

SIGNAL PROCESSING FOR WIRELESS AD HOC COMMUNICATION NETWORKS

© IMAGESTATE

[Linda E. Doyle, Anil C. Kokaram,
Senan J. Doyle, and Timothy K. Forde]

Ad Hoc Networking, Markov Random Fields, and Decision Making

[The challenges involved]

In a very general sense, an ad hoc network can be considered to be a collection of wireless mobile nodes that dynamically form a temporary network without the use of any existing network infrastructure. The resulting network is a nonhierarchical distributed structure. All nodes of the network act as routers and forward received packets to nodes within radio range. Ad hoc networks, by their nature, are highly adaptive systems that can come into existence on an as-needed basis. They can grow, reduce in size, fragment, and dismantle as desired. In this article, we focus on mobile ad hoc networks (MANETs). As such, we are interested in ad hoc nodes that are mobile and highly functioning.

As communication systems get ever more sophisticated, we can look to a future of highly self-organizing networks that can not only form networks as needed but can also reconfigure and adapt to prevailing network conditions. Such networks will

leverage, and make best use of, available node, network, and radio resources. The nodes of these networks will consist of both reconfigurable software and hardware components. Such entities will have the ability to change operating parameters to improve the performance of the network, to adapt to users needs, and to leverage business opportunities (e.g., reconfigure to avail of cheaper tariffs) among very many other capabilities. Viewed from this reconfigurable and adaptive perspective, *all* parameters associated with an ad hoc node, at all layers, will be treated as variables that are to be set and reset as required.

While a range of scenarios exist in which a node can adapt and reconfigure in a unilateral fashion, of interest here are those scenarios in which nodes in the network must make collective decisions. Such a distributed decision-making (distributed consensus) task is a challenging problem in highly dynamic MANETs and hence is the focus of this article. In

response to the challenge, we present a framework for distributed decision making that borrows heavily from the field of image processing and results in an elegant solution to the problem. Our central proposition is that by articulating decision making in an ad hoc network using a Markov random field (MRF) maximum a posteriori (MAP) approach, a viable decision-making framework can be designed.

This article explains why distributed decision making in ad hoc networks is so challenging. It describes the MRF-MAP decision-making framework and in doing so highlights the many insights gained from image processing. The framework is explored in the context of deciding between different ad hoc routing protocols.

THE CHALLENGES OF DISTRIBUTED DECISION MAKING IN AD HOC NETWORKS

Ad hoc networks are inherently self-organizing systems that meet spatially and temporally varying communication demands. Unlike static wired networks that are generally long lived, it is difficult to plan the configuration of a MANET in advance. As such, MANETs demand applications, services, and protocols that can manage the network without manual intervention and adapt to dynamic network conditions. Fundamentally, the ability of a network to self-organize rests on its ability to make decisions about the configuration choices available to it. Examples of self-organizing network-wide protocols are routing protocols [1], [2] and dynamic addressing schemes [3], [4], which eschew the use of centralized solutions.

Many existing decision-making algorithms make generous assumptions about the robustness of the communication model underpinning the decision-making solution, using a model formally described as the synchronous shared-memory model [5], [6]. Under this stable model, all nodes partaking in a decision are guaranteed to be capable of communicating with every other node within a defined time period. When this model is transposed into a network context, it demands system characteristics such as reliable multicasting and synchronized node clocks. However, a MANET would be more realistically modeled as a volatile asynchronous, message-passing system as the assumption of reliable multicasting and synchronized clocks is not practical.

Furthermore, the Fischer-Lynch-Patterson (FLP) [7] impossibility result clearly delineates the limits of deterministic decision making in systems of the kind characterized by open, mobile, multihop ad hoc networks. Fischer et al. have demonstrated that deterministic distributed decision making is impossible in an asynchronous message-passing system if even one node crashes. Essentially, nodes cannot place hard time bounds on the time taken for interactions with other nodes, making it impossible to distinguish between slow and crashed nodes. A

crashing node is the simplest form of system failure; a node can be perceived to have *crashed* in a mobile ad hoc networking environment due to the effects of both node mobility and signal quality on the status of node-to-node links. Nonetheless, the literature has shown that it is possible to circumvent the FLP result if some additional constraints are placed on the problem space. These fixes have generally involved limiting the scope of the decision-making process to the local neighborhood (as against the global network space) [8] and using soft-state techniques to terminate the global decision-making process in the absence of synchronized node clocks [3].

AD HOC NETWORKS ARE INHERENTLY SELF-ORGANIZING SYSTEMS THAT MEET SPATIALLY AND TEMPORALLY VARYING COMMUNICATION DEMANDS.

crashing node is the simplest form of system failure; a node can be perceived to have *crashed* in a mobile ad hoc networking environment due to the effects of both node mobility and signal quality on the status of node-to-node links. Nonetheless, the literature has shown that it is possible to circumvent the FLP result if some additional constraints are placed on the problem space. These fixes have generally involved limiting the scope of the decision-making process to the local neighborhood (as against the global network space) [8] and using soft-state techniques to terminate the global decision-making process in the absence of synchronized node clocks [3].

crashing node is the simplest form of system failure; a node can be perceived to have *crashed* in a mobile ad hoc networking environment due to the effects of both node mobility and signal quality on the status of node-to-node links. Nonetheless, the literature has shown that it is possible to circumvent the FLP result if some additional constraints are placed on the problem space. These fixes have generally involved limiting the scope of the decision-making process to the local neighborhood (as against the global network space) [8] and using soft-state techniques to terminate the global decision-making process in the absence of synchronized node clocks [3].

A SIGNAL PROCESSING FRAMEWORK FOR DECISION MAKING

The problem of reaching a consensus in an ad hoc network can be described as a labeling problem. This involves assigning labels to a set of sites. In the case of the ad hoc network, the sites are the nodes of the network and consensus means that the same label must be assigned to each site. The label to be assigned represents some network-wide parameter (e.g., radio frequency of operation and MAC scheme).

In a network of M nodes, the label at a site, x , is denoted $l(\bar{x})$ with the set of all labels represented by $L = [l(x_1), l(x_2), \dots, l(x_M)]$. Consider that each node can make some observations that can help determine the choice of label and let Θ denote the set of observations at all nodes. To choose the best label configuration implies choosing L to maximize $p(L|\Theta)$, the MAP estimate for L , and to do this we can proceed in a Bayesian fashion as follows:

$$\underbrace{p(L|\Theta)}_{\text{Posterior}} \propto \underbrace{p(\Theta|L)}_{\text{Likelihood}} \times \underbrace{p(L)}_{\text{Prior}} \quad (1)$$

Employing Bayesian inference in this way allows the nodes to make decisions by coherently combining the evidence they observe with prior beliefs. To proceed, specific functions must be designed that express the likelihood and prior. The likelihood provides a connection between labels and observed data, while the prior encodes some belief about the label configuration before any observations are made. The likelihood used here will be described later as it depends on the specific decision-making problem. The prior is of chief concern and is discussed next.

THE MRF AS A PRIOR

It is required to encourage a situation in which the choice of label at a node is not an independent choice of any individual node but depends also on the choice of the other nodes in the network. A MRF can be used as a prior on L to capture this interdependency. The MRF is a field in which the conditional probability that a site is in some state depends only on a subset

of the sites nearby and not all other sites available. It is through these local dependencies that longer term dependencies can be modeled. In other words, the MRF as a prior can be used to drive consensus.

FINDING A SOLUTION

To find the labels that maximize $p(L|\Theta)$, using the MRF-MAP framework is in fact quite challenging. The classic Monte Carlo approach to solving this problem is to draw samples of L from the distribution $L \sim p(L|\Theta)$. Collecting a large number of these samples then allows the *measurement* of the probability density function $p(L|\Theta)$, and hence the best L can be selected. In the case of the ad hoc network, the problem as currently formulated would necessitate some kind of control node or god-like object that can see/communicate with all other nodes to determine the labels that maximize $p(L|\Theta)$. Some kind of leadership elections could be carried out that would select a *leader node* that could subsequently manage the choice of label. Such an approach is not appropriate in highly mobile ad hoc networks as membership of the network is in constant flux and new elections would be necessary when the leader leaves. To make the problem tractable, it must be converted to a problem requiring only local decision making at each node. Mechanisms for doing this can be borrowed from the field of image processing.

