Demand Distance Vector* (AODV) [13] protocol for routing. The choice of a reactive protocol such as AODV rather than a proactive protocol such as OLSR [11] is based on results from our earlier work [3]. Indeed, the proactive protocol consists of every node emitting `hello` messages periodically in order to learn the network topology. On the other hand, reactive protocols invoke a route discovery procedure on an on-demand basis.

As mobile ad hoc networks are characterized by intermittent connectivity, the audio stream is interrupted when a route error causes the routing protocol to establish a new route to the destination. These interruptions are typical events that strongly disrupt the speech played on the receiver side. Benaissa et al. [3] have shown that such interruptions can be long, typically a number of seconds, regardless the routing protocol employed. However, OLSR causes more interruptions than AODV, thus AODV is better adapted to the deployment of audio applications over ad hoc networks.

The AODV parameter values are set as recommended in [14] such that they minimize network congestion and allow the protocol to operate as quickly and accurately as possible. As such, the HELLO interval is set to 1.0, the route reply wait time is set to 1.0, the reverse route life is set to 3.0, and the active route timeout is set to 3.0.

The network layer maintains a send buffer of 64 packets. This buffer contains (only) data packets waiting for a route. All packets (both data and control) sent by the routing layer are queued at the interface queue until MAC layer transmission. We set the maximum size of the interface queue to 50 packets and maintain it as a priority queue with two priorities, each served FIFO. Control packets receive higher priority than data packets.

Network Environment and Methodology

Our network model consists of 50 nodes in a 1000×1000 meter square flat area. Our results are based on the average packet delay of 20 scenarios, or patterns. Patterns are randomly generated by different seeds. Each simulation executes for $250s$. Input parameters for the simulation are speed s in m/s and the network load ℓ in kbps.

In order to avoid large variation in successive patterns, some seed number effects are cancelled. We make the following assumptions:

- We define the same initial position and heading of nodes for all patterns. Thus, when the seed changes, the initial network topology and traffic peers remain identical. All movements are different (see Figure 1).

- For different patterns at a given network load ℓ, enabled traffic peers are identical. When ℓ is increased, additional traffic peers are set.

- During the simulation, the effective speed of nodes and the network load are constant and equal to the input parameters s and ℓ.

Mobility

Nodes move according to the random waypoint mobility model. In this model, each node x chooses a random destination within the simulation area
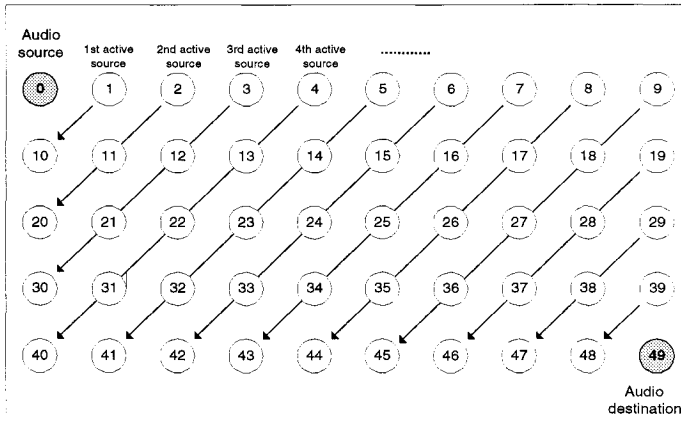
Figure 1.   The initial MANET topology.

and travels toward its destination in a straight line at a given speed. Upon node x's arrival at its destination, it pauses, chooses a new random destination, and continues its motion. Node speed s varies from $1m/s$ to $10m/s$ and we use a pause time of 10 seconds.

Traffic Pattern

We develop a traffic generator to simulate unicast voice traffic as well as background traffic.