CONVERSION FROM A GLOBAL TO A LOCAL PROBLEM

In image processing, the labeling problem is well studied. The sites to be labeled are usually pixels, and the labels can have a wide range of meanings (for example, pixel grey level or edge/nonedge). The goal is typically to design a process by which, through local interactions, entire patches of image are associated with the same label, i.e., are segmented into homogenous segments. This is precisely the problem to be addressed in the ad hoc networking context: the need to seek global optimization through local interactions. Insights into the problem posed in this article can be gained from the well established approaches to the problem in the image processing fields [9]–[12].

The most well known of these is the Gibbs sampler [13]. Samples from a complicated multidimensional distribution can be generated by recursively sampling from local conditional distributions. These local conditionals can be much simpler and may involve just a single node. This solves the problem of computational complexity but crucially converts the problem from a centralized problem to a fully distributed problem. Hence given three nodes for instance, with some initial configuration l_1^0, l_2^0, l_3^0 , the Gibbs Sampler requires the following iterations:

$$\begin{aligned} l_1^1 &\sim p(l_1|\theta_1, l_2^0, l_3^0), & l_2^1 &\sim p(l_2|\theta_2, l_1^1, l_3^0), \\ l_3^1 &\sim p(l_3|\theta_3, l_1^1, l_2^1), & l_1^2 &\sim p(l_1|\theta_1, l_2^1, l_3^1), \\ l_2^2 &\sim p(l_2|\theta_2, l_1^2, l_3^1), & l_3^2 &\sim p(l_3|\theta_3, l_1^2, l_2^2), \end{aligned} \quad (2)$$

where $\theta_1, \theta_2, \theta_3$ are the data observations made at nodes 1, 2, 3, respectively.

The fact the network is an MRF means that the label configuration of a given node is only dependent upon the configurations of the *neighboring* nodes. The exact definition of neighbor is dealt with in the next section but to understand how this works, consider three nodes lying along a line such that l_1 is connected to l_2 but not l_3 , and l_2 is connected to both l_1 and l_3 . In this case l_1 and l_3 have one neighbor, and l_2 has two neighbors. Then the Gibbs sampler would proceed as

$$\begin{aligned} l_1^1 &\sim p(l_1|\theta_1, l_2^0,), & l_2^1 &\sim p(l_2|\theta_2, l_1^1, l_3^0), \\ l_3^1 &\sim p(l_3|\theta_3, l_2^1,), & l_1^2 &\sim p(l_1|\theta_1, l_2^1,), \\ l_2^2 &\sim p(l_2|\theta_2, l_1^2, l_3^1), & l_3^2 &\sim p(l_3|\theta_3, l_2^2). \end{aligned} \quad (3)$$

The centralized and global optimization problem has now been converted into a distributed and local optimization problem. The process and the problem can now be expressed as follows:

$$\underbrace{p(l(x)|L(-x), \Theta)}_{\text{Posterior}} \propto \underbrace{p(\Theta(x)|l(x))}_{\text{Likelihood}} \times \underbrace{p(l(x)|L(-x))}_{\text{Prior}}, \quad (4)$$

where $L(-x)$ denotes all the neighbors of the node, as distinct from all the remaining nodes in the network. This is exactly the strategy needed for ad hoc decision making. Nodes can make decisions *autonomously* and *in place* without recourse to any knowledge that concerns nodes which cannot be *seen*, i.e., that are outside its neighborhood. The key point to notice here is that decision making at a site now depends *only* on this neighborhood. The neighborhood can be quite small. In this context, all nodes within one hop of the current one is sufficient: a first-order neighborhood. This specification does not compromise the overall system from allowing for more long-range dependencies. This can be understood by returning to the three-node problem introduced above. In that problem, the state of l_1 depends only on l_2 . And similarly l_3 depends only on l_2 . Despite this, the state of l_3 is influenced by l_1 because it is influenced by l_2 which in turn is influenced by l_1 . This shows that the MRF has specified local interactions *only*, yet is able to account for longer term dependencies through the interconnectivity of the network. As each node has its own computing power the solution can be compared to a massively parallel image processing solution.

One last key point in connection with the Gibbs sampler should be made. As mentioned earlier, the lack of network-wide time synchronization in ad hoc networks means that no time bounds can be placed on any processes. The Gibbs sampler supports synchronous and asynchronous updating of the choice of label at each node [9]. There are very interesting questions here about synchronous versus asynchronous updating, but it is fortunate that regardless of the strategy (and asynchronous is necessary for ad hoc decision making) the Gibbs sampler will converge to samples from the underlying global distribution.

THE NOTION OF NEIGHBOR IN AN AD HOC NETWORK

In image processing the neighbors typically tend to be adjacent pixels, the number of neighbors is typically the same for all sites

(except sites at the edge of an image) and the number of neighbors of any given site typically remains constant throughout the process. We define the neighbors of an ad hoc node to be all nodes from which it can successfully receive packets in one hop. The number of neighbors can grow and recede over the lifetime of the MANET; it is a highly dynamic number. Figure 1 illustrates the notion of neighbor in an ad hoc network at two different times, t_1 and t_2 . In Figure 1, the nodes of interest are node 1 and node 6. The radio range of each of these nodes is illustrated in the figure as a dotted circle around each node. At time t_1 node 1 has nodes 2, 3, and 4 as its neighbors. Node 6 has nodes 4, 5, and 7 as neighbors. At the later time there has been much movement of nodes and a new pattern arises. Of particular note is node X, which although within radio range of node 6, is not a neighbor of node 6 as node 6 does not know it exists. This may occur either because it is deliberately not communicating or due to some physical phenomena. The very dynamic nature of the neighborhood does not in any way compromise the MRF-MAP framework. It is simply the case that when calculating (4), use is made of whatever neighbors exist at the time the calculation is made. The different number of nodes is taken into account in the prior.

THE PRIOR

The MRF prior, $p(l(x)|L(-x))$, or $p(l)$ in short hand, can be expressed as follows [14], [13]:

$$p(l) \propto \exp -\lambda \sum_{l_k \in \mathcal{N}} \sum_{l \neq l_k} \frac{l \neq l_k}{\#C}, \quad (5)$$

where λ is a tuning parameter, l is shorthand for $l(x)$, and the set \mathcal{N} is the set of labels at sites that are neighbors of l . Hence l_k is one label in the neighborhood, and the probability $p(l)$ is highest when l agrees with all or most of the labels in the neighborhood. This is the effect that drives consensus in the framework. Because the number of sites in the neighborhood, $\#C$, is not the same from site to site, it must be incorporated into the prior as a normalising factor. This allows nodes that have fewer neighbors to attach more weight to achieving consensus (i.e., more weight to the prior) and vice-versa.

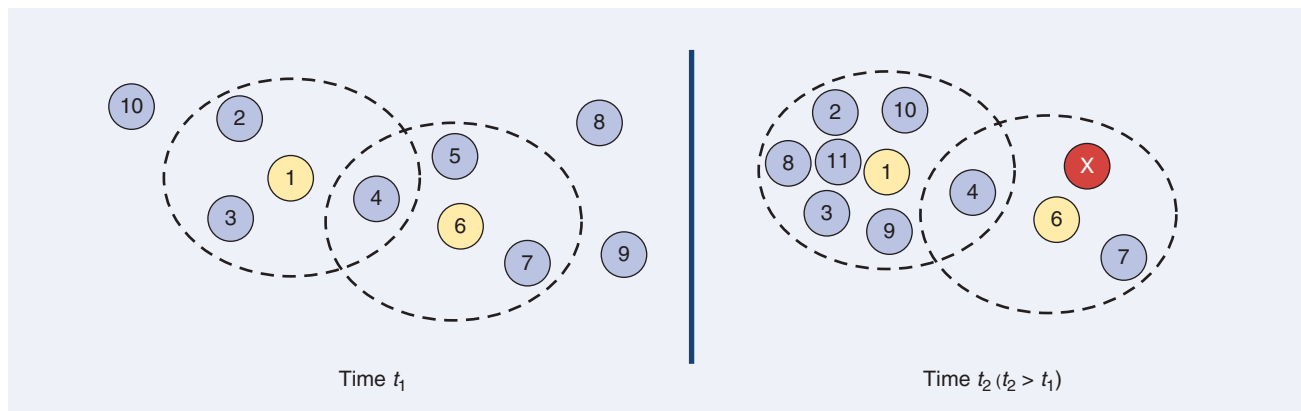
λ provides a means of stipulating how much *peer pressure* should be allowed in the network. In other words, the MRF-MAP framework can be tuned to provide a balance between the desire for consensus and the ability of a node to adapt to network conditions. The idea of a tuning parameter opens up many interesting discussions in the context of an ad hoc network as nodes could theoretically adopt different tuning parameters in very nonhomogenous environments. Of course λ can also be used to encourage fragmentation (texture). This may be a desired effect, e.g., in a dynamic frequency/cell planning exercise for a network. Groups of neighboring nodes could, for example, select a frequency of operation that would not interfere with the frequency selected by other groups. This discussion is for future work but nonetheless emphasizes the generality of the proposed framework.

AN MRF-MAP DECISION-MAKING FRAMEWORK FOR ROUTING PROTOCOL CHOICE

ROUTING IN AD HOC NETWORKS

Ad hoc routing protocols are at the core of ad hoc networks and must be specially designed to deal with the dynamic nature of the ad hoc network environment. A broad range of routing protocols have been designed and developed for use in ad hoc networks. Routing protocols can be loosely broken in to proactive protocols (e.g., optimal link state routing (OLSR) [2]), reactive protocols (e.g., dynamic source routing (DSR) [1]) and hybrids (e.g., zone routing protocol [15]). (There are also a class of routing protocols known as location-based routing protocols, but these depend on nodes having access to location information.)

In general, the goal has been to identify an optimal ad hoc routing protocol. However, it has been well established in the literature that the current routing protocols are not individually suited to all networking conditions that would typically be encountered by a MANET [16]–[19]. The literature, and our own simulations, identify the major factors affecting the choice of routing protocol as being relative node mobility, node density and traffic conditions when end-to-end data throughput and routing protocol control overhead are used as measures of a



[FIG1] Understanding neighbors in ad hoc networks.

protocol's performance. More specifically, DSR and OLSR have been comparatively evaluated in many surveys, among them [19] and [17], which demonstrate their disparate realms of optimal performance.

DSR reactively creates routes between source and destination nodes when requested by higher application layers. As such, DSR uses aperiodic network-layer signalling to discover and maintain routes. On the other hand, OLSR proactively uses local periodic beaconing to perform neighborhood discovery, and it uses a predominantly periodic flooding technique to relay full topology information throughout the network. In both cases, the flow of data across the network invokes aperiodic signaling at the network layer. The result of this is that the level of neighbor-to-neighbor signaling varies by protocol and by traffic demands. This signaling has, for all intent and purposes, the effect of acting like a network-layer radar that reveals the presence and state of neighboring nodes with varying resolutions of clarity.

Rather than continuing to search for that *one* routing protocol that suits all network conditions, an alternative approach is to design a system that allows nodes to dynamically reconfigure themselves to utilize the most suitable ad hoc routing protocol for the prevailing network conditions. An adaptive network layer would facilitate such a complete change of routing protocol. And of course, choosing the best routing protocol for the prevailing conditions necessitates a consensus decision to be made, as all nodes must run the same protocol. Some work has been carried out on adaptive routing protocols for ad hoc networks but this work has tended to focus on adaptivity within a given protocol (*nodal change*) rather than adapting between different protocols (*network-wide change*).

The application example used to illustrate the power of the MRF-MAP framework is based on a reconfigurable ad hoc network layer that allows nodes to dynamically configure and utilize the most suitable ad hoc routing protocol for the prevailing network conditions. This is a complicated example that illustrates many of the key features of the framework as well as highlighting the challenges. For the purposes of this article we have focused on reconfiguring the network layer between two different protocols, DSR and OLSR. In this case when we speak of a protocol being *more suited* we mean the data throughput is better than if an unsuited protocol were in operation [19], [17]. The remainder of the article focuses on DSR and OLSR when presenting results and findings.

EXPERIMENTAL FRAMEWORK

Before describing the experiments and results, it is useful to describe the experimental platform. The research described in the article was carried out using a real ad hoc network, known as the Dublin Ad hoc Wireless Network (DAWN) [20]. DAWN was

designed and created to facilitate research in the area of ad hoc networking. At the core of DAWN is a dynamic modular communication stack that runs on each of the nodes of the ad hoc network. A generic layer interface allows the dynamic assembly of these layers to form a network communication stack consisting of the relevant hardware and software elements. The MRF-MAP framework is what we call a *stack service* that can be called on by any layer of a DAWN node to help it make decisions. In this

case, the network layer uses the decision-making framework to decide between DSR and OLSR. The DAWN stack has a facility for swapping between protocols at a particular layer. The physical layer in DAWN can deal with a large number of physical frontends, including IrDA,

802.11, and an in-house software radio known as IRIS (Implementing Radio In Software) [21], which facilitates reconfigurability in the radio. There is also a simulation layer in DAWN, and this was used during the experimental work as it permits the simulation of highly mobile nodes.

BUILDING THE MRF-MAP FRAMEWORK FOR ROUTING PROTOCOL CHOICE

To build the MRF-MAP decision-making framework for the purposes of routing protocol selection, it is necessary to determine the required local observations and to compute the associated likelihood functions.

LOCAL OBSERVATIONS

To use the MRF-MAP framework in our application example, nodes must be capable of *locally* observing node mobility, node density, and node traffic conditions. This can be achieved as follows:

- 1) The local mobility of a node can potentially be determined if the node knows its current and previous location and the time taken to move between the two positions. In the absence of a positioning system, link change rate (LCR) can be used by a node to get a sense of its mobility. The LCR metric is defined as the number of communication links forming and breaking between nodes over a given time.
- 2) In terms of network density, average node degree (ND) is a local proxy measure. Node degree can be defined as the number of one-hop neighbors that a node is aware of.
- 3) The number of data packets per second (PPS) passing through the node is a local measure of the traffic.

There are a number of points that should be made in connection with the observations. First, the probability of the observed LCR statistic is dependent on the observed average ND. The more neighbors (communication links) a node has, the more link changes can occur in a mobile environment. Second, the observed PPS statistic is dependent on the current protocol, $\hat{l}(x)$. While DSR buffers data packets for transmission, OLSR does not and will only transmit data if a valid route is available.

MANETS DEMAND APPLICATIONS, SERVICES, AND PROTOCOLS THAT CAN MANAGE THE NETWORK WITHOUT MANUAL INTERVENTION, AND ADAPT TO DYNAMIC NETWORK CONDITIONS.

As a result, less data packets are observed in an OLSR network compared to a DSR network under identical traffic loading.