Voice traffic is generated between a source node (0) and a destination node (49) (see Figure 1). A voice flow is typically divided into *talkspurts* (periods of audio activity) and *silent periods* (periods of audio inactivity, during which no audio packets are generated). We consider an average talkspurt of 30.83% and an average silent period of 61.47% as recommended by the ITU-T specification for conversational speech [10]. Alternating periods of activity and silence are exponentially distributed with an average duration of $1.004s$ and $1.587s$, respectively. We consider the PCM codec (Pulse Codec Modulation — see Recommendation G.711 in [8]) at $64kbps$ as audio traffic. Then, 320 byte voice packets are generated periodically during activity periods, each voice packet representing a sample of $40ms$.

Background traffic is generated by *constant bit rate* (CBR) data sessions with selected sources and destinations. One CBR session transmits packets of 2048 bytes payload every $0.3s$ (i.e., $57.28kbps$ load per session). To increase

the load in the network, the number of active sessions is increased. In our simulation, network load ℓ varies from $1 \times 57.28kbps$ to $10 \times 57.28kbps$.

Simulation Results

Figure 2 illustrates average end-to-end delay as a function of the number of active sessions for three different node speeds: $2m/s$, $5m/s$ and $7m/s$. We find that from 1 to 5 active sessions, delays are nearly equal (less than $40ms$), regardless of the speed. For higher numbers of active sources, the increases in delay are stronger when node speed is higher. Delay increases from $50ms$ to $200ms$ for node speed $2m/s$, while it reaches $450ms$ at node speed $7m/s$ under the same load level.
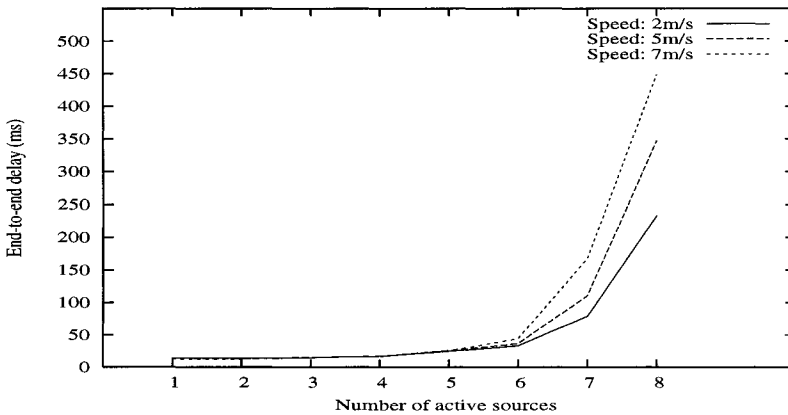

Figure 2. End-to-end delay as a function of load for different node speeds.

Our results illustrate that node speed does not have a significant impact on end-to-end delay under light load traffic. However, when load traffic increases, end-to-end delay increases rapidly when node speed is high.

3. Designed Experiments and ANOVA Analysis

Though often treated as isolated, the factors affecting a transmission in any network interact in ways that are often imperceptible when considered by system and protocol designers. Factor interaction is not a new consideration in other engineering domains and techniques have long been in place to effectively unearth and analyze them. Designed experiments, a technique originating from agricultural engineering in the early part of the 20th century, allows

the otherwise vast number of experimental runs necessary to detect and quantify factor interactions to be done in a limited number of trials. As defined by Montgomery [12], a *factor* (e.g., load) is an experiment variable taken to have some effect on the *response variable* of interest (e.g., end-to-end delay). A *factor interaction* is the failure of a factor to produce the same resultant value for the response variable when another factor is at a different level.

Key to DoE's ability to effectively analyze factor interactions is analysis of variance (ANOVA). In the simplest case, ANOVA examines the effect of a single factor on a response variable. ANOVA is defined by a sum of squares identity [12]. Using these sums of squares, ANOVA allows calculation of main effects for all factors as well as interactions. Our aim is to utilize DoE and the underlying foundation provided by ANOVA to significantly improve the predictability of end-to-end delay for audio-transmissions over MANETs.