So, if the observations at the node are $\theta(\mathbf{x}) = \{\theta_{\text{lcr}}(\mathbf{x}), \theta_{\text{nd}}(\mathbf{x}), \theta_{\text{pps}}(\mathbf{x})\}$ where $\theta_{\text{lcr}}(\mathbf{x})$ is the observed LCR, $\theta_{\text{nd}}(\mathbf{x})$ is the observed ND and $\theta_{\text{pps}}(\mathbf{x})$ is the observed PPS, then the likelihood function is $p(\theta(\mathbf{x})|l(\mathbf{x}), \hat{l}(\mathbf{x}), L) = p(\theta_{\text{lcr}}(\mathbf{x}), \theta_{\text{nd}}(\mathbf{x}), \theta_{\text{pps}}(\mathbf{x})|l(\mathbf{x}), \hat{l}(\mathbf{x}), L)$. This can be expressed as a product of conditional distributions given the aforementioned dependencies, resulting in

$$p(\theta(\mathbf{x})|l(\mathbf{x}), \hat{l}(\mathbf{x}), L) = p(\theta_{\text{lcr}}(\mathbf{x})|\theta_{\text{nd}}(\mathbf{x}), l(\mathbf{x})) \cdot p(\theta_{\text{nd}}(\mathbf{x})|l(\mathbf{x}))p(\theta_{\text{pps}}(\mathbf{x})|l(\mathbf{x}), \hat{l}(\mathbf{x})). \quad (6)$$

DETERMINING THE LIKELIHOOD FUNCTIONS

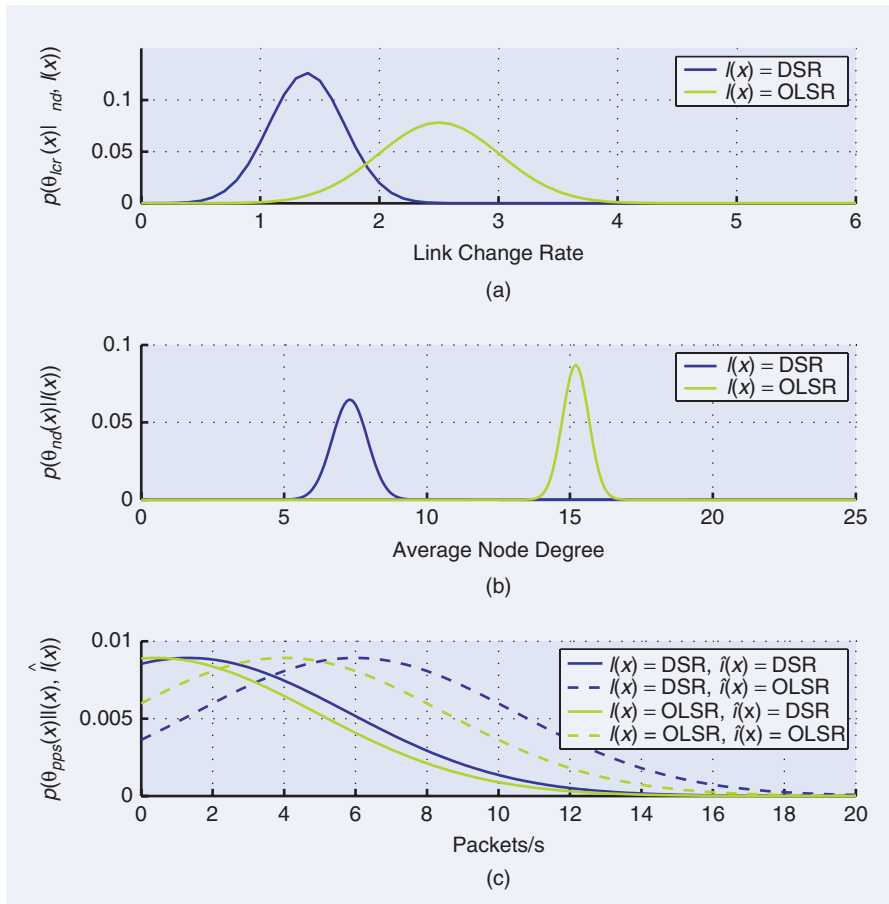
The form of each of the probability density functions for the local observations was evaluated by experiment using DAWN. In all simulations, the random waypoint (on a torus) model [22] was used as the mobility model. This model ensures that the local observations of LCR and average node degree are spatially sta-

tionary. This is essential so that artifacts of the mobility model do not influence the results, a point that can often be neglected. The global mobility is set by specifying node speed. Traffic is generated by constant bit rate (CBR) sources that select a uniformly random destination for each packet. Packets are presented to the network layer every 5 s, and traffic loading is altered by changing the number of nodes acting as CBR sources. The node density of the network was controlled via the simulation area. To create dense conditions smaller, simulation areas are used.

For the experiment, we defined two scenarios that exhibit different network characteristics. **Scenario A** features a dense network of 20 nodes in an area of $500 \times 500 \text{ m}^2$. Mobility is low with an average node speed of 1 m/s. Traffic originates from 5 CBR sources. Under these conditions, a network operating OLSR has a higher throughput performance than one operating DSR. On observing local measures of LCR, ND, and PPS indicative of this scenario, a node should favor implementation of OLSR. **Scenario B** features a sparse network of 20 nodes in an area of $700 \times 700 \text{ m}^2$. Mobility is also low with an average node speed of 1 m/s. Traffic load is high with 20 CBR sources. Under these conditions, a network operating DSR has a higher

throughput performance than one operating OLSR. On observing local measures of LCR, ND and PPS indicative of this scenario, a node should favor implementation of DSR. Hence, in the problem specified, each node can select one of two values for its label: $l(\cdot) \in \{A, B\}$ alternatively $l(\cdot) \in \{\text{OLSR}, \text{DSR}\}$. The likelihood functions were determined by analyzing the local observations of 20 nodes in the network conditions of Scenarios A and B. Ten trials of 500 s each were performed.

Our experiments show that $p(\theta_{\text{nd}}(\mathbf{x})|l(\mathbf{x}))$ and $p(\theta_{\text{pps}}(\mathbf{x})|l(\mathbf{x}), \hat{l}(\mathbf{x}))$ are approximated well by 1-D Gaussians, as illustrated in Figure 2. The distributions for PPS are vague and the dependency on the current protocol is not so pronounced. The likelihood $p(\theta_{\text{lcr}}|\cdot)$ is more complicated. Since there is a conditional dependency between LCR and ND, the framework should use the joint distribution, $p(\theta_{\text{lcr}}(\mathbf{x}), \theta_{\text{nd}}(\mathbf{x})|l(\mathbf{x}))$. This is potentially computationally expensive, so we approximate with $p(\theta_{\text{lcr}}(\mathbf{x})|\mu_{\text{nd}}, l(\mathbf{x}))$, where μ_{nd} is the mean of $p(\theta_{\text{nd}}(\mathbf{x})|l(\mathbf{x}))$. The variance of $p(\theta_{\text{nd}}(\mathbf{x})|l(\mathbf{x}))$ is small and thus this simplification does not impact on the solution. $p(\theta_{\text{lcr}}(\mathbf{x})|\mu_{\text{nd}}, l(\mathbf{x}))$ is also Gaussian, as illustrated in Figure 2(a).



[FIG2] Likelihood functions. Part (c) shows $p(\theta_{\text{pps}}|\cdot)$ and its dependency on the current label \hat{l} . Hence $p(\theta_{\text{pps}}|l = B, \hat{l} = B) \approx \mathcal{N}(1.32, 20)$, $p(\theta_{\text{pps}}|l = B, \hat{l} = A) \approx \mathcal{N}(6, 20)$; $p(\theta_{\text{pps}}|l = A, \hat{l} = A) \approx \mathcal{N}(4, 20)$, $p(\theta_{\text{pps}}|l = A, \hat{l} = B) \approx \mathcal{N}(0.4, 20)$. Part (b) shows $p(\theta_{\text{nd}}|l = A) = \mathcal{N}(15.2, 0.21)$; $p(\theta_{\text{nd}}|l = B) = \mathcal{N}(7.3, 0.38)$ and (a) shows $p(\theta_{\text{lcr}}|l = A) = \mathcal{N}(2.5, 0.26)$; $p(\theta_{\text{lcr}}|l = B) = \mathcal{N}(1.39, 0.1)$.

THE MRF-MAP FRAMEWORK ALGORITHM

Recall that the problem is now a local one. It is required to assign a label $l(x)$ to the node at site x . This is achieved by choosing the label that maximises $p(l(x)|L(-x), \Theta(x))$, (4). Given that in this case there are just two labels A, B representing the two choices for protocol at a node, then this amounts to selecting the larger of $p(l(x) = A|\cdot)$ or $p(l(x) = B|\cdot)$. This is the Iterated Conditional Modes (ICM) algorithm as proposed by Besag [9]. An alternate strategy is to run the Gibbs sampler as discussed earlier, but ICM is computationally more attractive.