Related Work — DoE and ANOVA

Designed experiments and ANOVA have rarely been applied to mobile networking. Vadde and Syrotiuk [17], inspired by the work of Barrett et al. [1, 2], used ANOVA analysis to identify main effects and factor interactions on service delivery in MANETs. Along similar lines, Perkins et al. [15] examined 2^k factor interaction particularly among node speed, network size, and number of traffic sources.

4.   Learning Theory and Decision Trees

Machine learning theory aims to develop agents and algorithms that are able to effectively improve future action by learning from both its environment and its own decision-making processes. Among the simplest learning algorithms to implement is that of *decision tree learning*. Decision tree learning examines a set of properties and returns a decision. Typically decisions are boolean, but are easily extended to numerical results. Thus, decision trees are naturally suited to learning parameter optimization.

In functionality, the inductive learning provided by a decision tree begins with a heuristic from which an initial decision tree structure is extrapolated. The branching conditions of this tree are then tuned via the inductive process to produce better optimized results. Thus, for the purpose of optimizing the predictability of end-to-end delay, decision trees allow us to quickly produce a proof-of-concept argument for inductive learning theory for MANETs as well to evaluate the potential benefit of integrating a decision mechanism into our simulation model.

Related Work – Learning Theory and Decision Trees

Within artificial intelligence and other fields, learning theory has evolved into a sophisticated and rich field in its own right. However, much in the same manner as ANOVA analysis, learning theory methodologies have not been significantly explored within mobile ad hoc networks. Farago et al. [5] developed a meta-MAC protocol utilizing computational learning theory. We take our direction in learning agent construction from Russell and Norvig [16].

5.   DoE and Learning Methodologies

DoE and ANOVA Methodologies

In order to validate the effectiveness of DoE and ANOVA techniques within our chosen application domain, we choose those variables from Benaissa et al. [3] that provide both the most sound basis for factor interactions as well as reasonable extension into a decision tree learning mechanism. We choose CBR traffic load and node mobility as our factors of interest and end-to-end delay as our response variable. We then use `Design Expert` [4], a software suite designed to aid in the construction and processing of DoE, to develop an appropriate model to fit our data set.

We design our statistical experiment as a 2^2 response surface populated by our data points, where 2^k is indicative of a two-level (high and low) model with k variables of interest or *main effects*. From this, we generate two models for ANOVA analysis. The first, an untransformed model of the data, is used to develop certain of our decision tree agents. The second, a model transformed such that any factor interactions present are effectively quantified, is used to verify the ANOVA validity.
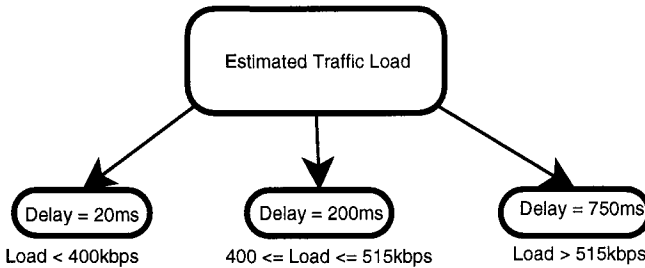
Decision Tree Methodologies

To leverage learning theory techniques, we choose decision tree inductive learning for a number of reasons: ease of implementation, small spatial requirements and a solid modeling of a top-down traversal of protocol stacks. For our proof-of-concept, we examine multiple means of constructing the inductive learning aspects of the tree, both with and without statistical influence. As is standard for decision tree tests, average-guessing or *strawman* algorithms are constructed to provide a baseline for judgment of improvements via the learning mechanism. Given our implementations of both statistically-influenced and standard inductive learning, we implement both statistically-influenced and standard mean-guessing strawman algorithms.