Using the negative logarithm of the a-posteriori distribution, $-\ln[p(l(x)|L(-x), \Theta(x))]$, avoids issues with scaling and exponentials, converts the product to a sum, and yields a more direct solution with

$$-\ln[p(l(x)|L(-x), \Theta(x))] \propto \ln(p(\Theta|l(x)) + \left(\lambda \sum_{l_k \in \mathcal{N}} \frac{(l(x) \neq l_k)}{\#C} \right)). \quad (7)$$

Hence the ICM algorithm at each node proceeds as follows, given the current node is at site i with current label \hat{l}_i .

- 1) Make measurements of θ_{pps} , θ_{lcr} , θ_{nd} at the site i .
- 2) Set $l_i = A$, measure $-\ln[p(\Theta|l_i = A)]$ by combining the likelihoods shown in Figure 2. Define this as an energy E_A^l .
- 3) Set $l_i = B$, measure $-\ln[p(\Theta|l_i = A)]$ in a similar manner to that above. Define this as an energy E_B^l .
- 4) Set $l_i = A$, measure $\lambda \sum_{l_k \in \mathcal{N}} ((A \neq l_k)/\#C)$, $i \neq k$. This is the MRF energy and requires measurement of the state of the nodes that are one hop away (l_k) from site i . Note that

$\sum_{l_k \in \mathcal{N}} (A \neq l_k)$ is the number of these nodes having a label that is *not* A . Define this as an energy E_A^s .

5) Set $l_i = B$, measure the MRF energy (as above), $E_B^s = \sum_{l_k \in \mathcal{N}} ((B \neq l_k)/\#C)$.

6) If $E_A^l + E_A^s \leq E_B^l + E_B^s$, then choose $l_i = A$. Else $l_i = B$.

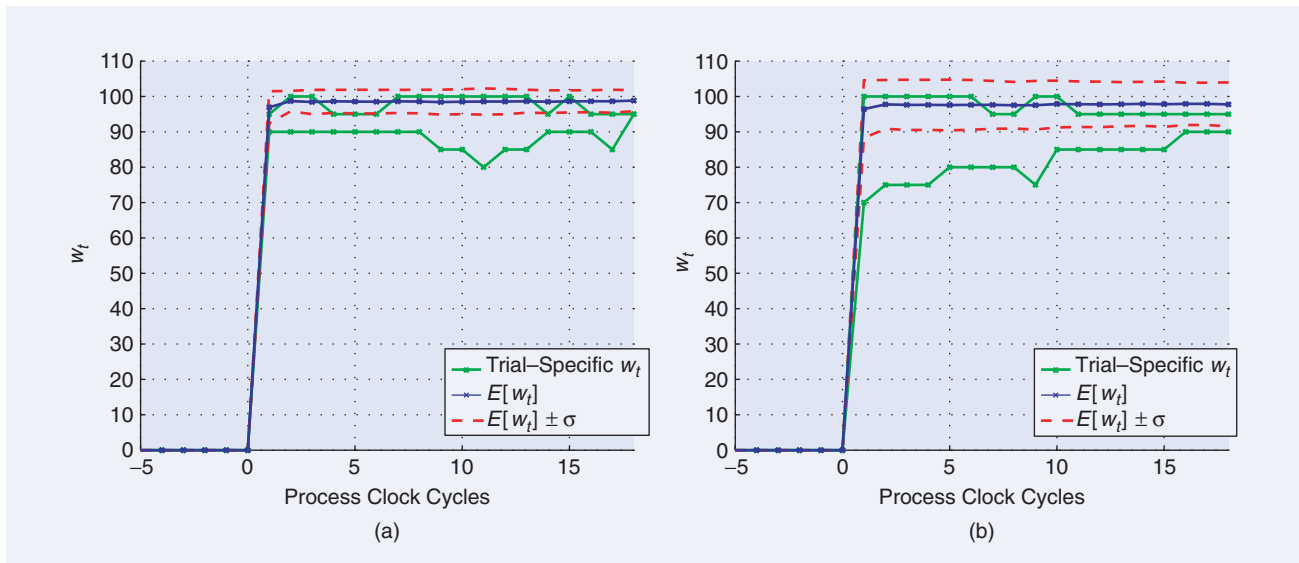
The implementation of the algorithm for the decision-making framework is therefore straightforward. At each process clock cycle, nodes incorporate the local observations of LCR, ND, and PPS in the likelihood function. The

prior (5) is used to get a measure of how many of the neighboring nodes are similar to l_i . The protocol that yields the minimum energy is then selected by the node.

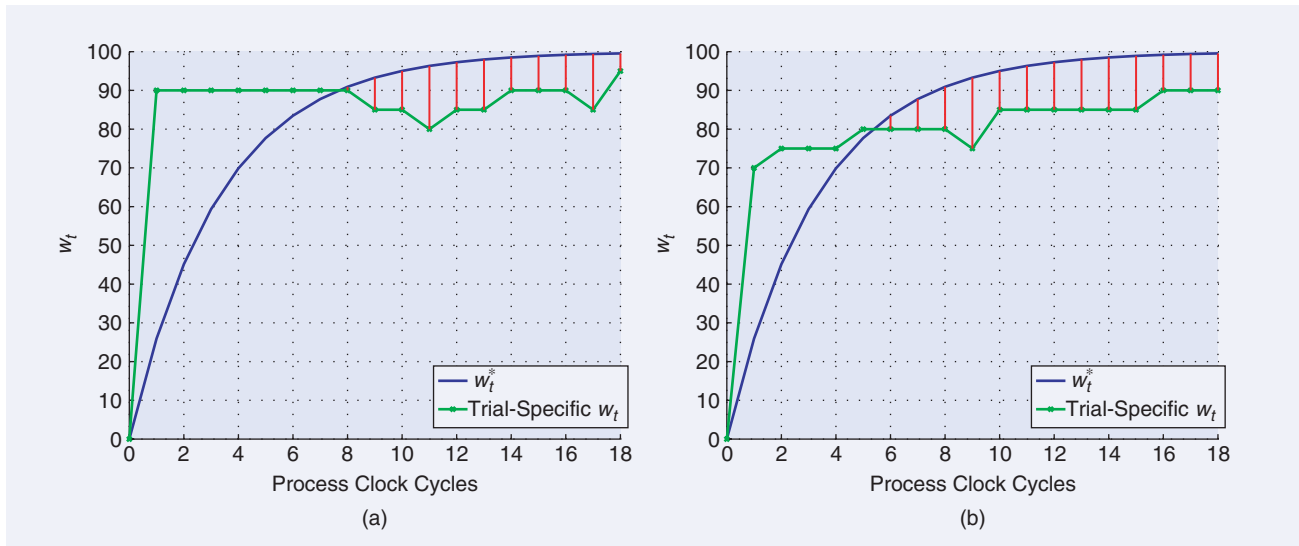
RESULTS AND ANALYSIS

Recall that in the overall problem, nodes initially operate under a specific set of network conditions (mobility, density, and traffic load). Consider that the nodes are operating the optimal routing protocol for that set of conditions. At some time τ , network conditions change so as to make the current protocol nonoptimal. The task of the MRF-MAP framework is to facilitate a distributed consensus process in the network, resulting in the selection of a new protocol which is optimal for the new environmental situation.