Our first step in developing decision tree agents is to create a heuristic that serves as the basis for construction of the initial tree. For purposes of conceptual proof, we do not consider factor interactions in the construction of this

heuristic, though such interactions are responsible for much of the power of this manner of inductive learning. Instead, we consider the simplest case, in which we assume end-to-end delay is primarily dependent on traffic load. Examination of the data set yields a decision tree with three distinctly different relations to end-to-end delay. As illustrated in Figure 3, we choose a simple, one-level tree branching at what initial observation indicates are significant differences in the load vs. end-to-end delay relationship.

Figure 3.   A simple one-level decision tree estimates delays based on noticeable variances of load vs. mobility. More refined decision trees take into account multiple factors and their interaction.

Having constructed an initial tree, we implement the inductive learning process as a number of agents, each utilizing different means of hypothesis correction during the learning and testing phases. These agents can be broken into two sets — those that do not use statistical influence in their correction, and those that do. For each of these categories, we create agents which employ incremental correction techniques and agents that employ averaging. Additionally, we design an agent that uses no correction following its learning phase, as well as one that employs only statistical means of self-correction.

Agent correction in both learning and testing phases is structured such that those agents without statistical influence utilize their particular means of correction, be it incremental or average-based, from the point of the value of the initial heuristic. Statistically-influenced agents utilize averaging based on the ANOVA-generating equation during the learning phase and their particular correction method during the testing phase to further refine the agent's accuracy. The agent with no testing-phase adaptation employs averaging in its learning-phase correction, and the completely statistically-driven agent averages against the ANOVA-generated equation during both phases.

6.   DoE and Learning Theory Results and Discussion

DoE Results and Discussion

As described in section 5, we create a pair of designed experiments. One uses no data transformation so as to yield equations suitable for adaptation into our decision tree agents and the other transformed to most accurately assess any factor interaction present. In each case, we utilize `Design Expert` to process our data sets. We specify our factors as L for traffic load, and M for node mobility, with respective low and high levels ($57.28kbps$, $572.80kbps$) and ($1m/s$, $8m/s$). We then choose as our response variable for end-to-end delay for the real time audio flow with low and high levels ($4.10ms$, $1663.96ms$).
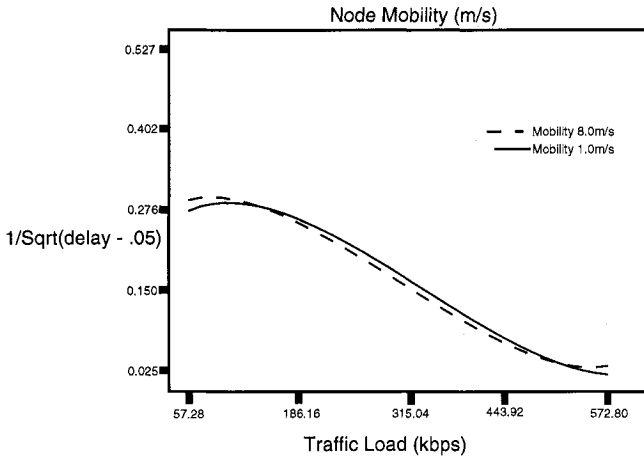
From initial sum of squares analysis, we choose to design the experiments as a cubic model analyzing factors L, M, L^2, M^2, LM, L^3, M^3, L^2M, and LM^2 where, e.g., LM denotes the interaction effect between traffic load and node mobility. In keeping with standard practice, we choose the cubic model because it yields the lowest F-value and $P(X > \text{F-value})$. On this cubic model we then use `Design Expert` to perform ANOVA analysis of the data.