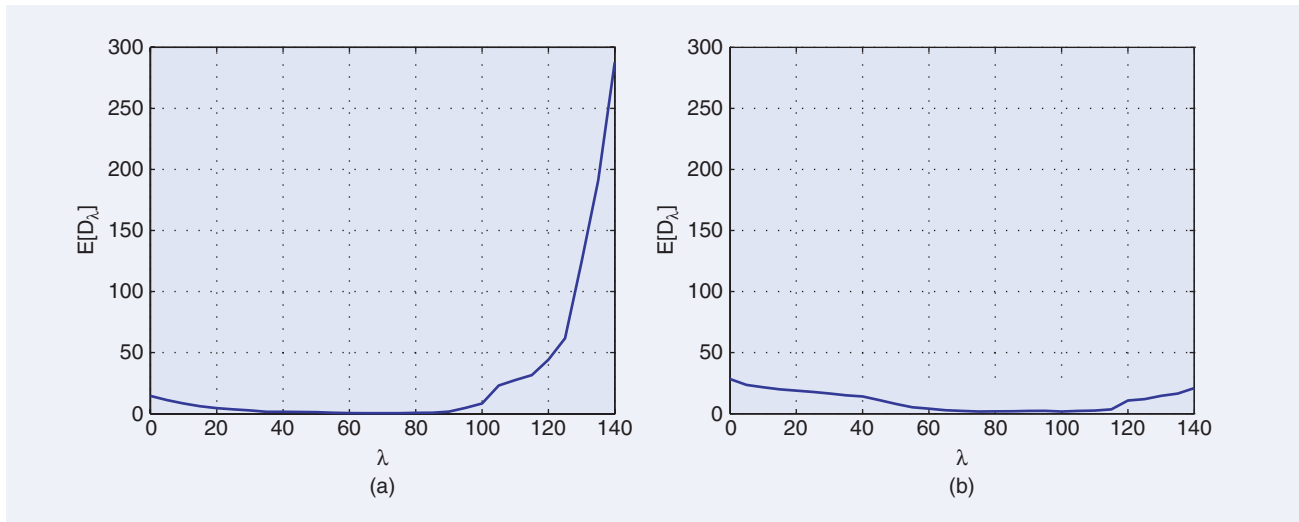
To demonstrate this behavior, two distinct simulations are set up. Network conditions in the first set of simulations change from Scenario A to Scenario B, and vice versa in the second set. The time at which network conditions change is set to $\tau = 0$. The network-wide choice of routing protocol is then analyzed. As stated earlier, OLSR is optimal for Scenario A and DSR is optimal for Scenario B. The optimal protocol is therefore



[FIG3] Expected $w_t \pm$ standard deviation for $\lambda = 5$. Example w_t for individual trials are also displayed. (a) Scenario A \rightarrow Scenario B (OLSR optimal). (b) Scenario B \rightarrow Scenario A (DSR optimal).



[FIG4] Target w_t^* compared to realizations of w_t for individual trials. (a) Scenario A \rightarrow Scenario B (OLSR optimal). (b) Scenario B \rightarrow Scenario A (DSR optimal).



[FIG5] Convergence statistic D_λ against λ for each set of simulations. (a) Scenario A \rightarrow Scenario B (OLSR optimal). (b) Scenario B \rightarrow Scenario A (DSR optimal).

known a priori. A good statistic to capture the dynamic behavior of the framework is the percentage of nodes operating the optimal protocol, defined as w_t , at time t . There are a number of issues to consider such as the ability of the network to reach consensus, the length of time taken to reach such consensus, and the effect of λ on the convergence to consensus.

Furthermore, we consider the whether it is possible to estimate an optimal value of λ .

Figure 3 shows results from the two sets of simulations. In each case the expected value for w_t , $E[w_t]$, is evaluated over 100 trials and plotted against *Process Clock Cycles* for $\lambda = 5$.

AD HOC ROUTING PROTOCOLS ARE AT THE CORE OF AD HOC NETWORKS AND MUST BE SPECIALLY DESIGNED TO DEAL WITH THE DYNAMIC NATURE OF THE AD HOC NETWORK ENVIRONMENT.

In both cases more than 95% of the nodes have attained the optimal protocol within five clock cycles. The dashed lines represent one standard deviation, σ , about the mean. Majority consensus has indeed been achieved with an expected 98% of nodes on the optimal protocol. However, there is fluctuation between each trial run due to the random nature of

the simulation, i.e., variance in local observations between sites. Examples of w_t for specific trials are also shown to illustrate possible node behaviors and the significance of the standard deviation plots. Although nodes are converging on the optimal protocol, some thrashing of protocol choice occurs in

individual nodes. This behavior is unacceptable as nodes on different protocols cannot communicate. Therefore, while it is important for $E[w_t]$ to converge to 100%, it is just as important for the standard deviation to converge to zero.

To capture the performance of the framework for different λ , a target convergence behavior is defined as $w_t^* = 100(1 - \exp[-\eta t])$. The parameter η is chosen to yield a target of 95% of nodes converged in ten clock cycles with exponential convergence. w_t is compared to w_t^* as shown in (8) to produce the convergence statistic D_λ defined as follows.

$$D_\lambda = \sum_t (w_t^* - w_t) \mathbb{I}_{\{w_t^* > w_t\}}, \quad (8)$$

where

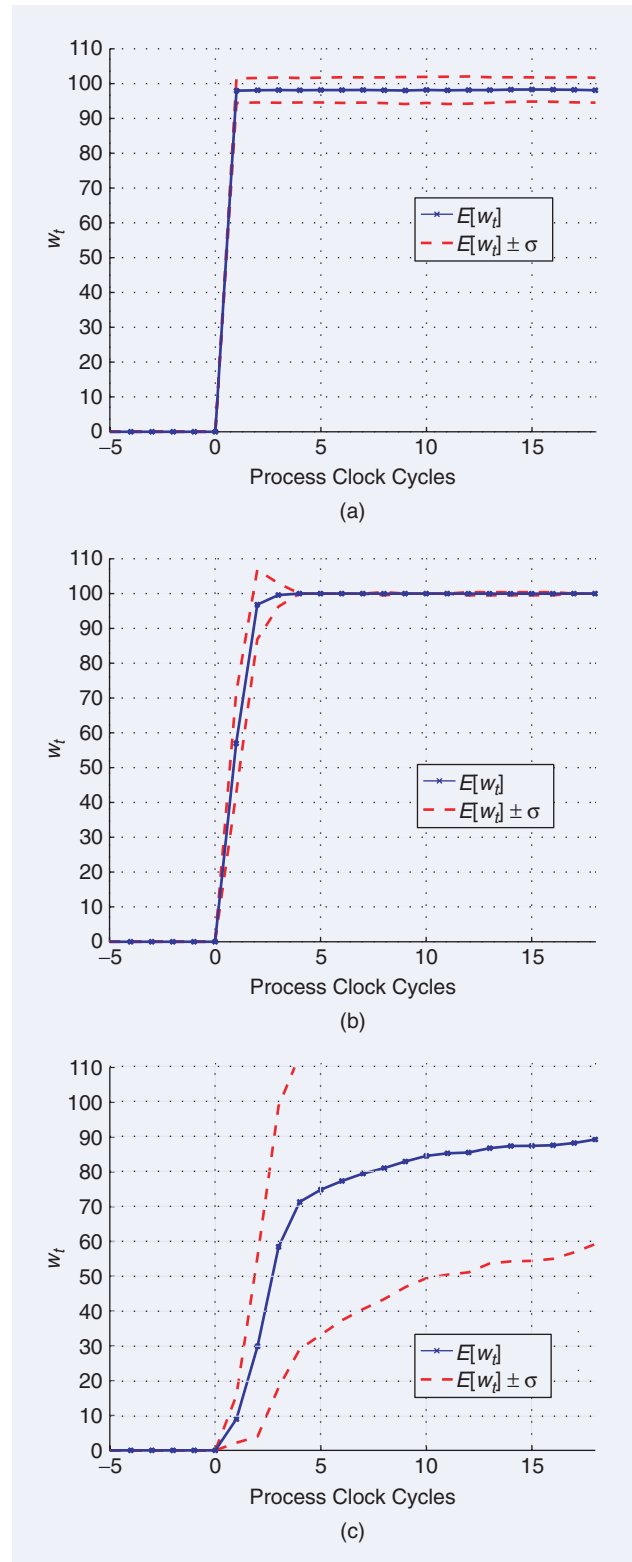
$$\mathbb{I}_{\{w_t^* > w_t\}} = \begin{cases} 1 & w_t^* > w_t \\ 0 & \text{otherwise.} \end{cases}$$

D_λ is proportional to the degree with which w_t is below the ideal value and is a function of λ . Figure 4(a) and (b) compares w_t^* to trial-specific w_t for both sets of simulations. The convergence statistic D_λ is equivalent to the sum of the red lines. Hence D_λ is a measure of the distance of a particular experimental observation, e.g., w_t in Figure 4, from the target performance w_t^* . Large values of D_λ imply poor convergence while small values imply performance attaining our target. Note that our target is chosen as an indicative baseline measure only. The expected value for D_λ is evaluated over 100 trials for each set of simulations.

Figure 5 shows $E[D_\lambda]$ against λ for each set of simulations. It shows that there is a broad range of λ that is optimal for decision making and that values of λ may be chosen such that they suit both scenarios. This implies a certain amount of robustness of the system to the choice of λ . Figure 5 also shows how λ affects the overall convergence. λ provides a balance between the desire for consensus (i.e., nodes wish to operate the same protocol as their neighbors) and the need to adapt to the prevailing network conditions. If λ is too small, then nodes do not have sufficient desire to form consensus and fragmentation of the network may take place, where different nodes operate different protocols and communication between them is lost. Nodes may also sporadically thrash between optimal and nonoptimal protocols due to a lack of dependence on neighboring nodes' choices. This behavior can be seen in Figures 6(a) and 7(a) for the case where $\lambda = 0$. If λ is too large, the nodes do not react to changing conditions, instead preferring to form consensus with their neighbors operating the nonoptimal protocol. Again, fragmentation of the network can occur. Figures 6(c) and 7(c) show this behavior at a value of $\lambda = 140$. Figures 6(b) and 7(b) show the MRF-MAP performance at a value of $\lambda = 80$. This value is chosen to be within the range of optimal λ for both simulation sets. The standard deviation tends to zero in the latter stages of simulation, indicating that the thrashing of protocol choice amongst nodes has abated. Thus it is possible to tune λ to produce the desired global behavior in the network.

It should be noted that in these simulations, network conditions strongly favor either DSR or OLSR, and this is reflected in

the local observations of the nodes. The network conditions are highly stationary at all points of the network, and so nodes are likely to infer the optimal protocol from local observations alone



[FIG6] Expected $w_t \pm$ standard deviation. Scenario A \rightarrow Scenario B (OLSR optimal). (a) $\lambda = 0$, (b) $\lambda = 80$, (c) $\lambda = 140$.

($\lambda = 0$). However, the stochastic nature of the ad hoc networks implies that some observations may cause a node to infer the nonoptimal protocol, and it is in these cases that the framework facilitates convergence to the optimal protocol and ensures that

nodes remain on the optimal protocol. The results prove that a nontrivial choice of λ leads to higher consensus among nodes and a reduction in nodes thrashing to the nonoptimal protocols. In practical terms, self-stabilizing techniques can be used to avoid thrashing once nodes have been perturbed from a suboptimal state to the optimal state using the MRF-MAP framework [23].

The MRF-MAP framework is shown to be a viable method of producing distributed consensus in ad hoc networks. The stochastic nature of ad hoc networks presents a challenge to the decision-making process, however appropriate selection of λ allows for quick convergence to the optimal protocol. Analysis reveals a large range of optimal λ and consequently that the system is relatively robust to the choice of λ .

APPLICABILITY OF THE MRF-MAP FRAMEWORK

The MRF-MAP framework has proven to be a useful tool in network-wide protocol selection from the evidence of the experimental work described in this article. The question now arises as to its use for other network-wide decisions. There are a number of key issues to be considered here.

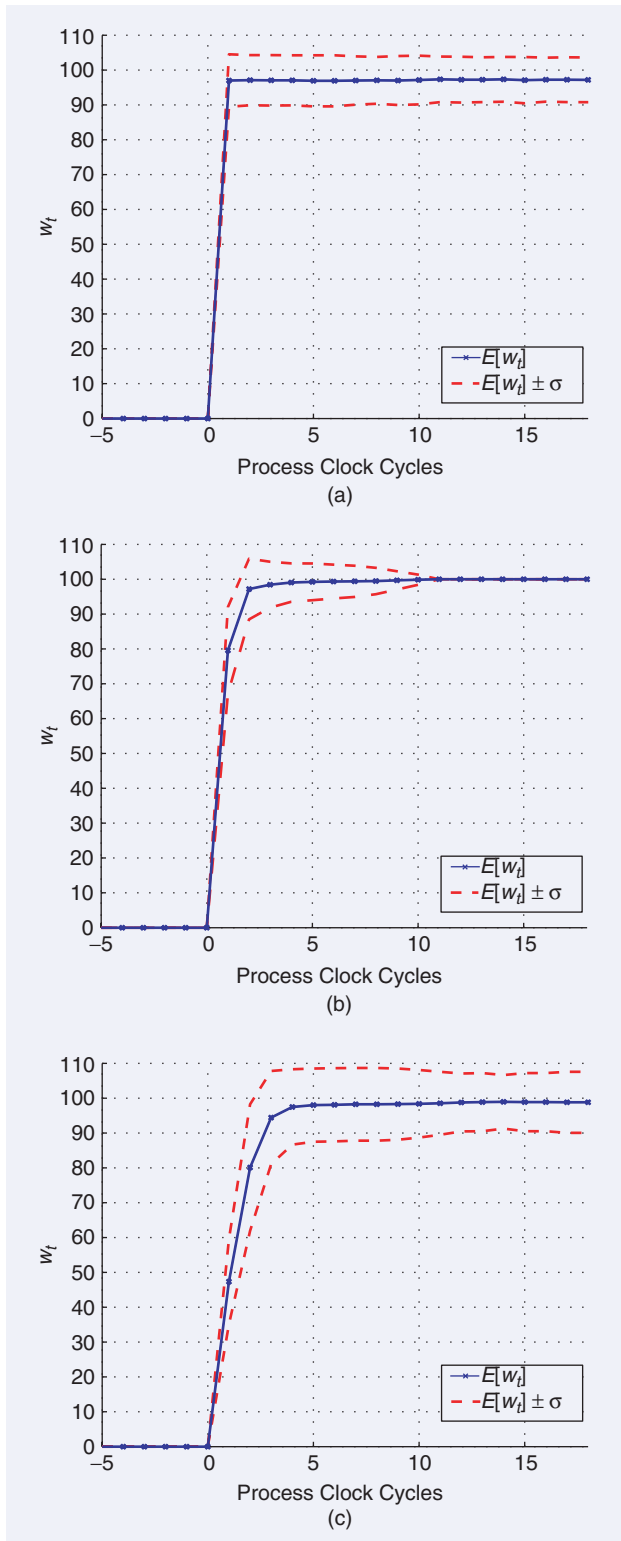
The first issue relates to the time dynamics of the problem at hand. Timing in the case of the MRF-MAP framework is a relative issue and very much depends on the decision to be made. If the time taken for a node to make an observation and the time taken for the framework to make a decision is so long that the network conditions have changed appreciably in the meantime, then it is not suitable.

The second issue in terms of the applicability of the framework is the issue of the observations themselves. In some cases there may not be a local physical manifestation of the effects of a global parameter. And in other cases, even when there is something that can be measured locally, there may be no causal link between local measurements and the global conditions. In such cases the framework has no role to play but nor does any distributed decision-making framework. In general, a centralized approach would be necessary.

The third issue that is important for the applicability of the framework relates to the actual implementation. This has been alluded to in the previous paragraphs in terms of the calculation of the clique potential. The implementation of the MRF-MAP framework must be carried out in such a way as to not overburden the network. For example nodes can actively query neighbors for their preferences or can opportunistically glean the information during the normal course of communication. The second option is more lightweight and less likely to overburden the network. Hence, implementation choices have an impact on the framework's applicability.

CONCLUSIONS

The MRF-MAP framework presented in this article is a general framework for solving distributed-consensus problems in ad hoc networks. The framework has borrowed much from the field of imaging processing. Ideas from networking have found their way to the image processing domain through such techniques as graph cuts [12]. In this article ideas from image processing are finding a place in networking, thus closing the loop.



[FIG7] Expected $w_t \pm$ standard deviation. Scenario B \rightarrow Scenario A (DSR optimal). (a) $\lambda = 0$ (b) $\lambda = 80$ (c) $\lambda = 140$.