As expected, the initial untransformed model uncovers significant factor interaction. We find significant factors to be L, M, L^2, LM, L^3, and L^2M. Thus, consistent with our hypothesis, the interaction between traffic load and node mobility within MANETs is significant with respect to end-to-end delay on real-time audio transmissions. Likewise, in this case the square and cube of traffic load holds some significant relationship to end-to-end delay. For the purposes of enhancing our decision tree agents, ANOVA yields the relationship for the delay, D:

$$
\begin{aligned}
D = {} & 70.40 + 0.17L - 52.54M - 3.3e^{-3}L^2 + 13.82M^2 + \\
& 0.024LM + 9.85e^{-6}L^3 - 1.17M^3 - 2.40e^{-4}L^2M + 0.01LM^2.
\end{aligned}
$$

By using a Box Cox test [12], we determine that by applying an inverse square root transform ($y' = \frac{1}{\sqrt{y+\lambda}}$) with $\lambda = -0.5$. we achieve a more precise measurement of the principal factor interaction, that between traffic load and node mobility. As shown in Figure 4, we find that the two factors interact with one another in at least two key points. From the generated interaction plots as well as the lack of fitting error, there exists a definite factor interaction between traffic load and node mobility for audio transmission over MANETs. Specifically, we notice significant interactions when traffic load is in the ranges $114.56 < L < 171.84$ and $458.4 < L < 515.52$. Thus, we say load-mobility interaction displays its most prominent effects when traffic load is in the more extreme regions of its possible values.

Figure 4.   A factor interaction graph for traffic load and node mobility. Path intersections indicate levels at which interaction is most significant.

Decision Tree Results and Discussion

For testing of both the validity of decision tree agents in reducing uncertainty in end-to-end delay in MANETs, as well as the integration of the technique with statistically-influenced heuristics, we measure the testing-phase performance of our previously designed agents in three categories over n estimates:

- Correct guess percentage: $\frac{\text{correct estimates}}{n}(100)\%$

- Mean error percentage: $\frac{|\text{delay}_{\text{estimated}} - \text{delay}_{\text{actual}}|}{n}(100)\%$

In order to establish a baseline for evaluating our agents, we run trials with both a "dead" strawman, which simply guesses the mean delay found in the learning phase for all testing-phase estimates, and a statistical strawman, which guesses the result of our ANOVA equation for each testing-phase estimate. The "dead" strawman produces correct guess percentage of 4.25%, and a mean error of 572.10%. With similar inaccuracy, our statistical strawman reports a correct guess percentage of 2.38% and mean error percentage of 1010.78%. Additionally, we consider our non-adapting decision tree agent, which produces results: correct guess percentage of 4.40%, and a mean error of 140.25%.

These baselines established, we find that both categories of decision-tree agents perform well, with certain techniques outshining others. The decision tree agents that adapt incrementally both perform with correct percentages of

15.14% in the statistical case and 19.65% for the standard model. In our normal and statistically-influenced agents, we find that those that employ testing-phase heuristic refinement via averaging produce correct percentages of 33.04% and 32.69% while maintaining mean error percentages less than 100%. Specifically, the statistically-influenced agent with difference-based testing phase adaptation produces a correct percentage of 32.69% while lowering mean error percentage to 42.00%. In similar fashion, our non-statistical model produces a correct percentage of 33.04% and mean error percentage of 39.84%.

In consideration of these results, we find that decision tree agents that utilize difference-based averaging perform with dramatic improvement over simple averaging means. Additionally, the lack of adverse effects produced when utilizing ANOVA-based heuristics leads us to believe that ANOVA-produced equations provide just as sound if not more sound a basis for decision heuristics as currently exist. That said, given the significance of the factor interaction detected by the DoE phase of our work, we believe that this interaction can be exploited to produce far more accurate decision trees than our current one-level model. In such a case, we expect the learning to be far richer and the results even more promising.

7.   Conclusions and Future Work

Our results in examination of real-time audio transmission over mobile ad hoc networks serve as an initial proof-of-concept for the validity of both DoE as a means of effectively verifying the results of the simulation, and detecting factor interaction among network parameters as well as decision tree learning as a method of increasing the predictability of end-to-end delay. The work is currently in a preliminary state. Further investigation of the interaction between traffic load and mobility is certainly required. Likewise, we expect deeper consideration and refinement of the decision tree model, based on the results of our ANOVA analysis of the data, will improve even present levels of predictability and error percentage. In the same vein, consideration of other factors involved in real-time audio transmission, particularly routing and MAC protocol configurations (e.g., RTS threshold, short and long retry length) may yield yet more information about the complex interactions between these components. We plan to embed such a learning mechanism within an application, such that delay may be minimized on the fly. As such, we have begun consideration of these directions and expanding our research as the domain demands.

References

[1] Barrett, C.L, A. Marathe, M.V. Marathe, and M. Drozda (2002). "Characterizing the Interaction Between Routing and MAC Protocols in Ad-Hoc Networks," Proceedings of the Third ACM International Symposium on Mobile Ad Hoc Networking and Computing (MobiHoc'02), pp. 92–103.

[2] Barrett, C.L., M. Drozda, A. Marathe, and M.V. Marathe (2003). "Analyzing Interaction Between Network Protocols, Topology and Traffic in Wireless Radio Networks," Proceedings of the IEEE Wireless Communications Networking Conference (WCNC'03)", pp. 1760–1766.

[3] Benaissa M., V. Lecuire, F. LePage, and A. Schaff (2003). "Analysing End-to-End Packet Delay and Loss in Mobile Ad hoc Networks for Interactive Audio Applications," Proceedings of the Workshop on Mobile Ad Hoc Networking and Computing (MADNET'03), pp. 27–33.

[4] "Design Expert Software," Stat Ease Inc.
    lOnlinel. Available: http://www.statease.com

[5] Farago A., A.D. Myers, V.R. Syrotiuk, and G.V. Zaruba (2000). "Meta-MAC Protocols: Automatic Combinations of MAC Protocols to Optimize Performance for Unknown Conditions," IEEE Journal on Selected Areas in Communications, Vol. 18, No. 9, Sept. 2000, pp. 1670–1681.

[6] Greis, M. (2001). "Tutorial for the Network Simulator ns,"
    lOnlinel Available: http://www.isi.edu/nsnam/ns/tutorial

[7] IEEE Standards Department (1999). "Part 11: Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) specifications: Higher-Speed Physical Layer Extension in the 2.4 GHz Band," Technical Report, European Telecommunications Standards Institute.

[8] International Telecommunication Union ITU-T (2001). "Pulse Code Modulation (PCM) of Voice Frequencies," Recommendation G.711, Telecommunication Standardization Sector of ITU.

[9] ISI (2001). "The Network Simulator ns,"
    lOnlinel Available: www.isi.edu/nsnam/ns/

[10] ITU-T (2004). "List of ITU-T Recommendations,"
     lOnlinel Available: http://www.itu.int/publications/itu-t/itutrec.htm

[11] Jaquet, P., P. Muhletaler, A. Quayyum, A. Laouiti, T. Clausen, L. Viennot, and P. Minet (2002). "Optimized Link State Routing Protocol," Internet Draft draft-ietf-manet-olsr-06.tx, Internet Engineering Task Force.

[12] Montgomery, D. (2001). *Design and Analysis of Experiments*, John Wiley & Sons, Inc.

[13] Perkins, C.E. (2001). *Ad hoc Networking*, Addison Wesley Longman, Inc.

[14] Perkins, C.E. and E. Royer (1999). "Ad hoc On-demand Distance Vector AODV Routing," Proceedings of the IEEE Workshop on Mobile Computing System and Applications, pp. 90–100.

[15] Perkins, D.P., H.D. Hughes, and C.B. Owen (2002). "Factors Affecting the Performance of Ad Hoc Networks," Proceedings of IEEE International Conference on Communications (ICC'02), Vol. 4, pp. 2048–2052.

[16] Russell, S. and P. Norvig (1995). *Artificial Intelligence: A Modern Approach*, Prentice Hall, Inc.

[17] Vadde, K.K and V.R. Syrotiuk (2004). "Factor Interaction on Service Delivery in Mobile Ad Hoc Networks," to appear in IEEE Journal on Selected Areas in Communications, (accepted March 2004).