The framework is attractive and elegant and through the initial experiments presented here, very workable. It is clear that the use of the framework is very dependent on designing good likelihood functions and understanding the tuning parameter. Depending on the complexity of the application area this can be an arduous task, but no doubt the same can be said of its application to areas of image processing.

We are continuing the work in protocol selection and looking at the use of nonhomogeneous tuning parameters across the network, among other things. We are currently also expanding the applications of the framework and much can be learned from using it in different ways. Reconfigurable wireless communications systems are very much a part of the future of communication systems in general and the facilitation of reconfiguration in ad hoc networks is in line with developing trends. We are particularly interested in opportunistic use of spectrum, spectrum pooling and spectrum rental for nonincumbents. The MRF-MAP framework can aid self-organizing collaborative and distributed entities to make decisions about key resources in this and many other areas.

ACKNOWLEDGMENT

This material is based on works supported by Science Foundation Ireland under Grant 03/CE3/I405.

AUTHORS

Linda E. Doyle (ledoyle@tcd.ie) is a lecturer in the Department of Electronic and Electrical Engineering, University of Dublin, Trinity College, Ireland. She leads the Emerging Networks strand of the Centre for Telecommunications Value-Chain Research (CTVR). One of the main projects she is currently pursuing, the Plastic Project, focuses on distributed reconfigurable and cognitive wireless networks for dynamic spectrum management.

Anil C. Kokaram (anil.kokaram@tcd.ie) is a senior lecturer in the Electronic Engineering Department at Trinity College Dublin, and he is a Fellow of that same institution. He was awarded the Ph.D. in 1993 from the University of Cambridge. He has published over 70 papers in the area of image and video processing and is an associate editor of *IEEE Transactions on Image Processing*. He is leader of www.sigmedia.tv and has research interests in numerical techniques for Bayesian inference, digital video processing, motion estimation, and visual media content analysis and manipulation.

Senan J. Doyle (doylesj@tcd.ie) is pursuing a Ph.D. in the area of adaptive ad hoc networking and distributed decision-making architectures within the Networks and Telecommunications Research Group of the Department of Electronic & Electrical Engineering, University of Dublin, Trinity College, Ireland. He received his first class bachelor's degree in electronic and computer engineering from the University of Dublin, Trinity College, in 2003 and is an Irish Research Council scholar. He has engaged in research focusing on stationarity in ad hoc simulation, feasible link statistics for adaptive ad hoc architectures and Bayesian applications in medical imaging.

Timothy K. Forde (tim.forde@tcd.ie) is a researcher at the Centre for Telecommunications Value-Chain Research (CTVR), Trinity College Dublin, Ireland. He received his Ph.D. degree in electronic engineering from Trinity College, Dublin, in 2005. He has engaged in various research activities focusing on mobile ad hoc networks, emerging network architectures, and dynamic spectrum access techniques. Currently, he is working on developing a dynamic spectrum access platform for use in licensed Irish radio spectrum.

REFERENCES

- [1] D.B. Johnson and D.A. Maltz, "Dynamic source routing in ad hoc wireless networks," *Mobile Computing*, vol. 353, pp. 153–181, 1996.
- [2] P. Jacquet, P. Mhlethaler, T. Clausen, A. Laouiti, A. Qayyum, and L. Viennot, "Optimized link state routing protocol for ad hoc networks," in *Proc. IEEE INMIC'01*, Lahore, Pakistan, Dec. 2001, pp. 62–68.
- [3] S. Nesargi and R. Prakash, "MANETconf: configuration of hosts in a mobile ad hoc network," in *Proc. 21st Annu. Joint Conf. IEEE Comput. Commun. Soc.*, vol. 2, June 2002, pp. 1059–1068.
- [4] N.H. Vaidya, "Weak duplicate address detection in mobile ad hoc networks," in *Proc. 3rd ACM Int. Symp. Mobile Ad Hoc Networking & Computing*, Lausanne, Switzerland, 2002, pp. 206–216.
- [5] H. Attiya and J. Welch, *Distributed Computing: Fundamentals and Advanced Topics*. McGraw-Hill: New York, 1998.
- [6] R. Guerraoui and A. Schiper, "Consensus: The big misunderstanding," in *Proc. 6th IEEE Workshop on Future Trends of Distributed Computing Syst.*, Tunis, Tunisia, 1997, p. 183.
- [7] M.J. Fischer, N.A. Lynch, and M.S. Paterson, "Impossibility of distributed consensus with one faulty process," *J. Assoc. Comput. Mach.*, vol. 32, no. 2, pp. 374–382, Apr. 1985.
- [8] L. Briesemeister and G. Hommel, "Localized group membership service for ad hoc networks," in *Proc. 2002 Int. Conf. Parallel Processing Workshops*, Vancouver, BC, Canada, Aug. 2002, pp. 94.
- [9] J. Besag, "On the statistical analysis of dirty pictures," *J. Royal Statistical Soc. B*, vol. 48, pp. 259–302, 1986.
- [10] B.S. Manjunath and R. Chellappa, "Unsupervised texture segmentation using Markov random fields," *IEEE Trans. Pattern Anal. Machine Intell.*, vol. 13, pp. 478–482, 1991.
- [11] R.C. Dubes and A.K. Jain, "Random field models in image analysis," *J. Applied Statistics*, vol. 16, pp. 131–164, 1989.
- [12] Y. Boykov, O. Veksler, and R. Zabih, "Fast approximate energy minimization via graph cuts," *IEEE Trans. Pattern Anal. Machine Intell.*, vol. 23, no. 11, pp. 1222–1239, 2001.
- [13] S. Geman and D. Geman, "Stochastic relaxation Gibbs distributions, and the Bayesian restoration of images," *IEEE Trans. Pattern Anal. Machine Intell.*, vol. 6, no. 2, pp. 721–741, 1984.
- [14] S.Z. Li, *Markov Random Field Modeling in Computer Vision*. New York: Springer Verlag, 1995.
- [15] Z.J. Haas, "A new routing protocol for the reconfigurable wireless networks," in *Proc. IEEE 6th Int. Conf. Universal Personal Communications*, 1997, vol. 2, pp. 562–566.
- [16] S.R. Das, C.E. Perkins, and E.M. Royer, "Performance comparison of two on-demand routing protocols for ad hoc networks," *IEEE Pers. Commun.*, vol. 8, no. 1, pp. 16–28, 2001.
- [17] L. Viennot, P. Jacquet, and T.H. Clausen, "Analyzing control traffic overhead versus mobility and data traffic activity in mobile ad-hoc network protocols," *Wireless Networks*, vol. 10, no. 4, pp. 447–455, July 2004.
- [18] D.D. Perkins, H.D. Hughes, and C.B. Owen, "Factors affecting the performance of ad hoc networks," in *Proc. IEEE Int. Conf. Commun.*, vol. 4, 2002, pp. 2048–2052.
- [19] T.H. Clausen, P. Jacquet, and L. Viennot, "Comparative study of routing protocols for mobile ad-hoc NETworks," in *Proc. 2002 Int. Federation for Information Processing Med-Hoc-Net Conf.*
- [20] D. O'Mahony and L.E. Doyle, "An adaptable node architecture for future wireless networks," *Mobile Computing: Implementing Pervasive Information and Communication Technologies*, pp. 77–92, 2002.
- [21] P. Mackenzie, "Reconfigurable software radio systems", Ph.D. dissertation, Trinity College, Dublin, Ireland, 2004.
- [22] G. Lin, G. Noubir, and R. Rajamaran, "Mobility models for ad-hoc network simulation," in *Proc. IEEE Infocom*, vol. 1, Apr. 2004, pp. 454–463.
- [23] T.K. Forde, L.E. Doyle, and D. O'Mahony, "Self-stabilizing network-layer auto-configuration for mobile ad hoc network nodes," in *Proc. IEEE Int. Conf. Wireless and Mobile Computing, Networking and Communications*, Montreal, Canada, vol. 3, 2005, pp. 178–185.